Listener Feedback #204

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-489.mp3
Quarter size (16 kbps) mp3 audio file URL: http://media.GRC.com/sn/sn-489-lq.mp3

SHOW TEASE: It's time for Security Now!. Steve Gibson is here. We're going to talk about the latest security news and then answer your questions. It's a Q&A episode. Security Now! is next.

Leo Laporte: This is Security Now! with Steve Gibson, Episode 489, recorded Tuesday, January 6, 2015: Your questions, Steve's answers, #204.

It's time for Security Now!, the show that covers your security and privacy online with this guy right here. He was in-studio last week, I'm sorry to say he's back in Irvine, our Security Now! host, our genial commentator, Mr. Steve Gibson. Hi, Steve.

Steve Gibson: Hey, Leo. It's great to be with you, as always, although from a distance. It was really fun to be in the studio. And in fact I try to decorate the show notes with a security-related photo. In this case, I chose a photo that Lisa happened to capture while I was attempting the very first ride of my life of the mechanical bull, which you had set up.

Leo: You made this face a lot, in fact, all day and night.

Steve: Did I.

Leo: Yeah. I have another one of you which I will dig up.

Steve: Not when Marilyn was on my lap.

Leo: With Marilyn Monroe kissing your forehead, yes.

Steve: Oh, I thought I was pretty happy with her doing that.

Leo: Yeah, you were happy. It was a surprised happy. Awesome. Awesome.

Steve: So for those of our listeners who don't know, all of the hours of the podcast are available at TWiT.tv/specials. And they're all enumerated there. I thought that, well, there was a lot that was a lot of fun. So I got a lot of great feedback from it.

Leo: I got this pop-up just now - this is very odd - on iTunes. Do you want to allow this computer to access information on Steve's iPhone 6 Plus? Do you have an iPhone 6 Plus?

Steve: Uh, yeah.

Leo: Is it called "Steve's iPhone 6 Plus"?

Steve: It is.

Leo: You know, it's very strange, because obviously your iPhone 6 Plus is not connected to this iMac here.

Steve: Oh, I know what it was. I was charging. And I had it plugged into the red connector, which Jeff explained was for your clone iPhone, and so he had me switch it over. And then I got the pop-up, do you want to trust this computer? And of course I said no because I didn't want to form an affiliation. But that's what that was.

Leo: The computer feels rejected and is continuing to try to affiliate with you. Isn't that strange? I've been here since then, but this is the first time that's popped up. You didn't leave your phone here, did you?

Steve: No, no.

Leo: Okay. So, well, anyway, thank you for coming. I apologize, as I have been to all the hosts, for the truly awful dinner that we served you at the hosts dinner.

Steve: Oh, god. I did - I'm glad you said that because I wasn't going to say anything. But, oh, lord.

Leo: I apologize. I have pictures. I have evidence that no one ate anything. And probably wisely so. It was a wonderful, convivial meal.

Steve: Yes.

Leo: And I just apologize because we could have chosen a nicer restaurant. We thought we'd choose something with a little local color. I didn't know the color would be brown.

Steve: Yeah.

Leo: But anyway, so I do, I apologize.

Steve: No, it was fun. It was a social gathering.

Leo: That's what it was.

Steve: And that was really the point. And, I mean, who really cared? I didn't care. We didn't go there to, you know...

Leo: Yeah, but I just - I figured, yeah, I figured probably you and everybody else were thinking, good lord, has TWiT gone bust?

Steve: But now that that's out, now that you have that out of your system, we're all glad.

Leo: It was so much fun. I have some great pictures, which I'll be sharing.

Steve: Oh, my god, it was a great time. Absolutely a great time.

Leo: I'll show - this is my favorite picture. I don't know if you've seen this. This is of you at the dinner, looking over your shoulder.

Steve: Oh, I didn't see that. That's neat.

Leo: Just a great smile. And there's Sarah Lane behind you. It was odd because how many times has Lisa, doing what she does so well, going for the money, how many times have we gotten - there's Denise Howell - all the hosts together? And it's so odd to see all of you, not on Skype, but in the same room.

Steve: Well, and one thing that we did that I think we should do more of, and that was the one segment where I and three others, I think myself, Rene, and I can't remember, maybe Jeff? Yeah, I think it was Jeff. Anyway, it was the host roundtable. And I got so much...

Leo: Oh, that was great.

Steve: ...positive feedback.

Leo: We'll do that again. In fact, we should have done more of that because we had you all in-studio. And I bowed out, I actually was sitting and listening, it was so good. Randal Schwartz, who is our FLOSS Weekly host...

Steve: That's who it was, yup.

Leo: Rene Ritchie from MacBreak Weekly. Paul Thurrott from - it was Paul, right, Paul Thurrott?

Steve: It was either Paul - I think it was Jeff, I think Jeff Jarvis.

Leo: Jeff Jarvis from This Week in Google and Steve Gibson from Security Now!, all talking about kind of a wide range, I think it was kind of like a quiz show.

Steve: Just anything. Yeah, just anything.

Leo: It was great.

Steve: Yeah. But it went really fast, and we jumped around, and just it was just, you know, it's not the level of interaction you can normally get through Skype connections.

Leo: Right.

Steve: And it was atopical. It was just about whatever we wanted to talk about. And I got, I mean, of all the things that happened, that was the thing that people said, wow, we should have had more of that. So...

Leo: I agree. I agree. I agree. That was wonderful, yeah.

Steve: Tuning it as we go.

Leo: I'll find out what hour that is. Somebody's asking which episode that is because you know we put it all up on the website, on TWiT.tv/specials. So I'll find out for you. Go ahead.

Steve: So we have a Q&A this week. We haven't - we skipped two weeks, actually I guess three, because of end-of-year stuff. And I think I really liked, and I also got some great feedback, about the idea of the last Security Now! of the year being a previous year retrospective. I think that'll be something we also add to our format because a lot of people liked cramming all of the craziness of 2014 into a summary. It was like, okay, whew, we made it. Here's what happened.

Leo: Yes, yeah.

Steve: You know, because it's sort of a nice opportunity to do a review. So that we will do. Q&A 204, and not a ton of stuff has happened. We're going to talk about the HSTS supercookie that has people kind of needlessly worried, but it's an interesting, from a security theory and technology standpoint, an interesting hack. Then of course Gogo's in-flight cert spoofing has generated a bunch of buzz. I will talk about Thunderstrike that you talked about in the previous hour on MacBreak. A quick note about CryptoLocker's successor. And then of course we've got 10 questions and comments and thoughts from our listeners. So another great couple hours here.

Leo: Marvy. Chatroom is saying that it was Hour 4. I can't believe it was Hour 4.

Steve: No, it had to be later than that.

Leo: Yeah, I think it must have been. No, I don't think all those people were here. But maybe. I don't know. For me it's all a blur, to begin with.

Steve: And Leo, I've got to say, I'm impressed. It took me 48 hours to recover. I came home, and I slept for 15 hours.

Leo: Yeah. Me, too.

Steve: I went to sleep at 7:30 and got up at 10:30 the next day. I just, you know, it just knocked me out. But, boy, you really...

Leo: Oh, I did the whole thing, obviously. It was TWiT Live Special #212. And there you have it. And believe it or not, so it was like 4:00 and 5:00 in the morning. Something, I don't know, 7:00 in the morning.

Steve: Well, I know that Rene and Randal both were planning to be part of the Breakfast with Steve, and we moved that forward, and so...

Leo: I never did have breakfast, by the way.

Steve: No. But you and I had a neat...

Leo: I was counting on breakfast. They told me there'd be breakfast. But this was good, and I highly recommend it. We should probably just put this out as a little bit by itself because it was so good. But it's about, let me look here, it's about halfway, a little less than halfway through TWiT Live Special #212. Looks like it starts about - after the trust exercise with Josh, about 15 minutes in, there you go. It was such fun, and we opened so many bottles of champagne.

Steve: And by the way, I've been hearing you talk about understanding the need for body hair. And of course that's why I wear the cap. I was just reminded because there I have my little French cap on, as I do. And it's because I keep my hair so short that, as you have found, it really is cold.

Leo: It is not - hair is not an evolutionary mistake or leftover or vestigial. It really protects your head.

Steve: And we know that you lose a lot of heat through your head also.

Leo: I've been wearing a cap to bed. A nightcap.

Steve: I think our brain uses, what is it, 20% of our body's total energy budget?

Leo: There's a lot going through there, yeah, yeah. And this is just a perfect radiator now.

Steve: Yeah. Ah, it'll grow back in no time.

Leo: Anyway, it was fun. We raised $75,000 at final count for UNICEF, a little more than 10 of that through the auction, and about $61,000 through the...

Steve: Nice. Just donations.

Leo: Direct cash donations. And you'll feel good about this, that 91% of the money UNICEF receives goes right to the kids. It is one of the most efficient charities out there. So I feel very, very good about what we did. It was worth getting a tattoo and a haircut. I tweeted, I woke up with such a sore butt. What happened last night?

Steve: Such a good sport, Leo. Yeah, when that happens in Vegas, then you have a

problem.

Leo: I was looking for a tiger. There were no tigers.

Steve: In Petaluma. Petaluma.

Leo: No crazed monkeys.

Steve: So before I forget, I've also been listening to you talk about "The Imitation Game."

Leo: Which I haven't - I just got the DVD. So I'm talking through my hat, as they say.

Steve: Wait, wait. Did you get "The Imitation Game" or the invitation?

Leo: No, no. I should explain. As a member of the Screen Actors Guild, I get screeners for nominated movies.

Steve: [Gasps]

Leo: And that's one of them. So I got the screener.

Steve: What?

Leo: I know. It's not...

Steve: Wow. Wow.

Leo: Life's not fair.

Steve: Wow.

Leo: But I haven't seen it yet. And I want to see it. It's the story of Alan Turing.

Steve: I wish I knew you, Leo.

Leo: You know what? They're so paranoid about piracy that you have to click a button before you can watch the movie on the DVD that says you will not lend or give this to anyone else, and you will shred the disk after you're done watching it.

Steve: Wow. Okay. So the point I wanted to make was, because I've heard you talking about the way Hollywood portrays...

Leo: I'm nervous.

Steve: ...mathematicians. And the way I would characterize the portrayal of Alan Turing is as a prima donna. I think that's the perfect word is that - and I'm not giving anything away because you can see it in the trailer that they keep running.

Leo: Yeah, it's obvious, yeah.

Steve: Where he says, well, "The code is supposed to be unbreakable. Let me take a crack at it, and we'll find out," you know, as if he's going to be the ultimate authority for crackability of this unbreakable German cipher. So anyway, I really liked it, although - and I have to say that his prima donna-ness, it would be nice to know who the real Alan Turing was because of course now we have this particular view of him. But it was crucial for the plot that that be his character. So it could well have been accentuated, exaggerated.

Leo: Well, and that's, you know, some my critique, I haven't see it, so I will watch it, and then I'll come talk back to you next week. First of all, all for people, the general populace, understanding who Turing was, his genius, he's the father of modern computing in many ways and was persecuted horrifically by the British government in a shameful fashion. And I'm all for anything that tells that story. I just - I think that, just as geeks are often misrepresented in mainstream fiction as being geeky, mathematicians are often - I'm thinking of "A Beautiful Mind" and "The Social Network." The numbers appear through the air. And I've read some reviews that say this is perhaps not doing him a great service. His other achievements aren't mentioned. Maybe it's not as, you know, this is a thriller, and so they have to make it a thriller.

Steve: It did induce me to figure out exactly how the Enigma machine works.

Leo: That's fascinating; isn't it?

Steve: And I thought I would give our listeners an explanation of it next week.

Leo: Would you do a show on the Enigma machine?

Steve: We could. I mean, it turns out it's really...

Leo: Oh, that would be awesome.

Steve: It fits into the discussions we've had before.

Leo: It's crypto.

Steve: And it, well, it's...

Leo: Mechanical crypto.

Steve: Yeah, I don't want to give it away, but I completely understand it, and it's not so complex that we could not explain it in an audio podcast. So it's a great - it would be a great topic.

Leo: I would very much like to do that. I've seen an Enigma machine in a museum somewhere, and it was quite amazing.

Steve: I'll explain it next week. That'll be our topic for next week.

Leo: Yay. Great.

Steve: So we had a Chaos Computer Convention that just occurred. And, oh, no, that's the wrong topic. Well, okay, I'll talk about that first. And you did talk about this during MacBreak Weekly, the so-called "Thunderstrike" exploit. And, yes, as you also noted, we've commented, we did over the holidays, for the holiday podcast, that you really have to have a good name for an exploit in order for it to get hooked, or for it to really get picked up. And of course the famous one was Heartbleed, where it came fully with a website and its own logo when the Heartbleed vulnerability hit us last year.

Thunderstrike has at least a good name. And we already anticipated the problem on this podcast because I did mention some time ago that Thunderstrike offered the same, you could almost argue too much, power that the Firewire interface does, or did, since Firewire is sort of fading. And that is, it is a very high-performance direct connection to the system bus. And it allows the peripheral to be a master on the bus, not just a slave. And if the peripheral can be a master, it's able to generate both the addresses and the data. And that means it's like another processor. It's like something as powerful as the processor outside the case, which you're connecting through a serialized interface. So, and even Wikipedia, they have a page called "DMA Attack" because this is a direct memory access (DMA) vulnerability that's built into the specification. And on the Wikipedia page they say, "Examples of connections that may allow DMA in some exploitable form include FireWire, ExpressCard, Thunderbolt, PCI, and PCI Express. And that's absolutely right. Those are all bus-level interfaces. Now...

Leo: We've seen a FireWire attack; right? I mean...

Steve: Yeah, yeah. In fact, there was a famous credential key extraction through FireWire where somebody plugs a little box into your Mac when you're not looking and is able to rummage around in RAM. I mean, they have complete access to memory. Well, it turns out that, in the case of Thunderstrike, there is, similarly, bus-level access to the firmware. The firmware of the Mac is just a region of memory. And what this Thunderstrike does is leverage its access to the firmware in order, well, I mean, to the entire hardware of the machine to rewrite the firmware and then change the public key which is in the Mac, which is used to verify any additional firmware updates, so that it can't be removed. So the good news it is hardware-level physical access required. The bad news is, if you get infected with this, you can't get rid of it.

Now, Apple has already released some updates for a current - one of the minis and something else. And they will absolutely, you know, they're responding to this as quickly as they can. And there are things they can do to mitigate this because there are controls that are apparently not in place at the hardware level, where you can restrict the regions and ranges of memory access through Thunderbolt. And so Apple's going to get better about doing that. So I'm sure we'll see some updates to these systems, and they're already pushing them out for a few platforms. And I'm sure they'll give us coverage because this is not good.

Leo: Yeah. But you do have to have physical access to the system to do it; right?

Steve: Yes, yes, yes, yes. You've got to plug something in physically into the Thunderbolt connector. And really what you're doing is you're plugging it...

Leo: Well, like BadUSB you could actually have a corrupt device; right? Would that be possible?

Steve: Sure, sure.

Leo: So you could have a corrupt Thunderbolt drive that would then infect the computer.

Steve: Right. Yeah, I saw some notions of like a crossover with NSA, saying that this is the sort of thing that some of the Snowden slides were implying that they were able to do. And so we may be foreclosing another one of their tricks by locking this down and being more secure with what Thunderbolt is able to access.

Leo: I'm sure there'll be more.

Steve: Yeah. Okay. So news also broke of an HSTS supercookie. Now, okay. HSTS is the HTTP Strict Transport Security, so HSTS. And that's the feature which allows a website, if you first have a secure connection to it, it allows it to send you a reply header in

response to your browser's query that specifically says I want you, Mr. Browser, to henceforth - and actually in this response header is a max age, which starts a timer that says, for the following number of seconds, only ever access this site over secure connections. And what it does is it specifically - having this HSTS, the Strict Transport Security header, in the reply specifically allows the browser to autonomously upgrade any non-HTTPS, that is to say, any HTTP connections, to SSL/TLS connections until that max age expires. And there's an option on there, "include subdomains."

So, for example, GRC has long been HSTS, Strict Transport Security. And GRC's max age is the max allowable. It's something like 31,536,000 seconds, which is like, you know, forever. And so browsers all over the planet who have ever visited GRC, all have received - everyone who visits is receiving that reply. And I'm saying don't ever even try to access over HTTP.

Now, the reason we do this is that, remember, there was an attack which was being exploited where, if a bad guy could intercept your first HTTP access, they could strip out the HTTPSes from the response and prevent your browser from upgrading its security to SSL/TLS, to secured. And so the problem was that first access created essentially a persistent vulnerability because most users just put in the domain name. And the browser defaults to HTTP, if you just give it that, because that's been the standard in the past. And so all initial connections, even if the site wanted to be secure, the initial contact would be nonsecure. That creates an exploitable window.

So what the whole Strict Transport Security effort has done is, once the browser gets you to secure, it tells the browser, remember, in its own sort of separate HSTS cache, remember for this length of time you have permission to never connect to this domain non-securely. If any URL comes along that matches that domain, you the browser upgrade it. And so what that does is, after a first contact where the browser receives that cookie, or it's not really a cookie, it's a response header, when the user puts in GRC.com, the browser doesn't any longer go to HTTP the first time. It itself sees, oh, that's in my HSTS cache forever. I'm going to automatically make my first connection secure so the bad guy is cut out of the loop from the beginning.

Now, what a clever hacker recognized - and what's interesting about this is this is not news. This was in 2012 it first came to someone's attention that there was a way, if you could run JavaScript on the client - and so NoScript, for example, or any script blocker forecloses this automatically unless you deliberately run scripts on the site - it would be possible to probe the HSTS cache in the browser by having the JavaScript make a bunch of queries to subdomains to see whether they were upgraded or not, which is, really, it's a hack, but it's a clever hack. So the idea was that you would go to a site that really was determined to track you. And script running on the page that you received would issue a whole bunch of nonsecure queries out to - call it, like, a.trackme.com, b.trackme.com, c.trackme.com. And your browser would contain a unique combination of HSTS upgrades for a pattern of the subdomains of trackme.com.

And the point is that it does create a supercookie. And what's interesting is that, because the other privacy-enhanced modes are trying to protect their users, the incognito mode or private mode, whereas cookies may not cross that incognito or private mode boundary, HSTS information does. So you think you're being sneaky, going into incognito mode. But if you went to a site that was actually doing this - and by the way, this is all just sort of theoretical - then script running on a page you receive from such a site could make it blast out a whole bunch of queries and essentially get binary bits, a one or a zero, for every one of the subdomains that it tested, and then build up a composite ID which would tend to be sticky.

Now, again, because this is two years old, Chrome and Firefox and Opera have long since dealt with this. If you erase cookies, they flush the HSTS cache. Now, that's actually not a good thing to do because you want your HSTS cache to protect you. That's why we have Strict Transport Security. You'll get it back as soon as you visit sites again, but you don't get it back the first time you visit those sites. But the problem is this is a problem that no one has a good answer for, no one has a good solution for.

And so what Chrome and Firefox and Opera have done is they said, okay, well, we'll flush this information when people erase cookies. Safari provides no provision for this. And on iPhone and iPad, and I don't know on the Mac for sure, but at least on the iPhone and the iPad there is absolutely no way for users to clear the HSTS information through any of the UI. And the iCloud sync syncs HSTS information. So if you wiped your device and then resynced it with the iCloud, you would get back this potentially sort of kind of hacker-esque flaky supercookie.

Anyway, that's what that is. I got a lot of tweets from people saying, oh my god, you know, HSTS can be used against us. It's like, yeah, okay. And in fact there's a test site, RadicalResearch.co.uk, that has an example. And when I went there, it came up blank because I've got NoScript. And I saw that it wasn't working, and so then I permitted NoScript to run, and then it gave me a token, sort of a crypto-looking thing, about eight characters' worth or so. And then they said, okay, now if you wander around, this thing's going to follow you wherever you go. It's like, okay, yeah, with scripting enabled, and with this kind of flaky HSTS hack. So that's what that's all about. I don't know that it's anything really to worry about. Browsers may start looking. Browsers may take additional action. It'll be interesting to see how this evolves, if it rises to the level of additional concern within the browser developer security community.

And then the other interesting bit of news is that a company called Gogo, which is doing in-flight Internet access, was caught by, interestingly enough, by a Google employee, minting a Google cert that they sign. So this person was using Chrome, that won't put up with any nonsense with Google certificates because Google is really protecting themselves. They've got certificate pinning going on in Chrome; and if there's anything in any way screwy about any Google cert, Chrome will alert you. So she got alerted when she was doing some in-flight Internet use and tweeted that this Gogo company was essentially doing what we've talked about many times, that is, trying to perform a man-in-the-middle interception, I won't characterize it as an attack because we don't know that it is, but an interception of her secure connection to YouTube.

And what they've said is that they're trying to block access to YouTube. But that doesn't really hold up to technical scrutiny because all of your traffic is going through their proxy. So all they have to do is block access to YouTube. I mean, like IP-level access. Or they're also providing all the DNS, so they could redirect YouTube.com to an interception page that says, we're sorry, you cannot stream video through your Gogo session while you're flying through the friendly skies.

But they don't do that. Instead they attempt to proxy the SSL, the secure connection that YouTube is attempting to establish with your browser. And a warning comes up saying this is an untrusted certificate. Do you want to proceed? And so here's the concern that's been raised, and that is, we absolutely do not want to train any fraction of the world to click through those, to say, yeah, yeah, I guess I have to, I guess I'm supposed to say yes to this, the way they're supposed to say yes to license agreements and things.

So first of all, it's not clear at all why they're doing this. I think this is just lame. If they want to block YouTube, block it. Don't try to intercept and decrypt. I mean, they, like,

went to some serious extremes to get into the YouTube traffic, rather than just blocking it. So I'm skeptical of their response at all. And Ars Technica sort of underreported this. Dan Goodin normally does a great job. But the Ars Technica story said: "Gogo has been caught issuing a fake digital certificate for YouTube, a practice that in theory could allow the in-flight broadband provider to view passwords and other sensitive information exchanged between end users and the Google-owned video service." But the fact is it's much worse than that. The certificate is *.google.com and then presumably has some server alternative name fields to allow it to do things also like www.youtube.com.

So my point is this certificate that we saw a screenshot of because this Google employee took a screenshot and tweeted it, it's carte blanche across all Google properties. So this is much more than just intercepting the Google-owned video service. This is really bad. And so the last thing we want is for people to think, oh, I guess I should accept this fraudulent certificate in order to proceed. So anyway, it's troubling to see this happen, and I hope that there's a big backlash because we need to not have this become standard operating procedure when people are wanting to use a third party's Internet provisioning. That just is way bad.

Leo: Is there a nefarious reason they do it this way? Or is it just kind of goofy?

Steve: It feels - I don't know. They're saying they want to prevent people from doing YouTube streaming.

Leo: Yeah, I understand that.

Steve: So block the IP.

Leo: Right, I mean, it's easy to block YouTube. I'm sure they block Netflix already.

Steve: Yeah. So do that. But what they're doing is they're trying to get people to accept a *.google.com certificate which they have signed.

Leo: I mean, that's just bad practice. I understand, but...

Steve: Yeah, yeah. And so...

Leo: But is there anything after you get off the plane? Now are you compromised in some way?

Steve: If you were to statically accept that, then yes. Then if you were accept them as a CA that has signed the Google cert, then that's a static compromise of your machine that you'd be carrying around with you.

Leo: That is bad.

Steve: Yeah.

Leo: Yeah. Don't accept it. But a lot of people would. And if you weren't on Chrome, you might not even know.

Steve: Right. Well, no, anything...

Leo: You still get a warning.

Steve: Anything's going to pop up and say this is an untrusted certificate. Do you want to proceed? And that's my concern is that this is - everyone should always say no. But if a service that you want is putting themselves on the other side of this, then we're going to start training people to say, oh, well...

Leo: Right, right, right, you don't want that.

Steve: No. And that...

Leo: But maybe they're - are they caching YouTube content on the plane or something? I mean, it seems like, I mean, obviously it's a long way around to do this. So I'm trying to figure out why they did it this way.

Steve: Yeah. I cannot give you - blocking is easy. I did see the word "throttling," so I don't know what that means, like throttling YouTube? Well, then the video's not going to work very well. And again, you don't have to get involved in the decrypted connection. You could throttle the encrypted connection just as well. So, I mean, I can't - I don't want to ascribe anything nefarious, but I think they must be doing more than they say they're doing, whatever that is. But the concern is people are going to get trained to ignore security certificate warnings, and that's not what we want.

Leo: Right, that's not good. Dr. Mom says she noticed this behavior months ago. So it's probably not new. People just found out about it.

Steve: Right, right.

Leo: That's interesting.

Steve: So I did want to mention CryptoWall 2.0, only because it's not going away. It's six months old, and I just haven't mentioned it. This is the successor or the family

member of the CryptoLocker society. We of course have beaten CryptoLocker to death. And we said at the time that there would be more of this, and CryptoWall is more of this. It is similarly, unfortunately, well designed. It is encrypt-your-files extortionware, where you need to pay them in order to get the key in order to decrypt your files. And there was just an article in The New York Times, "How My Mom Got Hacked Over the Holidays." And that just sort of reminded me, you know, this isn't gone, and it is really bad. And so the good news is AV is really on top of this. And so as long as you've got current antivirus, you've got the best protection we know of. But if you're going to do a sponsor insert, Leo, about anything about backing up, this would be a great time to put that in because that's really the only thing you can do.

Leo: I don't, unfortunately. But you want to take a break now? No, keep going.

Steve: Yeah, we'll keep going. But anyway, I'm just saying...

Leo: We don't have a backup sponsor; but, yes, that would be a good thing.

Steve: That is, we do know that Carbonite is often a sponsor on the show, so that or whatever you want to do. But ultimately we're seeing people having their lives turned upside down. Well, I mean, they have to pay $500, so it's not the end of the world. But still, it teaches people a lesson. And many infection vectors. It's phishing. It's malware that you may already have on your machine. It's clicking a link. It's opening a PDF. It's like links to an Excel spreadsheet that's exploiting a vulnerability in Windows. I mean, the things that people did in the past to get themselves infected can now be getting themselves infected with something that encrypts all their files, rather than just sort of says thanks for letting me borrow your computer, I want to attack people with it. So, yikes.

Okay. Miscellany. A couple things. I rebooted my server. Leo, you and I were talking a week ago about how, you know, I - now if you go to SSL Labs and check out GRC.com, I got a better grade than I predicted I would a week ago. I am, GRC.com is back to an "A." So I have a nice green "A." I do not have the "A+" that I had for a while because I'm deliberately staying with SHA-1 certificates, both to thumb my nose at Google and their effort to force everybody off of them before necessary because there's nothing anyone knows that's wrong with SHA-1 except Google wants to preempt everyone waiting until Microsoft's 2017 drop-dead day.

Leo: You just wanted a better grade at SSL Labs. Let's admit it.

Steve: Well, yeah. No, people were saying, why are you a "C"?

Leo: Why are you a C"?

Steve: What's wrong with you? I thought you were a security expert. Apparently you're numb. So we also talked about the cipher suite list. I think it's a bit.ly link, bit.ly/grcciphers, which takes you to a text file. And remember, Leo, that I left out a comma on one line that broke it. So I waited until this weekend, when I fixed - I updated

the list to the latest and greatest. It's got the comma back in, and it's updated.

So what I did was I used a judicious selection of - and so that's an ordered list, from the one the server wants most to the one it wants least, carefully chosen so that it will give us Perfect Forward Secrecy. Those are all the ephemeral Diffie-Hellman key agreement at the top. Also the longest encryption key lengths, 256 as opposed to 128 when we can get it, and so forth. And it's something like nine - you only have 1023 characters for the total length of that when you take the line breaks out. And this thing is absolutely as long as it can be, fitting in the optimal selection of cipher suites for a Windows machine.

So what I've done is I've turned off SSL3 because even XP SP2 is able to use TLS 1.0. And I still have an SHA-1 cert; but, as I mentioned last week, mine expires on midnight, in fact I will probably be with you, Leo, when that cert expires. I, however, will have already then moved to an SHA-256 cert. I already have that, but I'm not running it because I want to stay for a year with a certificate that SP2 users, Windows XP SP2 users will still be able to use all the services of GRC. They will stop being able to use services of other websites that have been forced to switch to SHA-256. But my site won't be one that makes them switch. They'll be forced to switch to other sites, and that's my logic.

Then, by the end of 2015, anyone at that point who somehow is still not able, I mean, all you have to do is use Chrome or Firefox. They both run fine, even on old Windows. You just can't use IE, which is now the remaining browser that uses the built-in security suite for Windows. So all anyone has to do is move to Firefox or Chrome, and then you're fine. So anyway, I'm back to an "A" grade. I will have an "A+" one year from now when I finally decide that I'm going to drop SHA-1 and switch over to SHA-256.

Also, you have talked about, and this is in Miscellany, our still waiting, we're waiting for the Kickstarter Temperfect Mug. And on January 1st I received an update email from them saying: "The Temperfect Kickstarter campaign was funded one year ago today, on January 1st, 2014." So he wrote: "I considered doing a 'year in review' update for you [because it's been a year now we've been waiting] but decided that would just be long and tedious. In short, 2014 was the year we learned how hard it can be to work with a factory eight [and he says] or 12 time zones away, with a very different language, different ways of working, and a different concept of what a quality product is." And so he says, "That leaves me with just the last month to review for you." Because, as I said, I've been getting constant updates. At least they have not gone, and they haven't given up.

And he says: "We made a lot of progress in December, and things are starting to come together." And then I'll just finish by saying that one note they made was they said: "But after all the back-and-forth with the mug factory in the last months, we found ourselves with enough sample parts to assemble a few mugs ourselves. So we did. Logan and I put together a mug and tested it to see how its performance would compare to our computer model predictions and prototype measurements. The performance is better than predicted, and the temperature hold time was over an hour even without vacuum insulation."

So just to review for our listeners, the cool thing about the physics of this, their concept was that hot coffee, when it first comes out of the pot, is too hot to drink. You need to wait for it to cool down to drinkable temperature. So their concept was to create a mug that had a great deal of what I'll call "thermal inertia." That is, it's not just a vacuum container, but it's like a vacuum container where on the inside, that is, on the coffee side, it's lined in like a big copper ring. It's not actually copper, but they have something, a material, a thick layer of a material that is initially cool. And so the idea is that, when you pour this too-hot-too-drink coffee in, the heat in the too-hot-to-drink coffee is taken

up by the liner and then preserved by the vacuum seal. So what happens is the coffee temperature immediately drops to drinkable, but now it stays at that drinkable temperature far longer than it normally would.

That's the Temperfect mug. All we have is promises. But you and I have both - we're funders, and they haven't given up yet. So maybe one of these days we'll get it.

Leo: Anybody who's ever bought anything on Kickstarter realizes that Kickstarter is great, but it gives people who have no experience manufacturing...

Steve: That's exactly it.

Leo: ...a pulpit to talk about it. And they're learning on our dime.

Steve: Yes. What we keep seeing is people who don't understand that making something is not as simple as it seems on the outside. So lots of people are getting an education. And every so often we get something cool. You got your Pono Player. That was Kickstarter.

Leo: I did, that's true, yeah.

Steve: Yeah.

Leo: Yeah, I've gotten almost everything I've tried to buy.

Steve: Yeah, I think overall the idea is very cool. So I wanted to ask you, because I have not been keeping up, has anything interesting happened at CES?

Leo: Yeah. I mean, it depends what you mean by interesting.

Steve: Well, cool stuff that we have to have.

Leo: We have Father Robert Ballecer down there. We have Dick DeBartolo and Scott Wilkinson. I'm interested...

Steve: So is coverage happening? Is TWiT doing...

Leo: Oh, yeah, yeah, yeah, we're covering it on TNT every day. We have - have we put it up on Specials yet? - ShowStoppers and Pepcom, the little events that they do. And, yeah, and then Home Theater Geeks will have a special from there. And I'm kind of interested in the quantum dot backlighting that some of the manufacturers

have announced. That technology interests me. We'll see if it makes a difference in TV. Scott thinks it will. Yeah, there have been some announcements. There's a lot of announcements. I don't know, you know...

Steve: Nothing really stunning stands out.

Leo: Not yet. But it's hard to sift. It's like all Temperfect mugs. It's hard to sift the stuff that's actually going to appear in the marketplace from the stuff that seems like a good idea.

Steve: Ah. Right, right. And that's a good point. The nature of CES is that many times the manufacturers are showing prototypes to gauge reaction and to see if it - and also just to say, look what we were able to do. Yes, we're able to do a 16K display. We don't add any wires, and we can't get the image to it fast enough; but, still, look at that screen.

Leo: Well, you heard, who was it, Sony that has the really, like, a TV, a big-screen TV thinner than an iPhone.

Steve: Ooh.

Leo: But, you know, I don't mind if my TV is an inch thick. I really - it doesn't really affect me.

Steve: No, true. So okay. So I did get a nice note, a holiday note from a Tim Green in Germany. The subject was "SpinRite fixes PS3s, too." And he said: "Just for a change, SpinRite also fixes PlayStations. We have a relatively old PS3 that we use mainly as a DVD and Blu-ray player and media center. It started acting up recently, hanging on boot-up and exhibiting other strange stutters and pauses at unexpected times. So I took out the hard drive, connected it to a PC, and let SpinRite at it on Level 2. Just over an hour later, I put the drive back in the PS3 and booted up for a complete transformation. Not only has it stopped hanging on startup, but it now also feels generally snappier and smoother.

"Thanks once again for the only really useful hard disk maintenance and recovery utility I have ever encountered, and also for all your insight, information, and wisdom in the show every week, which I have listened to as unfailingly as you have produced it ever since Episode 1. All the best to you, Leo, and all of both of your loved ones for a happy and joyful Christmas break. Thanks again. Tim Green."

Leo: All of both of our loved ones.

Steve: Thanks very much. All of our loved ones.

Leo: I understand what he's talking about.

Steve: We're all covered.

Leo: All of both of them.

Steve: Yeah.

Leo: All of many. Dr. Mom sent me an email with a certificate from her hospital. I should show you this. It's kind of funny. She was visiting GRC.com, and the hospital replaced your certificate with one of those man-in-the-middle things.

Steve: Oh.

Leo: But a lot of businesses do. This is not kind of what Gogo did, really. But, you see, she's on ShieldsUP!. It says GRC.com, but it also says North Shore-LIJ Health System, Roxbury, New York. That's common, though.

Steve: Yup.

Leo: Lot of businesses do that.

Steve: Well, yes. And once what would normally happen is you would embed the proxy server's certificate in your browser, and then you would no longer get those messages. The message only happens...

Leo: The warning, right.

Steve: Right. The idea being that it says, hey, the cert is signed by somebody we don't know.

Leo: Right. Says you're on GRC.com, but they didn't sign it.

Steve: Right.

Leo: What Gogo's doing is different because they say Google signed it. Right?

Steve: Yes, correct.

Leo: That's a bridge too far. Or something.

Steve: Yes.

Leo: Is that you? You have an alarm going off?

Steve: Yeah, I don't know what's going on. Someone is doing a dance out in front of the house.

Leo: Okay.

Steve: They've gone away.

Leo: Question time. Number one from James Allen. He's looking for clarification on the SQRL cross-site tracking and adoption. Steve, thanks for your efforts on the Security Now! podcast with Leo. I'm new to security, and it has become a weekly staple of my life. I love SQRL. SQRL, I love you. I just have one nagging question in the back of my head, and I wondered if you'd considered this and what your opinion is. One of the features which is great for users is the per-site identification tokens, public keys, which are unique for each site a user visits. This means that, unless the user volunteers to the website additional identifying information, an email address let's say, one site, as an example Audi.com, cannot match up the user's identity from another site, say Porsche.com.

This is great from a user's perspective as it gives us more control over what websites know about us. But it seems to me it's not good for corporate groups like Volkswagen, who owns both Audi and Porsche, because they can no longer track users across their multiple services, or sneakily sell on your information to other companies - again, great for us, not so good for the people making money off of us. This seems to me a potential reason for large and/or unscrupulous corporations not to implement SQRL. Is this going to be a problem, to get some adoption?

Steve: Okay. So of course the podcast before last was the Christmas holiday, and we played the presentation that I gave during the DigiCert Security Summit in Las Vegas. And as a consequence, I got a lot of this, and I wanted to address the concern. I think I overstated, without exaggerating anything, but I've confused people. So the cool thing about SQRL is, well, one of the many, is that it does synthesize a per-user identity. But that's all it does. That is no different than a user synthesizing a per-site password.

So the tracking thing isn't something that SQRL prevents in any way. And so that's what's got people confused. Because SQRL generates a per-site identity, people think, they assume that that somehow prevents them from being tracked. But the tracking happens, for example, at the cookie level, or the supercookie level, which is different from your identity. So a bad identity system would be one that explicitly, where like for example you always had to have the same username and password for every site, well, then they can obviously track you by your username and password. But people know better than that, so they often, if they want anonymity, they'll create different

usernames and passwords per site. SQRL just does that automatically.

So anyway, many people got confused by that, so I just wanted to take this opportunity to explain that it's identity that it creates uniquely. But, for example, if you receive a cookie from Volkswagen, and then you go to the Audi site and they have an image or a tracking beacon over to the Volkswagen.com site, that cookie will follow you. So all the tracking stuff isn't changed one way or the other. It's just the identity which is unique per site. And so it gives you sort of a head start, I guess, on the tracking problem. But by no means did I want to oversell that or confuse people.

Leo: Well, and you make an excellent point, which I hadn't really thought about. But if I use my email as my login, which many sites encourage you to...

Steve: Yes. Yes.

Leo: I'm not going to make up a new email address for every site. I use the same email all the time. So that means they know who I am.

Steve: Right, same you. Yes, exactly.

Leo: If you log in.

Steve: Right.

Leo: But obviously there's no point for SQRL if you don't. Right? I mean, that's the whole point is authentication, yeah.

Steve: Right. And at least SQRL doesn't give away your identity. At every site you visit, you just get a unique random gibberish blob, and they go, oh, okay. Well, you going to tell us anything more about you? And then [crosstalk].

Steve: Exactly. Oh, he's back.

Leo: It's Blob again. Right, they don't know my email address. And that's, to me, right there, a huge improvement.

Steve: Yup.

Leo: I forgot that just by giving them the same - I always use the same email address. Even if you didn't, maybe - but you're not going to make up a new address for every site. That'd be...

Steve: It would be really burdensome, yup.

Leo: Pat Cho, Sacramento, wonders whether it's safe to log into POODLE TLS vulnerable sites. Steve, Fidelity is still vulnerable - ah, this is interesting to me, I use Fidelity - to the POODLE attack, according to SSL Labs. How much risk is there for someone to log into their servers while they're vulnerable? It doesn't seem like getting it fixed is a very high priority for Fidelity, even though they assure me they take security very securely. Pat. Or seriously. Or something.

Steve: Yeah. So, okay. A server is vulnerable to POODLE if it still supports SSL v3.0. TLS 1.0 and above isn't a problem. And so POODLE came back in the news, our listeners will remember, because it is possible to downgrade the browser's intent to support TLS back to SSL. And if the server also supports it, then they negotiate an SSL 3.0 connection, which then can be theoretically attacked. So what everyone now should do is simply disable SSL 3.0. It is that simple.

So to answer Pat's question, it's always been the case that the POODLE attack is more theoretical than real. I mean, on a spectrum of things, we're preempting something that could be done. But remember, to be done, you have to get malicious script from a site on your browser, and then it has to generate thousands of queries which a man in the middle alters, because the browser is unable to do it, in order to probe error messages coming back from the server about the handshake being broken. And after thousands of queries, you are able to determine one byte from the headers that might be from a cookie that might contain information you care about, that might create a problem.

So the answer is, is this really a problem? It's hard to imagine that it is. But the fact that Fidelity Investments, who say they care about security, don't care enough to turn off SSLv3 when that's all it takes, that's the bigger cause for concern than that you could actually get bitten by POODLE.

Leo: Well, they may have a lot of little old ladies using it with Windows 95 or something; right?

Steve: But even, well, good question. I don't know how far back 3.0 goes.

Leo: They know. They know what their customers are using. And they'd probably say, you know what, if we turn it off - it's a minimal threat. And if we turn it off, we're going to have a big compatibility issue. I would guess.

Steve: And it is true, it's why I didn't bother with it until I finally rebooted my server. It's like, yeah, okay. I mean, no one's ever seen an exploit in the wild. No one's ever exploiting it. And as I've always said, the bar is very high for actually getting bitten by POODLE.

Leo: But once you get bitten by a POODLE, you'll never want to get bitten by a POODLE again.

Steve: Yeah, that's true. Especially if it's not one of those little mini POODLEs. If it's the full-size, a maxi POODLE, that'll bite you.

Leo: Owie, yeah. Steve Nagy, an "Average Joe in Tampa, Florida," self-described, muses about Sony's Security. Hey, Steve and Leo. Love the show. Been listening now for about a year. Got to tell you, the more I hear, the more I wonder if anyone is ever really secure. My question is, what if Sony, or anyone else for that matter, had used some sort of dongle or YubiKey for critical security logins? Would this have at least hindered system-wide exploitation? Thanks again for a great show. Steve.

You know who famously does this is Bloomberg. They put terminals, Bloomberg terminals, in all sorts of places. And the terminals are very expensive. Investment folks, stockbrokers and stuff buy them. And they have a key, a card key with a temporary code. It's a cycling, one of those VeriSign-style cycling card keys that is a swipe and a code that is required to use it. And nobody to my knowledge has ever broken into Bloomberg's terminals. That's a valuable asset.

Steve: So I liked this question because the whole issue of remediation, not only recovery but lessons learnable, to the degree that this is - like the topic of the podcast a few weeks ago was expensive lessons. So we'd like to learn something from this, rather than just, oh, my god. And I, of course, famously said, I don't think I could secure Sony Entertainment. Not while their network is as functional as they need it to be. But people keep asking, what could you do, then? I mean, what could be done?

And the thing I think, I think the takeaway lesson from the experience is that there are varying levels of integrity. That is, many security models have this notion of, we'll call it "rings" because, for example, even the Intel chip has multiple rings. We talk about Ring 0 and Ring 3. There's actually a Ring 1 and Ring 2, but no one ever bothers with them because it turns out having the kernel and having not the kernel is really all we needed, but the architecture's got four levels with gradiated permission.

But that's sort of the idea, where you'd have all of your administrative assistants and your outside people and probably your VPNs and all that stuff would sort of have access at the outer ring. And things like financial plans and the stuff that was really painful for Sony to lose, presumably you have a smaller cadre of C-level and related executives upon whom you actually could impose stricter requirements for authentication where, yes, in return for their access to the crown jewels, they're going to jump through some hoops. And they can be expected to have a greater burden of inconvenience in return for needing to secure the more important data. We don't have any evidence that there was any sort of hierarchical security. But that's the architecture that can be imposed that is still practical.

Unfortunately, email is still going to probably be embarrassing, that's probably going to be out largely in the public. But you could also have protected email servers for the inner sanctum executives that are kept in a more highly contained environment. So I think that the security model that needs to be imposed is not one that is flat. The evidence is Sony was operating in a flat model. And you really can't do that. I think moving forward, if Sony's going to restructure themselves, they need to do it in a hierarchical model where people who have more privileges have a greater burden of honoring the security of that privilege. And that's a mode that could be implemented that would not be unduly burdensome to the organization overall. And I think that's pretty much what you have to do to secure something of that size.

Leo: Well, and maybe not use email. I think a lot of organizations have moved to

solutions other than email for internal communication. Secure messaging can be completely secure; right?

Steve: Ah, right. So there just isn't that persistent record of stuff that you really don't need to keep a record of.

Leo: Right, and you may not want to for a variety of...

Steve: Yeah, exactly.

Leo: But then there may also be, and I don't know what the mandate is on this, but I do know that you can't destroy email in a business. You need to keep it for a number of years so there's a paper trail for legal reasons. So there may be legal ramifications that also make this difficult to do. I don't know.

Steve: Is it the case that you cannot destroy it? Or if you have it, you cannot destroy it?

Leo: Well, presumably you have it.

Steve: I don't keep mine.

Leo: What was the rule? Maybe somebody in the chatroom knows. But I feel like you're supposed to - it's Sarbanes-Oxley, so it's not everybody. You're not a publicly owned company.

Steve: Okay. Okay.

Leo: But Sarbanes-Oxley requires a certain amount of email need to be preserved for a certain length of time.

Steve: Okay, that makes sense.

Leo: Yeah. Sarbanes-Oxley was the rules, the law that was passed after, was it Worldcom? No. Some famous failure of a company lots of people lost money in because the company was basically hoaxing, messing around with its books.

Steve: And so then the idea would be...

Leo: Enron, Enron, thank you.

Steve: Ah, okay, that makes sense.

Leo: That's the Enron scandal.

Steve: So the idea would then be that a company that...

Leo: It has to be publicly owned.

Steve: ...is publicly held, then is able to affirmatively respond to a subpoena to produce the email. So if they're sued, they're going to have to say, okay, here's our stuff.

Leo: You have to preserve it. You can't throw it out. Yeah. And I don't know if Sony Pictures Entertainment was publicly held. I'm pretty sure it is. So I don't know. I don't know. Don't ask me, I'm no attorney. Rafael Beraldo in Brazil wonders about GRC's Windows Servers. Hey, Steve, Rafael says, why not Linux? Longtime listener here. As a Linux guy, and knowing you like the BSD family, I've always wondered, why you use Microsoft's IIS to run the GRC website? Why not run it on one of Unix's spiritual successors? I decided to send this question after Episode 486, when you mentioned that Windows has a very small string size for supported TLS ciphers. I'm curious to hear your thoughts. Thanks.

Steve: You know, people ask. And more than anything, it's just sort of an accident of history. I'm a Windows developer. I've been writing Windows client-side stuff, well, for 25 years. That's SpinRite, and even before that, FlickerFree was my previous software, have always been on Windows. And then I had my ChromaZone screensaver that I wrote. Anyway, so when Microsoft was offering a server and used sort of the same OS that I was already a programmer of, I just sort of went with it. I think if I had it all to do again, it would absolutely make sense to have used a fully open solution.

The good news is that IIS allows deep hooking, that is, the web server allows me to insert my own code in front of it and behind it. So there's like a little core now of IIS that runs, and then the so-called GRC Net Engine is what I call it, it just sort of, like, stomps on top of it. And so all the extra stuff that GRC does is my own code running where IIS just sort of does the low-level grunt work of serving simple queries. But things like the DNS Spoofability Test, the Perfect Passwords, all of the server-side stuff I am writing in Windows. So even now it's convenient for me because, as I said, I'm a developer that knows the Windows API pretty much inside and out. So it's been practical.

But I do wish that I was on open source platform. And again, I'm not going to go rewrite everything now for it. But if I had to do it again, yeah, I probably would have chosen FreeBSD. That's my platform of choice when I'm not on Windows. And I do have a FreeBSD server running. My DNS server at GRC is on FreeBSD. I'm running a true NNTP server on FreeBSD Unix. So I've got one, but I just don't do that much with it.

Leo: Infosec Institute says IRS requires seven years, payment card one year, California Franchise Tax Board four years. This is email retention. DISA Security Technical Implementation Guides state one year. Many state revenue departments

three years. HIPAA, six years. So you may have regulations that require that you preserve email.

Steve: Right.

Leo: Don't listen to these guidelines. Ask an attorney. Damien in Nashville has been told TrueCrypt has a backdoor: Steve, thanks for all you do. I'm writing today about something a security consultant told me today. When I mentioned I use TrueCrypt volumes to secure some more sensitive items on my computer, his response is, "Oh, we're going to have to get you off TrueCrypt." When probed about why, he said, "It was deemed insecure long ago and has been found to have a major backdoor. I can't tell you the details; I'm under NDA. But if you do enough googling, there have been papers presented on how to break TrueCrypt." What?

I will admit, I haven't dug into this for more than the past few hours. I also haven't been keeping up with the show the last couple months, so I apologize if you've already covered this. But is TrueCrypt truly broken? No, but I think your security consultant might be. Is there any hard evidence that something isn't right with it? I don't see any change in your TrueCrypt archive pages, so I'm thinking your stance is still the same as it was this summer. Can you reassure me - or put me in my place? Thanks for everything you do. Your work is truly appreciated by the community.

Steve: So Damien, you're not being put in your place. You're being reassured. However, you can, as Leo suggested, put your security consultant in his place.

Leo: I'm a security consultant. I'm sorry, Steve, I know this stuff.

Steve: We've seen these clowns who sort of adopt a know-it-all attitude.

Leo: Oh, yeah. No, you must have seen that, yeah.

Steve: Yeah. And the first thing to worry yourself is anyone who purports to be a know-it-all because none of us know it all. One of my favorite things is to say "I don't know. I can go find out, but I don't know." But, yeah, there's no papers that we are aware of, or the industry is aware of, or anyone is aware of, that this guy is suggesting talks about a TrueCrypt backdoor. We don't know that there is one. But the only thing that is bad about TrueCrypt, arguably, is that it is now unsupported. So it is unsupported, as far as we know flawless, software.

If you prefer to use supported stuff, well, there are forks of TrueCrypt source which people are developing, and there are entirely non-TrueCrypt things like BitLocker, and the Mac's got whole drive encryption now and so forth. But if I were to choose something, I would choose - and if it were to work. The other problem is that TrueCrypt will start getting long in the tooth when the platforms we start moving to are no longer TrueCrypt compatible. So that will become a problem over time. But today, I like TrueCrypt better than solutions from the manufacturers because, frankly, I don't trust the manufacturers; and I do trust the spirit and the intent of the guys who wrote

TrueCrypt.

Leo: Well, and there's good reason not to trust the manufacturers. They do business in the United States, and they may have been compelled by the U.S. government to provide a backdoor. And we would not know about it, and they would not be able to say no.

Steve: Nope.

Leo: Whereas TrueCrypt, which was run by, as far as we could tell, a couple of guys from - where were they from? Lichtenstein or somewhere?

Steve: Yeah.

Leo: Yeah. They probably were not. And the code is published, even though it's not open source.

Steve: Yeah, exactly. The code is all there for anyone to criticize. And the initial pass of the audit found nothing, although that wasn't a deep audit of the crypto. And that apparently is still underway.

Leo: I look forward to that.

Steve: Yeah.

Leo: That's actually probably what this security expert was thinking of. He'd heard dimly in the back of his mind, TrueCrypt hasn't been audited or is in the process of being audited, and he just kind of translated that into "is unreliable."

Steve: You're giving him too much credit, Leo.

Leo: I am, maybe. No, no definitely, I'm under NDA [crosstalk].

Steve: I know that you've read through security forums where everybody knows everything, and it's like, oh, goodness.

Leo: I have a friend who used to work with a friend who had a guy who his uncle I think was at the NSA.

Steve: That's right. And he told you that Windows is totally backdoored. That's right.

Leo: They can see everything you do, all the time.

Steve: That's right. Just give up now. But I'm a security consultant, so pay my bills.

Leo: Mark Goldstein in "North Virginia," and we know what that means, notes that HTTPS can be faster than HTTP. What? Test the thesis at this website: httpvshttps.com. It compares the load time of an unsecure HTTP and encrypted HTTPS versions of a webpage. Each test loads 360 unique, non-cached images for a 2.04MB total. And I guess, he doesn't say what the result is, that the HTTPS is faster. How could that be, Steve? Isn't there a lot of work being done?

Steve: Because - there is. And this is a neat site, so I wanted to put it on everybody's radar. You should try it, Leo. Everybody should try it: httpvshttps.com. And thank you, Mark, for pointing us to it. I wasn't aware of it. The secret - oh, here it comes.

Leo: It looks cool.

Steve: Yes, they did a nice job.

Leo: Done. Please try HTTPS. So that was 5.837 seconds. Now, this could be completely fake.

Steve: Nope, it's not. So now change it.

Leo: Whoa. Eighty percent faster with HTTPS. What?

Steve: Because of SPDY, Leo.

Leo: Ah.

Steve: So we talked about SPDY in the past. This is a very nice, legitimate comparison between not negotiating an SSL HTTPS connection and negotiating it, but using the SPDY protocol. What Google did, just to refresh our listeners, is they carefully looked at just all the extra cruft that is in the original HTTP spec. And remember that HTTPS is only a security tunnel through which HTTP runs. SPDY is essentially just an optimized and accelerated and carefully redesigned HTTP which is the protocol for HTTP/2.0. Right now we're at 1.1. The HTTP/2.0 spec incorporates these improvements.

So what's interesting is that here we see that the boost that we get from taking the very creaky original HTTP protocol and updating it by really optimizing it, that boost is substantially greater than the cost of negotiating even that many connections. Now, the test is a little bit - it tends to exaggerate the difference, specifically because the images are tiny, which means the handshake overhead is maximized. So for, like, much larger

objects that you are downloading, the query overhead, which is what SPDY speeds, the query overhead would be a much lower percentage of the overall than this page shows. But this is, I think, a very useful test. And so, again, I think everyone ought to give it a shot. It's fun. And IE, by the way, doesn't work at all because it doesn't support SPDY. So you have to have Firefox or Chrome or - are you using Safari?

Leo: That was Chrome.

Steve: Oh, okay, cool. I don't think Safari is a SPDY client, either.

Leo: Oh, that's interesting. All right. And they do suggest you run it in an incognito window over and over again. You do get different results each time. So that's kind of...

Steve: Yeah, you're going to. Well, because, I mean, it is how fast can it get all these little images. So little glips and blitches and glurps and things in the Internet...

Leo: Gliptches and mitches.

Steve: Those things.

Leo: To continue on, more questions for Steverino. Hey ho, Steverino. Let's see here. This is Question No. 7; right?

Steve: Seven.

Leo: Yup. All right. It comes from Kevin Garman in Illinois, and his chosen domain. Seems they had a slight problem: Hi, guys. Thanks for a great podcast and hard drive tool. He's talking about SpinRite. A heads-up to fellow listeners and a question: I was recently excited to add SSL support to my own personal OwnCloud server. OwnCloud is software that lets you do kind of your own cloud. So I was going through StartSSL's process to get a free cert when they sent me an email saying I'm not eligible for a free cert because banks and financial institutions are not allowed to use their free service. How does this apply to me? They said it's because my domain has the word "money" in it. Wow. Some check. Hey, says "money," must be a bank. To their credit, they were prompt at replying with an explanation, but I guess - they didn't change it. I guess I'm back to self-signed certs. Unless I can find another source of free SSL certs, I guess I'll have to wait for Let's Encrypt. Thanks again, Kevin. That's too bad.

Steve: Wasn't that weird? I just - that just sort of popped on. Kevin explained it, and it's like his domain is like mymoney.net or something, or something .money.net. And they just see that in the domain name, and that's what they key on. It's like, whoa, what? I mean, Bank of America doesn't have the word "money" in it. I guess it has the word "bank" in it.

Leo: I bet they look at "bank." I bet they would look for other words, too.

Steve: Yeah, wow.

Leo: It's funny they can't reverse that.

Steve: Yeah. And so for anyone who's going to get a domain for themselves, if you want free certs...

Leo: Don't put "money."

Steve: ...from StartSSL, avoid anything that sounds like a financial institution, and maybe you can get one. Wow.

Leo: That's interesting.

Steve: Isn't that, yeah.

Leo: And they didn't overturn it. I think they really don't want to...

Steve: Yeah, they just said no.

Leo: They don't want to be held liable for people losing money because of a cert.

Steve: Right.

Leo: So they're just staying away from it. Jeff in Baton Rouge, Louisiana shares some great AppLocker experience: I'm the IT Director for a major university athletic program, a Security Now! listener since 2012 - Yay, Jeff! - and was excited to hear you discussing AppLocker for malware protection in the podcast. We have been using it for years with great success and are trying to spread the word about how effective it can be in an enterprise environment. Pre-AppLocker, we were cleaning up three to five malware infections per week - per week. He didn't say how many, oh, yeah, he does say later how many seats, 400-plus seats. That's a lot of malware infections for 400 people - despite running a popular, up-to-date, enterprise AV program and having users operate with limited accounts. Wow. He has determined users. Whitelisting executables via AppLocker has resulted in us not having a single malware infection across 400-plus Windows machines in more than four years.

The prospect of whitelisting every executable a user could legitimately need to access sounds daunting, but actually it's pretty simple, at least in a corporate

environment. Rules for digitally signed executables are the easiest because you can trust all executables by a given publisher with a single rule. Want to allow everything that Google, Adobe, Citrix or Cisco offers? Okay, maybe not Adobe. Just create a publisher rule allowing anything signed by those guys, and you're done. Path rules are easy, too; but use them sparingly, and only on locations when users don't have write NTFS permissions. For example, allow c:\Windows and c:\Program Files, et cetera, but not c:\Users\Username) for executables. I have a set of 14 rules which allow 99 - I want to get these rules.

Steve: I know.

Leo: - 99% plus of the legitimate applications that our users need to run. I rarely have to revisit these rules or make exceptions. But when I do, it takes significantly less time than what I used to spend cleaning up malware. The users rarely even know that these rules exist, and it has blocked the execution of hundreds, thousands of executable malware droppers from infiltrating our Windows machines over the years. This is great. And he does provide a link.

Steve: Yup.

Leo: Here's my write-up on our specific implementation. It's at community.spiceworks.com. You could probably google "free almost perfect malware protection with GPO AppLocker" or something of the sort.

Steve: And the link is in the show notes, and the show notes are linked to the podcast now. So people can find them under Question No. 8.

[http://community.spiceworks.com/how_to/show/59664-free-almost- perfect-malware-protection-with-gpo-app-locker]

Leo: Good, good. All that said, AppLocker is really not suitable in its current form for users in a home or small business that doesn't have Active Directory implemented and requires an Enterprise license for the Windows machines in question. The only way I'm aware of manipulating rules is via Group Policy Objects. If MS was to implement some sort of more user-friendly GUI for home users and small business users, it could be a useful tool; but I'm not aware of any such option at the moment. What a great email. And I'm going to send this link right now to Russell.

Steve: Yeah. Yeah. So I loved this. We were talking recently - and this is what prompted Jeff, of course - we were talking about the notion in the context of Sony and how you lock down a big enterprise, the idea that maybe the only solution is going to be doing the same thing we've ended up doing with firewalls, where we've flipped the sense of a firewall from blocking the bad stuff to permitting the good stuff, and doing the same thing with applications, where we default disallow the OS to run something unless the app has been specifically whitelisted, and it's built into Windows as AppLocker.

So I really appreciated Jeff sharing his experience. And I did want to also plant a bug in

our listeners' ears, if they are aware of something, or something becomes aware that Jeff is referring to that allows for non-active directory class tweaking of AppLocker rules, then make sure that I find out about it so I can tell everyone because, as I said when I switched to Windows 7, I hope to, I plan to adapt a whitelisting approach from the beginning, and we'll see how it goes. So, Jeff, thanks so much for sharing that, and also for providing the link to your specific implementation.

Leo: Yeah, I just sent that along to Russell.

Steve: Neat.

Leo: Because I think that's how this conversation got started. We were talking about Sony, but about this AppLocker feature of Windows. I think it comes with Windows Ultimate as well as Enterprise.

Steve: I do, too, yes. So I think if you get - I remember looking it up and seeing, okay.

Leo: Not Pro, but Ultimate.

Steve: It is available, yeah.

Leo: And the idea of whitelisting is great. Of course, Russell was a little concerned, I mean, we have a perfect use for it, which is our editors' machines. There is a very limited set of applications they could or should be using on those machines.

Steve: Right. Right.

Leo: Basically Adobe Creative Cloud, and that's it. And so locking those machines down is just prudent. He's worried, though, and I've read people's stories saying, oh, yeah, but you turn on AppLocker, then caching doesn't work in your browser or whatever. I mean, I'm just making up stuff. But it's like address randomization. It breaks things in an unexpected way because Windows is really not designed to be doing it this way.

Steve: Well, or it's a little bit like when we were talking about it before, it's like NoScript. If you've got it turned on to, like, alert you...

Leo: It's annoying.

Steve: You're always saying, yes, yes, yes, yes, yes, you know.

Leo: But this is good. If he's got it working so well, clever, you know, giving it the domain or the certificate blanket authorization, things like that.

Steve: Yeah. Well, and so, for example, you could probably whitelist Adobe.com.

Leo: Right. We'd be done.

Steve: So signed executables, that are signed by Adobe, and bang. And so when something new is added, it's automatically permitted if it's also from Adobe.

Leo: Right.

Steve: So, yeah.

Leo: And really that just blocks the malware because the malware is not signed from Adobe.

Steve: Exactly.

Leo: Yeah. I think, yeah, this is good. Whitelisting is a great solution. If you can do it, it's a great way to get rid of spam. It's a great way to get rid of a lot of things. Pete Shaw in Warner Robins, somewhere[Georgia, USA], sounds like...

Steve: I didn't even know. He said "Warner Robins," and I thought, well, that sounds like Christopher Robin, but maybe it's his brother.

Leo: Probably Australia. That just feels Australian. I don't know what it is. He wondered - could be Arkansas, I don't know - wondered about Security Now! episodes: Steve, a big fan. Lately I've noticed episodes are not available even after a couple of days. What gives?

Steve: Okay. So I got a bunch of people. It was totally my screw-up. He sent this on the 18th, referring to the episode, presumably on the 16th, which I never posted. Elaine sent me a note when I was up with you, Leo, saying, hey, you know, just thought I'd mention that that never showed up. Then I thought that I hadn't updated the Security Now! main page, but all the little resources were there. No, I never even did them. So I'm embarrassed to say that, in the 10th year of the podcast, I'm still doing this manually, which sort of put me in mind of the cobbler's kids who run around barefoot, even though his customers all have shoes. I could have so easily, at any time, automated this process. But every week it's just like, well, okay, I've got other things to do, I'm just going to post this manually. And so I get the dates screwed up, I get the numbers screwed up. You know, it's like I'm human.

So anyway, I did go - we did go through, like, several weeks, mostly because I forgot, then I was out of town. Everything's caught up. Everything is synchronized. Everything is correct now. So anyway, that's what's happening, Peter, is it's just me. So when someone notices that something's missing, just send me a tweet. I'll probably see it, and I'll fix it. So apologies. But that's what happened.

Leo: A little self-serving, but you could also come to TWiT.tv/sn, where we also post all the audio and video.

Steve: Yes, go look - yeah, exactly.

Leo: You don't have to get it from Steve. We put it up, too.

Steve: Right.

Leo: Although sometimes, from time to time, things do take a while to get out or whatever, and we get the same kind of tweets. What's great is people don't want to miss an episode, and they want it when it's available. They want it right away.

Steve: Yup, yup.

Leo: So Warner Robins, thanks to the chat room, is an Air Force base in Georgia.

Steve: Ah, nice.

Leo: There you go. Robins AFB. So he obviously works in the Air Force. Pete Shaw - I'm sorry, that was Pete Shaw. This is Druce MacFarlane in Santa Cruz, California. Not Bruce.

Steve: Our final question.

Leo: Druce. I used to work with a Bruce MacFarlane. Wonder if he's related.

Steve: Well, those MacFarlane brothers, you know, their parents thought they'd have some fun with their first names.

Leo: And their sister Spruce. He has some perspective from the trenches: Steve and Leo, first of all I'd like to thank you both. I'm a longtime listener to Security Now!, and the things I have learned listening to this podcast have helped me advance my professional career. I'm just glad you guys aren't on commission. I was listening to your "Expensive Lessons" episode, where you expressed concern that Target and

maybe even Sony had been alerted to the attacks while they occurred, and did not take action on these alerts. In other news reports, but not on this podcast, I've heard this characterized as "gross negligence."

Steve: Well, and of course they're being sued, as we know. Target is - that suit is being allowed to proceed.

Leo: Right. The FireEye products they use monitor all incoming network traffic and look for objects that may contain malware or Advanced Persistent Threats (APTs). In a company the size of Target, it would be expected to see 20,000 alerts a month correlating to truly malicious objects being downloaded to end-user desktops. Whoa. 99.9% of all of these downloads will end up being harmless if the endpoint has an updated virus scanner or even a well-maintained and patched operating system. As it is, in practice, impossible to follow up on each and every one of these alerts, many companies simply ignore downloads and wait until the endpoint starts exhibiting behavior that indicates it's infected. Yeah. I suggest we wait and see if it swells up.

Steve: [Laughs]

Leo: Commonly, the PC starts sending command-and-control messages, and this is the point where organizations tend to take actions. Yeah, they go to their botnet. In fact, this is often the recommendation of the FireEye systems engineers themselves. While it's true FireEye may have provided early warning to Target or even Sony, suggesting that Target and Sony exhibited negligence in ignoring the FireEye alert is like claiming the townspeople were guilty of negligence after the little boy warned them about the wolf.

I'm going to mention that the stories I saw say that there had not merely been a download of malware, but in fact an active incursion into their systems; that they had a hacker inside the network, and they decided, eh, whatever. So I guess the question is what did they know, and when did they know it?

Security professionals currently suffer from a deluge - I can only imagine, though, this is good information - of what we are in the industry starting to call "trivial true positives." You used to call it "Internet background radiation." These are alerts that, while true, provide little relevance and only serve to tap the limited resources an organization has to spread across their entire information technology infrastructure.

As we know, Sony had five people in their security department, and three of them were administrative, were managers. So there were only two people in that whole company.

Steve: Yes, two techies.

Leo: As with all stories like this, it's always tempting to look for the easy answer, but the problem is far more nuanced than can be easily answered in a quick sound bite. That's why I applaud Steve for his statement that he would probably not have

been able to prevent a Sony-like attack. It helps bring perspective to the problem and recognizes the difficult job performed by all the security professionals that you count as listeners. Thank you, Druce.

Steve: Yeah.

Leo: Well, that's a very good point. I mean, it's extremely challenging. I hope we haven't in any way implied it's anything less than extremely difficult.

Steve: No. But I did love, you know, we talk about false positives. I love the term "trivial true positives." So they're not false positives. They actually are true. These are true problems. But they'll be knocked out before they can take root by AV or a well-maintained OS. But still, 20,000 of them coming in, the problem is the really nasty ones can get hidden in the noise.

Leo: I should also point out that, if you were using or could use AppLocker, you wouldn't have that many malware programs downloaded.

Steve: That's exactly right. I really think that AppLocker or a whitelisting solution, that's where we're going to have to go. I mean, think about it. Only allow things that you know are safe to run. Then, I mean, it's a sea change. And, yeah, it's not easy. It's like turning off scripting unless you explicitly know you want it on. So there is going to be a bit of a problem. But especially in a corporate environment, where they're not supposed to be running their own stuff anyway.

Leo: Well, that's the thing. But then your users bitch and moan because, "But I want to run Picasa Web. I have pictures to look at." Or whatever.

Steve: Yeah. During lunchtime.

Leo: Right. And you have to deal with management that says to you, no, don't worry about it. Or we're going to cut your budget. Or you have two active on-the-ground security professionals for a company of 10,000 people, something like that. I think a lot of companies are going to take this more seriously.

Steve: Can you imagine these poor guys? It's like, oh, my god, why do they even get up on Monday morning?

Leo: But they have, like, three managers. That's the worst thing.

Steve: Right.

Leo: There's only two guys doing the actual work, and there's three other guys just sitting around yelling at them.

Steve: Yeah, exactly, saying, "Why did you let this happen?"

Leo: Oh, god.

Steve: "My boss's boss's boss is all upset, so how did this happen?"

Leo: Believe me, we have total, total sympathy for anybody who's on the front lines in this. All I have to do is look at my web server and see how many attacks there are on SSH every day. Hundreds, thousands, it's constant. And to have even a tiny glimpse of what you must be dealing with. But I do think that better policies would make your job easier; right? Rather than saying, hey, we got some malwares downloaded 10,000 times this month, but I'm pretty sure the antivirus got it.

Steve: And so policies flow from the top down. And as you said, I think this has been an expensive lesson, but it's one the whole industry can learn from.

Leo: And by the way, I might mention that, if malware gets on your point-of-sale terminals, you probably should investigate that each and every time.

Steve: Yeah.

Leo: Just a tip. It's okay if it gets on the secretary's computer. But the POS terminals, that's maybe a little more important.

Steve: Yeah.

Leo: Steve Gibson, always fun to talk security with you. I learn so much. I love this show. You make us all look like we know what we're doing. So thank you. Security Now! is every Tuesday at 1:00 p.m. Pacific, 4:00 p.m. Eastern time, 2100 UTC. If you want to watch live, we'd love it if you do, but you don't have to because, and I want to reiterate, not only does Steve have a copy, but I have a copy. Steve's got his 16Kb audio versions. He's got the transcripts. And that is the only place you can get those transcripts.

We have transcripts on many of our other shows, and I got an email from the guy who does those, the company that does them, saying, you know, "You always mention that Steve's transcripts are written by a human. What are we, chopped liver?" So all the transcripts for all the shows are written by humans. I mean only to say that there is not, like, these aren't those machine-created transcriptions you might see on other podcast networks or websites. You've seen the horrible

transcriptions on YouTube. It's not that. These are good. For Steve's show and all of our shows.

Go to GRC.com also, not only to get the 16Kb version and the transcripts, but go there also to get SpinRite, the world's finest hard drive maintenance and recovery utility, a fabulous, fabulous tool that everybody who has a hard drive should have. He's got lots of free stuff. You can go there to read about SQRL, Vitamin D, and everything in between. You also can go there, I might add, to ask questions. So these feedbacks that we do every other show, some are generated through Twitter - he's @SGgrc - and some are generated on the website. None are generated through email. So don't email him. He doesn't even see it. He doesn't know where it goes. There's a black hole somewhere, it all falls in there. If you have a question, go to GRC.com/feedback. That's where you go. Fill out the form, and that's a good way to get a question or a thought or a suggestion to Steve. Or tweet him. Honestly, he reads all those tweets.

Steve: And next week, how does the Enigma machine encrypt?

Leo: Are you going to do - you've got to have some illustrations; right?

Steve: I do not need visuals.

Leo: You can talk us through it.

Steve: I can talk us through Enigma.

Leo: This was the amazing mechanical encryption machine that the Nazis used in World War II. And it's a very famous story. At Bletchley Park, that's where Alan Turing was, if you saw "The Imitation Game" you know the story, and he was able to, with some help, wasn't just him, crack the Enigma machine. The Germans didn't know it, and it was thanks to that that we were able - we - the Brits were able to turn around a very vicious submarine warfare...

Steve: Actually, one of the things that the movie does a really good job of - and will you have seen it by this time next week?

Leo: Yes.

Steve: Now that you have your fancy...

Leo: My DVD? It's not a Blu-ray. Don't get excited.

Steve: All right, all right.

Leo: It's just a DVD.

Steve: Anyway, one thing that they really did beautifully is understand that, having cracked it, they couldn't immediately act on the information because that would give away the fact that they had cracked it. So many times they had to, like, really do the tough call of, like, letting people die.

Leo: We know a ship is going to be attacked by a wolf pack in the North Atlantic, and we have to ignore it because, if we admitted it, they'd know we had Enigma.

Steve: And I did remember hearing the story, and I hope it wasn't apocryphal, that said, for example, that sometimes they would arrange to have a fishing boat, like, happen to be there, or a civilian plane would fly over, and the Germans would go, darn that plane, when in fact in order to create a coincidence that could then allow them as a means of having found out. But anyway, I loved the movie. So you and I will discuss what you think about the movie next week.

Leo: I'm sure I'll like it. I'm sure.

Steve: And that'll fit perfectly in with our description of how Enigma encrypts.

Leo: I can't wait. And remember, this is not digital. This is cogs and wheels and a crank.

Steve: And light bulbs, light bulbs.

Leo: Light bulbs. I mean, it is cool.

Steve: Yeah. The "W" lights up. Ooh. Write that down, Sherlock.

Leo: Speaking of Sherlock, you liked Benedict Cumberbatch in the movie? He was good, wasn't he.

Steve: He really was good.

Leo: I think he's up for some awards, yeah.

Steve: I wouldn't be - no, it is a good movie, although he did really play up the whole prima donna thing that I found a little over the top. But, you know...

Leo: He didn't choose to act as if he had Asperger's, though; right?

Steve: No. No, no, no.

Leo: Because some people thought that Turing was on the spectrum, and I was worried that he might - see, this is what happens with, you know, a lot of these movies, math whizzes are looked at as, like, freaks. Prima donna, hell, I'm a prima donna. That's nothing.

Steve: I think you'll like it.

Leo: I can't wait. I have to watch it before Sunday. That's the Golden Globes.

Steve: Good, do.

Leo: I like to see all the movies before the awards.

Steve: And we'll talk about it on Tuesday.

Leo: Good. Thank you, Steve.

Steve: Along with how the machine works.

Leo: I can't wait. That's going to be fun. Enigma next week.

Steve: Bye, Leo.

Leo: Bye, Steve.