Listener Feedback #224

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-536.mp3
Quarter size (16 kbps) mp3 audio file URL: http://media.GRC.com/sn/sn-536-lq.mp3

SHOW TEASE: It's time for Security Now!. Steve Gibson is here. We've got, of course, security news. We'll get you up to date on all of that. And then we're going to answer questions from our audience, some really great ones, and of course Steve's great answers, coming up next on Security Now!.

Leo Laporte: This is Security Now! with Steve Gibson, Episode 536, recorded Tuesday, December 1st, 2015: Your questions, Steve's answers, #224.

It's time for Security Now!, the show that protects you and your loved ones online with this man here. He is the Explainer in Chief, Mr. Steven "Tiberius" Gibson, from his Fortress of Solitude in beautiful Irvine, California. Hello, Steve.

Steve Gibson: Hey, Leo. So we're going to give another try for a Q&A. Last week there was just so much to talk about, and we spent so much time, that we ran out, and we only got two out of the targeted 10 questions. So I've got the eight that we didn't get to, and I found a couple more to round it out to 10. So this will also be a Q&A #224. There were a couple big stories, interestingly, I think deep technical, that I was initially thinking, well, maybe we can, like, squeeze it in. But I thought that hasn't worked out so well for us recently. So they will probably be the topics for next week's, and perhaps the week after, podcast, rather than trying to get too much news in. And we didn't have, you know, it hasn't been a crazy week.

I did want to briefly follow up on my controversial comments, which drew a huge amount of friendly fire from our listeners, about law enforcement and iOS unlocking under warrants. I found an interesting note about Mozilla's life after Google. Microsoft has responded to Dell's epic mistake with the self-signed certificate, including the private key. Turns out that Arris cable modems are in the doghouse for...

Leo: Oh, no.

Steve: ...a bad reason. BlackBerry has said no to a major government. Sixty-seven - actually the press is covering it as 66, but one just got filed - 67 companies are being sued for their use of HTTPS TLS Elliptic Curve Crypto by, guess what, somebody in East Texas. And we know what that means.

Leo: Uh-oh.

Steve: Then we have another welcome nail in the Adobe Flash coffin, which actually I cribbed that phrase, the nail in the coffin, from TechCrunch, that noted this story. A little bit of miscellany. And 10 questions and answers from, well, questions from our listeners. Answers...

Leo: Answers from you.

Steve: ...from me, yeah.

Leo: Sometimes listeners have answers.

Steve: So a great podcast.

Leo: Well, we're excited. And...

Steve: Pursuant to this, I will do this a little bit out of order because I had this down in miscellany. But I got a tweet from clearly a listener of the podcast this morning, Bill Griffith, who said @SGgrc, he said, "'Why Zebras Don't Get Ulcers' is on sale on Audible for 4.95."

Leo: Oh. I already bought it, yeah.

Steve: And he said, "I got it. Excellent call. Thanks again for your recommendations."

Leo: This is the one that's about cortisol and stress and, yeah, it's great.

Steve: Well, yeah. It's one of the things that I've just sort of, in my interest in health and longevity, I've recently been sort of looking at the whole issue of stress in people, in society, and so forth. So this is not written by, as some books like this are, by a so-called "science writer." This is written by Robert Sapolsky. He's got a Wikipedia page which begins: "Robert Morris Sapolsky (born 1957) is an American neuroendocrinologist, professor of biology, neuroscience, and neurosurgery at Stanford University, researcher

and author."

Leo: So don't let the name fool you, is what you're saying.

Steve: Yes. Well, and also don't let the title fool you. This was recommended by someone who, it's like, okay, well, thank you for the recommendation. You know, "Why Zebras Don't Get Ulcers," you sort of think, okay, well, is it illustrated? Are there going to be pictures of zebras sitting under a tree, gazing off into the distance? No. This is serious, wonderful, but also kind of humorous, but still very hard science about the nature of stress, and the fact that the same sort of causes of traditional survival stress are being triggered by worries about credit card payments and mortgages and teenage daughters and the things that stress people out these days, and that the problem is that this sort of environmental stress isn't acute, it's chronic. And our body wasn't equipped to handle continuous low-level sort of background chronic stress. And all kinds of things go wrong. And there's an argument to be made that stress in our lives is one of the major deleterious factors in long-term health.

And anyway, for what it's worth, I completely agree with Bill. This is a great book. And so it is available everywhere, and also on sale right now, or at least when he sent the tweet. I don't know how long it took him to read it or listen to it. But at the time it was 4.95. So don't use that for your free one, use...

Leo: No, yeah, exactly, yeah.

Steve: Save this one for one you want to get on sale. But I recommend it as Bill did.

Leo: Audible does a lot of sales. I mean, we're no longer in the ad, but they do a lot of sales, and they do a lot of giveaways. I notice they just have, for free if you're a member, which means if you signed up you could get this for free, "A Brief History of Holiday Music." This is - I've taken the Great Courses that Robert Greenberg does on hit music. It's the best thing ever. I mean, if you want to understand music, it's incredible. He's got a little single chapter on holiday music, and I think I'm going to download this one because he's great. It adds to your appreciation to understand the history and the back story to what's going on.

Steve: Oh, which is the case for, like, even classical music. If you understand...

Leo: Well, yes, exactly.

Steve: ...the environment in which it was written, why it was created, what the point was, it makes it so much more easy to appreciate.

Leo: That's his classic course, how to listen to and understand great music. It's all about classical, the origins of classical music. And I've found so much great stuff in it. Yeah, it's wonderful. I've listened to this, and it's fantastic.

Steve: So our Picture of the Week on the front page of our show notes is Mozilla saying "We don't need Google's money anymore." Which, to me, comes as a huge relief because I still need Firefox. And I really believe that the industry needs Firefox in the way that it needs Chrome because Microsoft will have its browsers, Apple will have Safari, but these are our interfaces to the Internet. And I see a role for Firefox. And so I was a little worried, I remember, toward the end of last year when the announcement came, toward the end of 2014, that they were getting off of the Google gravy train. Essentially, Google had been financing the Firefox browser.

So the good news is Mozilla has just posted their 2014 numbers. And they had revenue of $330 million, and that's up from 2013's revenue of $314 million. So they're not in trouble. This doesn't mean, oh, darn, Firefox is going to wither and die; which, again, to me, I'm breathing a sigh of relief. And I know that a chunk of our listeners are diehard Firefox users, as I am. What, of course, Mozilla has done is they're getting a lot of revenue from the default settings of the search engine in the browser, and they've moved from a single Google global strategy into a regional search engine relationship strategy, where they now have by default Yahoo! as the search engine in the United States, Baidu in China, and Yandex in Russia. At the moment, Europe still uses Google by default, though not under a paid relationship with Google. So anyway, I saw this. It kind of flashed by, and I thought, oh, yay. Good. Because I want Firefox to endure.

And I did want to mention, I wanted just to acknowledge everyone's outrage at my suggesting last week that Apple being able to arrange to unlock locked iOS devices under court order seemed like a reasonable thing. I mean, I'm not changing my position. But I wanted everyone to know I heard them. I listened. I read all the outrage. Some people wrote blogs about it, specifically about my comment. And I read them to see if there was any point that I had missed.

And it's not that I'm saying I don't want the iOS devices to absolutely remain as secure as they are today. It's that I'm watching Washington, and I know how politics goes, and I breathed a huge sigh of relief over the Obama administration's statement a month or so ago that they weren't going to push for sweeping cryptographic effort. But then we've got, post-Paris attacks, Congress is rumbling again. And to me, if we had to give something, this seems like something in keeping with precedent. And again, we could be absolutist, but I think maybe some compromise will be warranted. Again, I'm not saying I want it. But I did want to acknowledge everybody who just, like, fell out of their chairs, as I knew some of our listeners would.

Leo: Incidentally, this just in. Mark Zuckerberg and Priscilla Chan, his wife, had their baby just a few hours ago.

Steve: So now he's off for two months.

Leo: He's off for two months. Their baby daughter Max came into the world with a post on Facebook on Tuesday.

Steve: So Maxine is maybe the full name?

Leo: I don't know. It just says Max here.

Steve: That's interesting.

Leo: Of course it's the Daily Mail. I should probably read the Facebook post and see.

Steve: Well, congratulations to him and them.

Leo: Yeah. Yeah.

Steve: So Microsoft has responded to the news we covered last week of Dell's dangerous certificates. And of course Dell themselves, sort of semi-reluctantly dragging their feet, oh, this is not malware, this is not the same, blah blah blah, it's like, okay, fine. Anyway, Microsoft has a technology called the CTL, the Certificate Trust List, which has been present by default from Windows 8 onward, and which can be installed back at Vista, and from Vista on. So Vista and Windows 7, and also on the synchronized server platforms. They just posted yesterday this notice.

And in fact, because for Vista and 7 it's not by default, I also made it the bit.ly link of the week. Bit.ly/sn-536 will take you to this Microsoft page, where Microsoft just wrote yesterday: "Microsoft is aware of unconstrained digital certificates from Dell Inc. for which the private keys were inadvertently disclosed. One of these unconstrained certificates could be used to issue other certificates, impersonate other domains, or sign code. In addition, these certificates could be used to spoof content, perform phishing attacks, or perform man-in-the-middle attacks against Dell customers. This issue affects all supported releases of Microsoft Windows. Microsoft is not currently aware of attacks related to this issue.

"To help protect customers from potentially fraudulent use of these unconstrained digital certificates, the certificates have been deemed no longer valid by Dell Inc., and Microsoft is updating the Certificate Trust List (CTL) for all supported releases of Microsoft Windows to remove the trust of these certificates.

"An automatic updater of certificate trust lists is included in supported editions of Windows 8" - and then they go on from there, 8.1, RT, RT 8.1, Windows Server 2012, 2012 R2, Windows 10 and so forth. And they even mention Windows 10 v1511, which we know was just recently. And they said, oh, and for devices running Windows Phone 8 and on. Then they make it clear for systems running Windows Vista, Windows 7, Server 2008, Server 2008 R2 that are using the automatic updater for certificate trust lists, and then they have a link to a Knowledge Base article, customers do not need to take any action as these systems will be automatically protected.

But that was in addition, we talked about it at the time that it was released, some time ago. If users are not sure, it certainly makes sense, if you're still using Vista and 7, to add that to your system so that this and other future certificates can get yanked, can be essentially marked as untrusted preemptively by Microsoft in order to deal with this kind of problem. So it's a good thing that Microsoft did this. I noticed they are clearly happy with their relationship with Dell and worded it accordingly. So they didn't want to upset Dell.

Leo: Incidentally, I finished the long letter that Mark Zuckerberg wrote to his new

daughter, newborn daughter Max. At the end of it, kind of a stunner. He says, "We pledge we're going to give away 99% of our net worth in our lifetime for public good, including health and education initiatives." That's $45 billion.

Steve: Today.

Leo: Today. Yeah, who knows what it'll be over that length of time. So quite an astounding letter to his daughter that he posted on Facebook. It's very long, so it took me a while to get through it.

Steve: Well, and if he holds back that last 1%, that's, what, $500 million.

Leo: No, yeah, that's plenty.

Steve: That's still plenty.

Leo: Yeah. And Bill Gates did the same thing, by the way, pledged he would give away all but 1% of his value.

Steve: Yup.

Leo: And I think that that, boy, they're setting a very good precedent and standard for these tech billionaires.

Steve: And there's a group of people who are all doing this; right?

Leo: Yeah, well, Bill Gates has created a group to try to lobby other wealthy tech folks to do this. We will give 99...

Steve: He and Warren Buffett and so forth?

Leo: Yup, yup, with mixed success, oddly. "We will give 99% of our Facebook shares, currently about 45 billion, during our lives to advance this mission of personalized learning, curing disease, connecting people, and building strong communities." That's just really great. Amazing.

Steve: That's neat.

Leo: Yeah.

Steve: So, okay. Arris cable modems in the doghouse.

Leo: Yow. Because, you know, I went out and bought an Arris to replace the modem that I was using, my Comcast-supplied modem. Motorola bought Arris not so long ago.

Steve: Correct. And Motorola has had a good name and label. In fact, that's what I was using until my friend at Cox said, oh, no, no, no, you need this Netgear CM600. That's the one that I mentioned a couple weeks ago that has now - it's got so many bands, I'm measuring 300Mb down and 30Mb up. And it's like, okay, thank you, that's a hundred times what I had with my two T1s. So welcome Gibson to the 21st Century.

Okay. So here's the story. It has been known since 2009 that there is a type of backdoor in Arris modems. It's a daily changing password based on a seed which is in the firmware such that the firmware algorithm generates the password of the day. And so it is possible, if you know the seed and what day it is, to potentially access these cable modems remotely. The various news coverage is calling this a "double backdoor" because there's a second password you need. But it turns out that is simply and always the lower five digits of the device's serial number, which can be obtained without knowing it. So once you know that, and if you happen to know the seed used in the algorithm to generate the daily password, you've got a problem. Turns out most of the manufacturers never change the seed.

Leo: Of course not.

Steve: They leave it set to the default.

Leo: Of course they do. That'd be too much work.

Steve: Of course they do.

Leo: This is getting to be an old story, isn't it.

Steve: It really is. All capitals, write this down, M-P-S-J-K-M-D-H-A-I. That's the seed. That's what all the modems use. And so all the bad guys know it. So the history here is that recently Bernardo Rodrigues, who is a vulnerability tester with Brazil's Globo TV network, he was looking at the firmware in some of his company's cable modems which they were going to be providing. And he found an undocumented library, "libarris_password.so." It provides this backdoor which allows, given that you had this information, privileged remote login, given that you know what day it is.

So he then looked at three different models, the TG862A, the TG860A, and the DG860A. All three of those have the firmware, have this vulnerability, and using the Shodan search engine that we've talked about often, Shodan is this unnervingly powerful, it's like Google for Internet of Things, or Google for everything on the 'Net where it's indexing, not web pages and websites the way Google does, it's indexing anything that responds to packets anywhere on the Internet. It has found more than 600,000 of these devices

publicly - well, of course a cable modem is public. It's publicly facing.

So whereas their IP address may change, that is, so Shodan may not be good for absolutely nailing for sure a device by IP because, as we know, ISPs can and periodically do, or at least may, change the IP address on specific physical modems at customer premises, still there's way over half a million of these. And you can imagine, now that this news is public, that the bad guys are going to start having a field day.

So this guy did contact Arris, and they asked him not to reveal details about the modem's password generation algorithm. And he didn't. But he doesn't need to because this is the problem with these sorts of things is it relies on secrets. And we know, well, it relies on a secret which is unfortunately identical, unless changed, for every single modem. Now, it's worth noting that Comcast told DSL Reports that they don't use the default. So until everyone finds out what Comcast uses, Comcast subscribers will be safe. But as soon as someone looks in their firmware and figures out what Comcast is using, then again we have a problem.

So the lesson here is that it just - there's no way to offer something like this with anything that is universally defaulted. I would argue that the biggest problem is that these backdoors are in place and may well not even be used. No one may be using them. So here's like a feature, a bullet point, which was made available for the cable providers that allow them to access for who knows why. Maybe that's the way they update their firmware. You can also - so this does create, Bernardo wrote, a full busybox Telnet/SSH shell session that allows you to do anything you want to on this.

So these little guys are running some version of Linux, or a small Unix or something, and this lets anybody get in. So the only way to do this safely, if this is what you want to do, is the random number per device, where, for example, when a cable company first pairs themselves up with the cable modem, they generate a random number, they stick it in the cable box, and they make a note of it in the associated file. And of course now the problem is, if you lose or forget that random number, you'd have to tell the customer hold the reset button down to do a factory restore or something like that. But it just - there isn't a way to equip 600,000 devices, no matter how clever you are - you could say clearly they tried to be clever. They created a seed that generates the sequence. The sequence creates a changing every day password. Then they also, beyond that, they also use the lower five digits of the serial number.

So, okay. The problem is, once that becomes out, once that information escapes, now everybody has access to 600,000 cable modems and can get up to all kind of mischief. So the problem is this is the kind of secret that is never kept. It cannot be preserved. So the only way to do this is, for example, maybe have it disabled by default. If the number is blank, then the backdoor is not even open. Of if a cable provider wants to do it, then generate a random number, rather than a clever algorithm, because then at least every single one of these 600,000 cable modems would have a large, absolutely unknown, random number that is associated with nothing else, not with the modem, with the vendor, with nothing, completely random.

And it would be as good as any of our crypto is in terms of protecting that, with the obligation that it somehow needs to be, you know, that random number cannot be algorithmically generated, and in fact the cable modem company cannot have a secret algorithm that generates it because, once again, when that gets loose, that one algorithm, then all the devices that were keyed by that algorithm are vulnerable. It's got to be just pull a number out of the air. It's the only way to do it securely. And so, again, who knows? This is a great lesson about security and how to do it wrong and why even being this clever didn't work. You can't be. If you're going to have the security, you have

to sacrifice the convenience. But then you get real security.

BlackBerry says no to Pakistan.

Leo: Yay.

Steve: Yes. Pakistan, I think it was earlier this year, it was in July, said to BlackBerry, we give you till the end of November to give us full, "unfettered" was the word they used, access to BlackBerry's servers. BlackBerry said no then. The deadline has been extended by one month. Pakistan said, oh, maybe you didn't hear us correctly, so we'll give you through December 30th. Then we're not kidding. BlackBerry still says no. So they had said previously they would pull out of Pakistan rather than comply with a demand for full access to content on the BlackBerry enterprise service. The company says the Pakistani government wants the ability to monitor all traffic in the country, including every BES email and BES BlackBerry Messenger message. BES communications, BlackBerry says, are routed through the company's servers in Canada.

BlackBerry did say they'd be willing to work with Pakistani authorities to protect public safety, but that the privacy of its customers is paramount and something on which it will not compromise. Marty Beard is BlackBerry's chief of operations, and said that the company recognizes the need to cooperate with lawful government investigations of criminal activity, but they have never permitted wholesale access to BlackBerry servers. And we'll remember, because we talked about this five years ago, that back in 2010, back when BlackBerry was still relevant, when I was carrying a BlackBerry and loving the little physical keyboard, both Saudi Arabia and the UAE were threatening to ban BlackBerry if they didn't provide essentially the same technology after various terrorist activities that they felt would give them the visibility they need. BlackBerry never complied, but did arrange a compromise more along the lines of, on a case-by-case basis, we'll consider individual requests.

So it'll be interesting to track this. I mean, this is a fascinating aspect of the times that we're in, as we know, with this fundamental tension between governmental need to have access to communications and the prevalent, trivial it has become, capability of technology to make that impossible. So, okay, now, Leo...

Leo: I love it when you - I know something's coming when you do that. All right. I'm ready.

Steve: Yeah. Patent troll. When you hear the Eastern District of Texas...

Leo: Always.

Steve: Yeah, mixed in with patents, we know we're in trouble. We've discussed this before. There's some strange court in the Eastern District of Texas that just thinks that anybody with a patent is golden. Doesn't matter anything else about it. You nasty infringers, what do you think you're doing, you're robbing these companies of their livelihood. And in fact remember that Samsung has been the victim of this court, so much so that Samsung began sponsoring the sports events, like built them a stadium and had "Samsung" on the Lexan panels around the fencing, in order to say, look, we're

good guys. Please stop picking on us.

Okay. This is really interesting. This is going to be another interesting thing to track because this company, CryptoPeak, has sued 66, says the press, but I checked, and it's now 67 because there's one that was filed on the 25th. So this is a patent which was originally filed on May 28th of '97, so quite a while ago. And as patents go, it took a few years. Typically the patent, you send it off, and then you wait a year. Then they come back and they say, yeah, we're denying this for the following reasons. Then you and your patent attorneys get together, and you explain basically to the Patent and Trademark Office why their interpretation of what you said isn't correct, or why the prior art samples that they have shown don't apply to your invention. So you go back and forth, back and forth, back and forth a few times. And as happened on March 13th, 2001, okay, so like four years later, almost four years later, this patent was granted.

Now, patents have a life of 17 years currently. So that means, if this was granted on March 13th, 2001, it will expire on March 13th, 2018. So here we are about to start 2016, so a little more than two years, what, 28 months, something like that, 27 months. So these guys, apparently earlier this year, Adam and Marcel Yung, Y-U-N-G, apparently sold this - we don't know what the terms and conditions were - to this company CryptoPeak that has no presence on the Internet, is not actually apparently using the patent that they purchased which has 27 months left of life on it before it expires. Instead, they've decided, starting in July of this year, so this summer, early summer, to start suing people who they believe infringed this patent. Now, what's controversial about this is that, by their reading of the patent they purchased, the use of elliptic curve cryptography is a breach of this patent.

Leo: No.

Steve: So they've sued Sony, Macy's, AT&T, Pinterest, Netflix, Yahoo!, Hyatt Hotels, Priceline, Best Western, Expedia, GoPro of all people, Progressive Insurance, and about 50 others. And in fact, Leo, there's a link here in the show notes, the search.rpxcorp.com. If you click that, it'll bring up a list of what was 66, is now 67, companies. And they're all major large companies that are doing secure transactions on the Internet. I mean, like, all these companies. Sony needs to have security, Macy's, AT&T and so forth. I mean, so it's every...

Leo: A lot of retailers, too, though, like Barnes & Noble and Bed Bath & Beyond, I mean, that's kind of an interesting - Kohl's, Groupon, Shutterfly, Netflix, Etsy. Holy cow.

Steve: Yeah, I mean, and there's no particular logic to it.

Leo: No. Macy's, VUDU, I mean, it's weird.

Steve: Yeah, well, although you'll notice, like, Priceline, Best Western, Expedia, sort of like they say, oh, let's go after the online travel companies. And so they just - and they probably go over to SSL Labs and see whether those companies are using elliptic curve crypto; and, if so, they rubberstamp the complaint that they've already synthesized. And in fact, if you look at the dates, you can see they're in batches. There's like a whole

bunch in one batch, and then they...

Leo: Yeah, it's very easy and cheap to do this, a few hundred dollars to file these. By the way, notice the number of closed suits. That means they've settled. They didn't go to trial. They just gave them money.

Steve: I know.

Leo: And that bankrolls the rest of the suits.

Steve: I know.

Leo: Yeah, wow, 67 defendants.

Steve: So the complaint says, I mean, it even says this: "Upon information and belief, Defendant has infringed and continues to directly infringe one or more claims of the" - and it's called the "'150 Patent." In patent litigation they typically just refer to the long number, which is now seven digits, by its last three digits, so it ends in 150 - "the '150 patent, including at least claim 1, by actions comprising making, having made, and/or using one or more websites that operate in compliance with the standards of Elliptic Curve Cryptography (ECC) Cipher Suites for the Transport Layer Security (TLS) protocol," which now in parens says (the "Accused Instrumentalities"). "A representative example of a website of Defendant that operates in compliance with this standard is" - and in this case it was livevol.com - "is secure.livevol.com."

So anyway, this is much in the news. I got a bunch of tweets about it yesterday and more today because it's beginning to be picked up. Of course The Register jumped on it immediately with their particular style. And then ZDNet picked up on it. And hopefully it's the kind of thing that the EFF will decide they want to deal with. I mean, this is - oh, and I should mention that Netflix has filed a motion to dismiss, alleging that this is - and in the show notes in a copy of their motion, if anyone's interested. So the problem, well, of course, there are many problems with our current patent system. But one is that anybody can be sued. And even defending yourself against a specious lawsuit is not inexpensive, as, Leo, you have found out because there was the patent troll who was going after podcasters for a while; right?

Leo: Thank god he's gone, yeah.

Steve: Yeah. But one of the interesting things about patent law is that it's not the designer of the invention, I'm sorry, it's not, well, it's the person using the invention, even if, for example, none of these companies designed this, or wrote it, or recreated it. Microsoft wrote the crypto suite in Windows, and an open source community wrote the crypto in OpenSSL. Yet the users of that are, the way patent law is, responsible for their use of a technology which may be subject to patent. So hopefully all these companies, and any others that are named, will band together, pool their resources, and arrange to get this resolved in a way that minimizes the expense that any of them have to go through because you get sued, you have to defend yourself.

Leo: Yeah, or settle.

Steve: Yeah. And also...

Leo: Of course they always try to price it so it's cheaper to settle than defend yourself.

Steve: Yes. I didn't spend enough time with this to, like, parse this. And, boy, it is impenetrable legalese. I mean, you would need Denise to go into this and figure out what it means. But it looks to me like it doesn't in fact bear on the technology. But unfortunately, that doesn't matter because, exactly as you said, Leo, it is cheaper to write these people a check and have the lawsuit go away, as some of the smaller companies appear to have done, than it is to argue in court because, boy, you're talking tens of thousands of dollars immediately before you even stand up and say "Your Honor." And it's a shame that that's the way the system is set up and that companies like this are allowed to operate as they are.

So as TechCrunch put it, Flash gets another nail in its coffin. Adobe just announced today, or maybe it was yesterday, that they are renaming what has always been called Adobe Flash Professional to Adobe Animate CC. And I guess "CC" must be, what, Content Creator? I'm just guessing.

Leo: Oh, no, Creative Cloud.

Steve: Ah, Creative Cloud. I did see that terminology there, too.

Leo: That's their phrase for their subscription service basically.

Steve: Okay. So what they said was, and they were clearly a little self-conscious about this, they said: "For nearly two decades, Flash Professional has been the standard for producing rich animations on the web. Because of the emergence of HTML5 and demand for animations that leverage web standards, we completely rewrote the tool over the past few years to incorporate native HTML5 Canvas and WebGL support. To more accurately represent its position as the premier animation tool for the web and beyond, Flash Professional will be renamed ?Adobe Animate CC?, starting with the next release in early 2016."

They said: "Today, over a third of all content created in Flash Professional uses HTML5, reaching over one billion devices worldwide. It has also been recognized as an? ?HTML5 ad solution? that complies with the latest Interactive Advertising Bureau" - that's the IAB, as we know - "standards, and is widely used in the cartoon industry by powerhouse studios like? ?Nickelodeon? and Titmouse Inc?. Animate CC will continue supporting Flash (SWF) and AIR formats as first-class citizens. In addition, it can output animations to virtually any format, including scalable vector graphics, through its extensible architecture."

So this is just good news. This is Adobe saying, well, I mean, for example, as we know,

Flash won't run on any iOS devices. And Leo, I heard you mention in your previous podcast, MacBreak Weekly, the percentage of Black Friday sales that went through iOS.

Leo: Yeah, it was like 80 something; right?

Steve: Yeah.

Leo: Eighty-three percent, 84 percent, amazing.

Steve: Right. So although we note that iOS instances are dwarfed by Android, in terms of dollar volume transactions happening on them, iOS dwarfs Android. And so the fact that iOS won't run Flash means that Flash-based ads won't work on the mobile platform, whereas HTML5 can. So I think it's clear that it's going to take it a long time to go away, it always takes these things forever to finally disappear, but they're probably going to.

And I found a nice note that lets me explain something that I wanted to about SpinRite, from just two days ago. Cornel DeLorean, who is in my hometown, San Mateo, California, his note, as I'm going through the mailbag for the Q&A, said "SpinRite saves the day again." He wrote: "Hi, Steve. I'm a longtime listener and owner of SpinRite. I purchased my copy back in 2009 to support you and the podcast; but I also figured at some point I would get a chance to rescue a disk, and after all these years that time came. I was helping a friend with his laptop. It was running slow and hanging. So after poking around for a bit and removing some unneeded and expired software from McAfee and so forth, I rebooted, and it refused to boot up. I tried every trick I learned from doing many years of desktop support, but nothing worked. So I thought, let's see if SpinRite can save the day.

"I started the scan on Level 2, and it happily got to 48% after about 40 minutes, but then it slowed way down. It ran overnight, and by morning it was at 51%. When I got home from work at the end of the day, it had crawled up to 53. I let it run overnight again and it was finally finished when I checked it in the morning. I rebooted it with fingers crossed; and, sure enough, it booted right up. I backed up all of his data and advised him it was time for a new laptop." And then he says: "But it ran fine for a number of months until he did get a replacement. Thanks for such a great product and podcast."

So I just wanted to take this - this is like a perfect case history of SpinRite zipping along to halfway, to nearly halfway, 48%, where it encountered trouble. And then it sat there and just essentially refused to take no for an answer from the drive. And it was clearly slow for a while, from like 48% to somewhere past half, like 53. So had it never hit trouble, it would have zipped all the way through, and maybe even fixed some things that were less recalcitrant. In this case, it had to really struggle. But this was a case of all or nothing for this person. And so he let it go, it fixed the problem, the drive worked fine, and he eventually replaced the laptop.

So the problem that SpinRite has is that it actually has to read all of the sectors on the drive in order to check them all. And drives have gotten so big today, nothing does that anymore. Not even formatting. Formatting doesn't read the drive. It used to, back when it could. But Microsoft said, wait, we can't take a week to format a drive any longer. So we're just going to do the so-called "quick format." Well, what the quick format does is just lay down the architecture of the format. It lays down the root directory, and the

various tables at the front of the drive, and bitmaps showing all sectors unallocated. And then it says, okay, it's formatted. That's the quick format.

So the problem is drives, as we know, are becoming ridiculously large. The good news is, in the work I did on the next release of SpinRite, which will be 6.1, the first thing I did was look at speed and performance and made through some really fun trickery a huge increase in performance. It clocked at two hours per terabyte, that is to say, half a terabyte per hour. So that makes it feasible to run SpinRite on a 4TB drive overnight, in eight hours, which makes it supremely practical.

And as I've said before, we're making good progress with SQRL. I'm finishing the rewrite of the semantics page, which actually required a rewrite because we had changed so many things down here toward the ends, simplifying it, tightening it, just really nailing it. And as soon as we get SpinRite out the door, I mean released, then I'm back to - did I say SpinRite? I meant SQRL, sorry. Soon as I get SQRL out the door, then I'm back to working on 6.1 and getting it into everyone's hands, as I have said, for a free upgrade that will give us this performance boost. So I thank everyone for their patience. And in the meantime, it's still rescuing drives every day.

Leo: Very nice. I have questions. Are you ready? Let me open the magic PDF, and we'll get your Q&A fired up. We start in Hamburg, Germany. Michael Walther notes assembly language is climbing the charts. The Top 10 of programming languages. This is TIOBE software, the TIOBE index.

Steve: Yeah, check out that link. I know that you're a fan of languages, Leo.

Leo: Oh, I always look at this. I look at this all the time because I find it fascinating.

Steve: Yes.

Leo: And I guess the way they do this is courses, third-party vendors, engineers worldwide, search engines, they have a methodology that I think is pretty good. We're not talking about best programming language. We're talking about literally lines of code that have been written. And of course Java is number one still.

Steve: So, yeah, what's interesting is Java is number one. And I wonder if you think - I know that it's used in corporations because it is still a multiplatform language. You put the Java Runtime on a system, and then you're able to host your applications anywhere. But I'm wondering also if it might be Android.

Leo: Yeah.

Steve: Which is, you know, Android apps [crosstalk].

Leo: It's the language of choice for Android, so.

Steve: Yeah.

Leo: You can use other languages, but Java's the main one. What's interesting is C is still number two.

Steve: Yeah, isn't that interesting. And in fact if you scroll past the bottom of that chart, there's an interesting graphic that shows the evolution of language popularity over time.

Leo: Java's been going down, then went up. It's really interesting. This climb here from - it had been declining. This sudden climb I really think has to be Android.

Steve: I think it has to be.

Leo: It has to be Android; right?

Steve: Yeah.

Leo: Because it's been steady for, you know, since 2001 it's been...

Steve: C is the language. I mean, that is because it's flexible enough, multiplatform, yet it's also highly transportable when you wrap it with the right libraries.

Leo: Yeah. And then next is C++.

Steve: Yeah, really, I thought that was really - and to Michael's point, assembly - and I don't quite understand why it's there. But it's like number 11.

Leo: Number 11.

Steve: The bottom of the Top 10 is Perl, which is slipping a little bit. Ruby on its way up. VB.net is there at number eight, JavaScript just above it at number seven. But assembly language is 11. Now, of course, that doesn't say which chip assembly.

Leo: Right.

Steve: And I'm wondering if it might be, for example, like ARM assembler for all, you know, the Internet of Things, sometimes...

Leo: Yes, that's what it is.

Steve: ...you're running very memory-constrained environments where you can't afford Linux, and you can't afford the inefficiency of any compilation. All compilers are generating code much bigger than somebody who writes in assembler. So I'm thinking that, for example, the little button that Amazon sells, well, it makes sense to them to have the code written in assembly because then they can dramatically shrink the chip by using much smaller ROM. And that absolutely maps to cost. So in high volume, highly cost-sensitive areas, assembly language is still what people are going to use because there's no way to get more efficient. And I bet that's driving the interest, or at least, even though it's not in the Top 10, it's number 11.

Leo: That's amazing, really.

Steve: Yeah.

Leo: Although, if you look at number 12, Delphi, you realize, well, mm-hmm. Then Visual Basic. Objective-C, which is Apple's language for iOS, surprisingly, number 14 and plummeting. It was number three last year.

Steve: How could COBOL be number 20? And coming up?

Leo: Up-and-coming, it's an up-and-comer, COBOL. R, plummeting. PL? Pascal, MATLAB, Swift is coming up. Interesting, though, the plummet in Objective-C. I guess that's the growth of Swift. It's directly comparable to the growth of Swift.

Steve: Yeah, that does make sense.

Leo: Yeah. Wow. That is fascinating.

Steve: I just always think it's sort of interesting. It's sort of, what, the social network within the developer ecosystem. Interesting.

Leo: Yeah, yeah. I love that stuff, frankly. Thank you, Michael, for the TIOBE index, T-I-O-B-E.

Steve: Yes. I don't think it's my assembler, though. I don't think it's x86. I have a feeling it's...

Leo: Yeah, I bet not. I wish they would tell us by processor.

Steve: Yeah, I think it's probably little embedded things.

Leo: Yeah, you're right.

Steve: And unfortunately, Intel has, like, zero penetration in there at the moment. They're trying to get in, but still not there yet.

Leo: Justin in Austin hit a snag with HSTS and router redirection. He says: The other night my internet connection went - oops, that's the wrong lower third. Let me put yours up. There we go. The other night my Internet connection went down, but the symptoms were surprising. I was attempting to navigate to a site I frequent which, unknown to me, uses HSTS. The result was my web browser informed me the site was known to use HSTS and was attempting to serve me a non-secured page. And the browser said, that's it. It completely prevented me from viewing the insecure page, with no other option.

Eventually I discovered, hey, it was not my browser, it was my router intercepting the web page request and returning its own "No connection, please contact support" page. But the browser's protective interface masked the true problem. If all sites implemented HSTS, it would have taken me longer to figure out. I guess I'm just writing to say that this was a surprising interaction between my router and HSTS. I imagine a WiFi network that has one of these "click to agree to our terms" redirects might show the same symptom. What's going on?

Steve: Yeah, this is really interesting. And we would broadly categorize this as another one of these things where the best intentions of fixing stuff that's wrong don't always work perfectly. So HSTS, of course, is the HTTP Strict Transport Security. That's the reply header which a site, a web service, a web server can provide to a browser, declaring that it's able to always and wants to only have secure connections. The browser caches, that is to say, remembers that declaration. And that gives the browser permission to promote any non-secure queries it might have otherwise made to that server to HTTPS.

So Justin surfs around the 'Net as any of us do, and his browser requires knowledge of all of the HSTS sites, like GRC.com. Mine's one of them. And then something happens. He loses his Internet connection, which causes his router to display an intercept page. And the router's trying to be helpful. The router is showing a page saying, hey, I've lost connection to the Internet. But it's not secure.

So because the router is intercepting the browser's attempt to connect to a secure site, the browser interprets the lack of security, which is not coming from the site, but is coming from the router, as a failure of HSTS, and puts up the wrong message. It doesn't - the browser doesn't say I couldn't get to the Internet because it got to something. It got to the router that intercepted the legitimate attempt to get to the site over a secure connection. And so the browser says, oh, sorry, this site that you're trying to get to, there's something wrong because it said only connect to me securely, and we tried, and it's not accepting a secure connection. Well, because the router intercepted it.

So again, this is sort of one of those oops, unforeseen side effects of a system which we're trying to make more secure. But this whole concept of HSTS, we have to call it a kludge. Yes, it's a useful kludge. It's giving us more security. As we know, it's preventing the problem of a bad guy being a man in the middle and stripping the HTTPSes out of all of the responses coming from a site in order to strip the security and be able to, for example, acquire session cookies when an unwitting user logs in. So this was sort of

there was an edge case of a problem where the industry's response was, oh, we can fix that. And so this kludge, useful and workable, but still an afterthought, was added to the HTTPS and generally sort of the web protocol to enforce security through this clever mechanism.

And unfortunately, it was a little too clever for its own good; and, exactly as Justin notes, if you went to a web portal that was trying to intercept you, I would imagine that you'd get the same sort of confusion from the browser. Now, maybe, if this becomes more prevalent, browsers will just do a better job at explaining the situation. The browser could understand better what's going on and present something that's less confusing. And Justin's point about, if all websites were doing this, what he meant was, he just tried a different site, and it worked. So, oh, I'm sorry, he tried a different site, and it didn't work because the problem was not the site he was trying to get to. The problem was he lost his Internet connection. Unfortunately, that was being masked by this kludge that was all put in place in order to keep us more secure.

Leo: Huh.

Steve: Yes.

Leo: Two steps forward, one step back.

Steve: Yes, and if we keep doing it, we do actually make progress.

Leo: Yeah, bit by bit.

Steve: But not always as fast as we want.

Leo: Walt in San Fernando Valley wonders about DDoS attacks: We've all heard of DDOS - Distributed Denial of Service - attacks against individuals and businesses. I've never heard of an ISP being attacked. Is such an attack possible? If so, why are bad guys not doing it? Or maybe ISPs just don't tell us when they're attacked? Many thanks.

Steve: So this is sort of an interesting question. And I think there's a couple answers to it. Normally, DDoS attacks are an attack launched by a individual or group against an entity for some perceived cause, some perceived slight of some kind. So it's like Company X does something that this group disagrees with philosophically, and attacks them. Now, and so my point is that ISPs, who are generally just carriers of content, are by that nature less targets of people's angst, I think. Which is not to say that people can't also get upset with ISPs. But that's part two of this, is that the nature of a distributed denial of service attack is that the traffic coming from all over the world, typically, is concentrated through many different ISPs, down finally to the subscriber's ISP, and then to the subscriber.

And depending upon the size of the attack, the ISP's routers at some point generally get overwhelmed when the concentration of traffic begins to saturate the bandwidth

capability of either the router or the outbound links from the router, so that it just can't allow all the traffic through. So ISPs are sort of indirectly attacked when any of their customers are attacked. But exactly as Walt also suggests, maybe they're just too big. What they are mostly is a large ISP will have, as we've talked about, peering with other ISPs or Tier 1 providers, will have many different contact points around the Internet. Many different contact points means that the traffic is far more diffuse in nature.

And, I mean, I guess you could attack an ISP's website. I was about to say that an ISP doesn't have an IP address in the same way that one of their clients has an IP address. An ISP has large blocks of IPs, which are then assigned to their customers. So you could certainly attack an ISP's website, where it lives. But by its nature, an ISP itself is potentially millions of IP addresses, rather than just one. So there really isn't anything to attack. And even if there were, the traffic is inherently spread out to a much greater degree. So I just sort of think they're sort of part of the cloud, and they're sort of part of the fuzziness that you attack through, rather than trying to attack it itself, because itself it's sort of amorphous.

Leo: Makes sense. Gregg Penn, Oakland, CA wonders about blocking ads only from running scripts. By the way, I don't know if you heard Triangulation yesterday. We talked...

Steve: Did not have a chance to catch it, but I know that you had those guys on.

Leo: Two people from Adblock Plus. We talked about the "acceptable ads" policy and all of that. And then Dean Murphy, who is the guy behind the Crystal adblocker on iOS.

Steve: Right.

Leo: And, yeah, it was a really good conversation, I thought, along the lines of stuff we'd talked about before.

Steve: Sure.

Leo: But in some detail. So that was yesterday's Triangulation. Anyway...

Steve: It's one of the other tensions that we have in the industry right now.

Leo: Yeah, yeah. Gregg writes: Steve, I've been listening - mostly reading actually, thanks to Elaine's work - to Security Now! for a few years. I find the discussions about adblocking to be very useful and interesting for the different perspectives everyone has on it. I was wondering if it makes sense to try to have a lighter touch on adblocking by combining it with suppressing scripting. Instead of blocking third-party data like ads, stopping scripting, or both, what about blocking only scripts that come from third-party sites? Wouldn't that improve security? Would it prevent the

website owner from getting credit for the ad view? You could still see most ads without them running Flash or JavaScript. It would be a lot less annoying. And finally, do any current ad or script blockers include this functionality? And then I think you have a screenshot here of uBlock Origin.

Steve: Yeah. So, yes. The current favorite tool of mine and many now of our listeners is uBlock Origin. And it does have specifically, and this screenshot shows it, there's a line item there, third-party scripts. And it's not normally accessible to a user until you turn on the advanced mode. So you need to go to whichever browser you have it installed, whichever one you're using it on, to the configuration. And on Firefox, I'm looking at it, it says "I am an advanced user." Which is not checked by default. You turn that on, and then you will find some little plus and minus signs to the left of the requests blocked and the domains connected verbiage on the UI. If you click that, it opens this whole next level of drilldown panel. And one of the items there is third-party scripts.

Now, conservative as the author of uBlock Origin is, this is not enabled by default. You can experiment with it because he has both local and global settings in the way this operates. If you're not familiar with this, check back on the podcast where we covered uBlock Origin in much greater detail [SN-523] because there is per-site and global options. So you could block third-party scripts and see how it goes. The problem is that ads which may detect that they are being blocked are third-party sourced, and many ads are now using scripts. We've talked about this. And for me, it's an annoyance, which is why I brought uBlock Origin up, and I'm unfortunately feeling I need to block pretty much everything, although I am making static exceptions for sites where I want to support them. And it looks like they're doing what they can to have a useful ad policy. But for what it's worth, uBlock Origin is an example. It gives you all the bells and whistles that you need in order to say, I'm going to try blocking third-party scripts and see how it goes.

Leo: uBlock Origin to the rescue once again. Brett in Dubai wonders whether we ever really even needed TrueCrypt: I completed my offsite backup moments ago, and I suddenly realized I'm completely happy with using my Mac-encrypted filesystem on removable disks. He's talking about Apple's FileVault, I guess. It's easy and, as far as I know, secure. I was using TrueCrypt on your recommendation for years and was really upset when it became unsupported. So I switched to using the readymade built-in OS encryption systems in Mac - actually systems because Mac uses FileVault, Windows uses BitLocker. Do you think they're good enough, or are we missing something?

Steve: I guess the question is what's your need? What's your application? That is, what degree of encryption assurance do you want? I think they're absolutely good enough to protect your content in the cloud, to protect your backup, and to protect your system. I can't help but be nervous about BitLocker. Just, I don't know, Microsoft just - I have less of a feeling of comfort with Microsoft than I do with Apple. I think Apple has - maybe just because Apple has been so vocal and so adamant about their stance on their enforcement of user privacy. And I understand what they're saying. Microsoft just doesn't say anything. And so I just - I don't feel like I have the same sense of confidence from Microsoft.

But, yes, they are strong encrypted file systems. I think you are always, in an absolutist sense, always better off using an open source, third-party solution, just because it's

better if it comes from somewhere else for absolute ultimate security. But I don't know that everyone absolutely needs that. And there's a huge convenience factor in simply turning on the Mac FileVault and having an encrypted file system. I really think it's fine.

Leo: Steve's not being blunt enough. If you're worried about government, then you might not want to use a corporate solution, a closed-source corporate solution because - and I don't care if it's Apple or Microsoft, whatever they say or don't say. Both are subject to U.S. law, and both may be required to provide a backdoor in their encryption. So if it's government, the U.S. government you're worried about, then open source would be safer. I've always said, if you're going to use encryption, use open source. But it is a lot less convenient. And people have - no, I wasn't talking to you, Siri. We have - I don't know why she woke up. Oh, Apple wanted to hear what I had to say, okay. Did you get that, Apple? That's really the issue; right? If you're worried about just common garden variety thieves and snoops, of course FileVault and BitLocker are fine.

Steve: Yes, absolutely.

Leo: It's the U.S. government. That's the difference.

Steve: Yes.

Leo: Ronnie in Montgomery, Alabama has an idea for fingerprint scanning: I just had a wild idea, says Ronnie. I've liked the idea of fingerprint scanning. I don't like the idea of it being the only form of unlocking my phone because I can be compelled to use my finger to unlock it. But what if the fingerprint scanner was programmed to check two or even three prints for multiple fingers? Only I would know which fingers, and in which order. No one would be able to just cut off my index finger thinking that was the one that unlocks it. It would be something I know - oh, because only he would know the order.

Steve: The sequence.

Leo: Yeah. So I couldn't be compelled to reveal the exact pattern. Oh, that's clever. This, of course, wouldn't be very practical. But for the ones who require that extra level of security, this could be an ideal solution. What do you think? Be gentle if it's dumb. Ronnie.

Steve: So I don't think it's dumb. But I think that probably the answer is a little simpler. And that is to have a panic mode finger. So I hesitate to say that you would give your phone the finger if the authorities wanted you to unlock it. And I doubt that it's the kind of thing that we would see Apple support. But it sort of feels like the thing that maybe Android could offer, where you would register the finger that you intend to use for unlocking, and then you would register the panic finger.

And if you really wanted to be safe, your regular finger would be unusual, like the pinkie on your non-dominant hand, so that no one would expect that to be your normal unlock

finger. And your what would expected to be your normal unlock finger, like your dominant hand index finger or thumb, that would be the panic. And so you would comply. You would say, oh, okay, fine, you got me, I'll unlock my phone, and you just give it one finger, and now it's bricked, essentially. I think that's probably the practical way, only because doing a multiple finger salute every time you want to unlock your phone probably would get tiresome, if you had to do three different fingers in sequence. I mean, yeah, you'd feel like James Bond, but I think you'd get tired of being James Bond pretty quickly.

Leo: Continuing on with yet another question for you, Mr. G, via a private tweet from someone who wishes to keep his identity private, a little DM action here. He writes: I never thought I'd be caught by CryptoWall - oh, no - but it happened. However, it happened because of a Windows feature I didn't know about, the Microsoft Windows Scripting Host. Oh, WSH. I was looking at a JavaScript file sent to me in a bogus email, and I accidentally double-clicked the .js file. It was immediately parsed by WSH and ran. Well, duh. Have you discussed this vector before? Why on Earth is this a built-in feature of Windows? I understand the need and value of scripting, but damn, allowing invocation of a downloaded JavaScript file with no warning? Wow.

Steve: So this is something we've never talked about. We've talked about JavaScript running in browsers. But it turns out that somewhere along the way Microsoft decided that they would allow JavaScript to be interpreted natively. On Microsoft's page discussing this they say: "The Windows Scripting Host" - which is WSH - "supports scripts written in Microsoft Visual Basic Scripting Edition or JavaScript. When you start a script, the scripting host reads and passes the specified script file contents to the registered script engine. The scripting engine uses file extensions - .vbs for VBScript, .js for JavaScript - to identify the script, instead of using the script tag used in HTML. Because of this, the script writer does not have to be familiar with the exact programmatic ID of various script engines," which in Windows jargon you would otherwise need to use in order to invoke the scripting engine.

And they conclude, saying: "The script host itself maintains a mapping of script extensions to program IDs and uses the Windows association model, that is, its file extensions, to start the appropriate engine for a given script." So I could empathize because, as a security person, there have been times when I've been messing with live viruses or live malware. In fact, CryptoWall was one of them. And it is just really unnerving. I mean, it's like we've all seen science fiction movies or various - or maybe even not science fiction, where somebody's dealing in a Level 4 biohazard lab with live virus, and you just really need to be careful. And our instincts about clicking and double-clicking, I mean, he clearly meant to select it, click once to drag and drop. Well, and Leo, you're making a face. But believe me, it's so easy.

Leo: This is why we say don't open attachments, because there are many, many executable extensions besides EXE.

Steve: Right.

Leo: You could, you know, a .pif file is an executable. You know that, I know that,

because we're old DOS guys. But you can't look at it and say, well, I wonder what happens when I double-click it. Don't. And by the way...

Steve: I know. No, and so but his point was that he's a Security Now! listener. He knows how to DM me. And he never thought he'd be caught by CryptoWall. So he was deliberately inspecting, because he was curious, a JavaScript host...

Leo: Maybe he thought it would open up in text edit or something.

Steve: Well, exactly, or he may have intended to right-click on it and then select "open with." I mean, it just - it is so - there's a weird thing that happens when you know you're dealing with something...

Leo: [Crosstalk] accidentally double-click it.

Steve: Yeah.

Leo: But JavaScript by itself wouldn't infect you. You'd also have to have a vulnerable system, wouldn't you?

Steve: The Windows Scripting Host is extremely powerful.

Leo: It runs at a high...

Steve: You can do stuff in there that you can't get from the command line.

Leo: So CryptoWall is not an EXE, it's a JavaScript file?

Steve: No, no. But, for example, it could have gone out and fetched it. That's the kind of thing...

Leo: And run it?

Steve: Oh, yeah, the scripting host...

Leo: Without a UAC warning?

Steve: The scripting host can do all kinds of things.

Leo: This guy turned down the security on his system. Or he has vulnerability...

Steve: He might be running as admin.

Leo: Yeah.

Steve: Yeah.

Leo: Well, not just running as admin. You would get a UAC warning, I'm sure of it, before it would allow a program to run, be downloaded and run by JavaScript. Well, maybe not. I don't know.

Steve: Remember that you're not in the browser. The browser has a different...

Leo: No, I understand, it's WSH.

Steve: Yeah.

Leo: No, I understand.

Steve: Yeah.

Leo: But UAC runs system-wide. It's not just in the browser. It runs system-wide.

Steve: But UAC is not just a run a program. It's saying something you're doing is needing elevated privileges. And I don't know whether CryptoWall...

Leo: Maybe not, maybe not, yeah.

Steve: Yeah.

Leo: Guy Smiley in the chatroom said, no, you wouldn't get a warning. So it wouldn't have to be an exploitable computer, and you wouldn't see a UAC warning, it would just be running it.

Steve: Yeah. And again, I really wish I had like an extra year to mess with the PowerShell because, boy, what Microsoft has built into Windows is so amazing. But that's not what I'm doing right now.

Leo: WSH is great. PowerShell has kind of superseded that, I think. But WSH is the engine that can run ECMAScript and these other - VBScript.

Steve: Right.

Leo: Nathan Rae (@nathanrae) tweeted @SGgrc. Long tweet: Steve, I have a question. As most crypto is based on large prime numbers, how many primes are there to choose from? I looked it up and there are about a billion, but I don't have - what? But I don't have any idea of how many there could be in a 128-bit binary number. My brain doesn't work with numbers that big. Well, that's an interesting question because there's an infinite number of primes, but in a constrained space there would be a number.

Steve: And isn't your intuition - it turns out this is a place where our intuition is wrong because we know that a prime is a number that is not divisible by any number other than itself and one.

Leo: Right.

Steve: Of course everything's divisible by one.

Leo: Right.

Steve: But no other numbers in between one and it are divided evenly.

Leo: Right.

Steve: And so, you know, we start one, three, five, seven, nine, then 11 and 13 and 17 and 19 and so forth. But sort of intuitively, if you actually write them down, you notice they start getting further and further spaced apart because there are more opportunities for them to be divisible by one of the things that came before.

Leo: Right.

Steve: And so your intuition sort of assumes, well, we're going to kind of run out. Like way, way out there...

Leo: Way, way.

Steve: There are just so many other numbers that have come before, certainly one of them would be able to divide by that. And so it's a really interesting characteristic of

primes that they continue to be richly available forever.

Leo: Yeah.

Steve: And in fact the way primes are chosen for crypto, like when we get two primes and want to multiply them in order to do a crypto operation, is a pseudorandom number generator, high-quality of course, just makes one up. And then what's called a "primality test" is done on it to see if it happened to guess one that's prime. And if not, it guesses again. And its primality is checked. And so on. And it turns out that, no matter how far out you go, that there are still primes like all over the place, which to me was a bit of a revelation when I first encountered it. I thought, isn't that interesting. They don't get more rare. They're just all - they're just scattered. And actually they're still - they're prevalent, even.

So it's not like they're really hard to find, and the fact that they're hiding because they're hard to find makes them more valuable. In fact, that would make them less valuable because, if they became really rare when they were really big - we need them to be really big for security. If they also became very rare, then it would be much easier to guess what they might be. But the fact that they're all over the place when they're really big, and a high population of them, means that that doesn't confer an advantage on somebody who's trying to guess what the primes might be that you chose by random and then worked with. So I just - I love this stuff.

Leo: But I guess in 128 bits there are a limited number of primes. Any subset of the infinite.

Steve: Well, and that's where I mentioned he's sort of confusing things because 128 bits is...

Leo: Just the size...

Steve: ...only relevant to symmetric cryptography, where you choose your key at random. All of the primes are useful for public key crypto; and maybe, well, in an RSA crypto you're dealing with primes like 1024 and 2048 bits. So really much longer ones. You would not be using primes down at 128 bits because they're just not - there are not enough of them down there to have security. There it really does become too small. Nobody uses 128-bit public key crypto. We use 2048 public key crypto.

Leo: So I thought I'd try this. I saved a little JavaScript file as a text file to my desktop, just to see if I could just double-click it and run it. This is Windows 10. And yes, indeed, the Windows Script Host ran it without warning, and that was it. So I stand corrected, and I apologize to our anonymous tweeter. In fact, yeah, if you have a .js file, and you double-click it, whether it's an attachment or just on your desktop, it runs. No warning.

Steve: Yeah, and clearly he knew that he got bit because he double-clicked out of habit.

Leo: Right.

Steve: I'm sure, as you said, Leo, he wanted to inspect the JavaScript, just out of curiosity. But instead he launched it. And he was a little bit surprised that Windows runs JavaScript.

Leo: So don't. Yes, it does. It's an executable. Don't run it. It's an executable. I'm kind of impressed that CryptoWall can use a .js file to install itself.

Steve: It's only because that Windows Scripting Host is unbelievable powerful. You can edit the global profiles. You can do...

Leo: Oh, it's great, yeah.

Steve: ...group policies. It's just incredible.

Leo: Yeah. Mike in San Francisco brings us the "You're Doing It Wrong" Foible of the Week. Not fable, foible. So, recently I tried to log in to Wells Fargo. My first mistake. The site informed me I must pick a new username and password to meet its new security standards. Oh, that's scary.

Steve: Mmm.

Leo: I looked at what was required and changed my username slightly. Then I tried to enter a new password generated by LastPass. Despite the fact that Wells Fargo's own new password hint suggestions were indicating a password up to 14 characters and all sorts of other rules, it wouldn't take anything longer than eight characters. This took a while to untangle. It kept rejecting everything I tried despite telling me the length was right, until it occurred to me to ignore that and backspace one character at a time, trying the submission each time. When I finally whittled it down to just eight characters, my new password was considered valid and accepted. This is pretty bad in this day and age. It's even worse the password hint text doesn't even match its own validator. What?

Steve: Wow.

Leo: Wow.

Steve: This sounds like a committee that needs to have more meetings, even though committee meetings are notoriously awful. It's like, how, like, three different groups are all working on their own aspect of this, and they're not talking to each other. And, you know, wouldn't you think someone would have tested this? That, like, how did this not get tested on a password longer than eight characters, when the whole reason Wells

Fargo was insisting this happened is their new security standards.

Leo: Yeah.

Steve: Wow.

Leo: I'm going to guess that it's also possible that maybe there was a disallowed character in the other characters that he deleted.

Steve: That could be. That could be it. It's like character nine could have been a curlicue or something, and...

Leo: But, boy, eight characters is not enough, is it? Even if it's...

Steve: No, and eight characters...

Leo: [Crosstalk] upper and lower and numbers and combinations. It's not great.

Steve: Even with a big alphabet, that's just - you cannot get enough entropy into eight characters.

Leo: Yeah.

Steve: If we assume about a six-bit entropy per character, which is a good rule of thumb, then we've got six eights is 48. So that's 48 bits of entropy, assuming completely random. That's just not - 48 bits is not enough security.

Leo: Alan Ambler is our last question from in Cincinnati. He brings us the NetWorx feature Tip of the Week. What's NetWorx?

Steve: Okay, so NetWorx is what I talked about, that really cool network monitoring app a couple weeks ago.

Leo: Oh, yeah, yeah, yeah. Right, right, right. Now I remember, yeah. He says: I've been using NetWorx for two or three years. You're right. It rocks. My fave feature is the ability to set an alarm for streaming below a "per minute" threshold when automatically doing stream capture recording. In other words, it alarms and notifies me if the connection drops. If a certain level of bytes per minute are not achieved within a three-minute interval window, I get the annoying three-tone phone company disconnected alarm. It's great for unattended downloading of huge files which might not complete.

Steve: Well, and so I just wanted to mention it again, N-E-T-W-O-R-X. I got a lot of positive feedback from my recommendation. And one person mentioned that he uses it, not to monitor his own machine's bandwidth, but his entire network's. And it's like, oh, that hadn't occurred to me. Turns out right there on the UI, where you select which network interface of your machine you want it to monitor, if you have more than one, there is a clickable link, "Monitor my router instead."

And what is so cool is that there's a protocol, I don't think we've ever talked about SNMP, Simple Network Management Protocol. SNMP is a UDP-based, very simple messaging protocol. You send a UDP packet to port 161, which is the default SNMP listening port, with a query in the form of a long dotted number. It's crazy. It's all standardized. They're called MIBs, M-I-B's. And so it'll be like 20 dotted numbers. It looks like an IP address from, like, three times over. But many routers support SNMP, that is, you can ask them things. And one of the things you can ask them is for the count of bytes in either direction on one of their interfaces. They just total them all the time. And so if you ask every second, and you get two totals, you subtract the second one from the first or the first from the second, and that gives you the number of bytes in that interval of time between queries, which converts then to bandwidth.

So the point is that just by clicking that little link on NetWorx - oh, first it said - it didn't come up with a list of interfaces. And I thought, ah, I'll bet I never turned on LAN-side SNMP on my pfSense software on my little router. So I logged into pfSense. Sure enough, SNMP was not turned on. I enabled it. It took a second, then it came up. Then I went back over to NetWorx, said I want to monitor my whole network. Now it showed me all of the interfaces. I chose the one for the LAN. And ever since, I've been watching the bandwidth, not only of my own machine, but of everybody, all of the systems and iOS devices and so forth on my network.

So again, props to NetWorx. It couldn't be easier to do that. I know I went into a lot of detail and made it sound complicated. But there's no need to look up MIBs and dig around or do anything. It just said, hey, yeah, I want to monitor my whole network. So very cool little piece of freeware. And thanks, Alan, for bringing it back up, and all of our listeners for saying that you know about it, or that you were glad for the reference.

Leo: Great. I've got to try that. Steve, we're done.

Steve: Yay.

Leo: We got them all in.

Steve: Still the same day, too.

Leo: Wow.

Steve: It's still Tuesday.

Leo: Did you have a good Thanksgiving?

Steve: Yes, very good. Just went out with a bunch of friends to a nice restaurant, relaxed, and had a couple bottles of '05 Jordan cab and...

Leo: Nice.

Steve: Yup.

Leo: Yeah. I know the Jordan family. They do a very nice job.

Steve: My favorite cab.

Leo: Is it? Oh, that's good.

Steve: Yup, it is, yeah. I ran out, I was drinking '03 Silver Oak for a while. I ran out of the '97, wait, no, the '07.

Leo: You love that Napa stuff.

Steve: I do, yeah.

Leo: I know the Silver Oak people, too.

Steve: I'm a California cab boy.

Leo: I know all of those people. That's funny. I served on the school board with the Silver Oak family and the Jordan family, Judy Jordan and Tim Duncan of Silver Oak. They're very nice people.

Steve: Well, they make very good cabernet.

Leo: I'm not on the board anymore, or I would get some for you. We do this show every Wednesday at about 1:30 Pacific, 4:30 Eastern time, 21:30 UTC. If you want to watch live, we love it. But if you can't, and I understand, you like to listen in the car or while you're on the treadmill or whatever, this is always on demand, and that's at GRC.com. That's where Steve has his audio versions. He also has great transcriptions from Elaine, as you've heard. Some people just read the show.

Steve: Or they read along.

Leo: They read along. I think that's a good idea, actually.

Steve: Yeah.

Leo: We also have audio and video at TWiT.tv/sn. It's on YouTube.com/securitynow. It's on every app. We've got TWiT apps everywhere, including four now on the Apple TV. If you want to watch, you could do that. Stitcher, Spotify, iTunes, everywhere. Just subscribe. That way you won't miss an episode. And I know there are lots of schools that use this for curriculum. Thank you. Thank you. This show is, you know, all of our shows are growing, but this show's growing faster than any of them, and it probably has something to do with the subject matter, but I think it also has a lot to do with you. Nice job, Steve. Thank you. If you want to get SpinRite, you ought to. That's at GRC.com, as well, along with all the free stuff Steve offers all the time, including ShieldsUP!, all that stuff. Thanks, Steve. Have a great week.

Steve: Next time we'll do, I think, a deep dive techno episode. So everybody get ready.

Leo: You know what the subject's going to be? Or is this...

Steve: We'll have some fun. There's two topics. There's Australia did a really bad, some sort of chip-and-pin standard that Matthew Green had some fun with, that I haven't had a chance to dig into yet, but I'll see about that one. And then there's also a horrible security device which is like a residential security system that apparently got everything wrong. So it might just be a gourmet episode of, boy, are you doing security wrong.

Leo: And I begged Steve, and I know he's tabling or considering it, to do a block chain episode. We've done Bitcoin, but block chaining in general.

Steve: Yes, we need to, right.

Leo: Because I think I need to understand better how it's used for things beyond currency.

Steve: Yes, and that is a...

Leo: By the way, we record this Tuesdays, not Wednesday. The chatroom's saying, what are you talking about? This is Tuesday, Tuesday, 1:30 p.m. Pacific.

Steve: This feels like Wednesday by the time we're done.

Leo: It's my Thursday, but I don't want to confuse people.

Steve: That's all right,

Leo: Thank you, Steve. We'll see you next time.

Steve: Thanks, my friend.