Listener Feedback #119

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

High quality (64 kbps) mp3 audio file URL: http://media.GRC.com/sn/SN-304.mp3
Quarter size (16 kbps) mp3 audio file URL: http://media.GRC.com/sn/sn-304-lq.mp3

Leo Laporte: This is Security Now! with Steve Gibson, Episode 304, recorded June 8, 2011: Your questions, Steve's answers, #119.

It's time for Security Now!, the show that covers your security online, and privacy, too. And here he is, our guru, our leader, our fearless leader Mr. Steve Gibson - actually, he's fearless, but he scares the hell out of me - from GRC.com, the Gibson Research Corporation, creator of SpinRite and a lot of great utilities, and now the king of passwords, too. Did you get some good reaction on the Haystacks episode last week?

Steve Gibson: Tons of reaction. I didn't want to make this Q&A all about the reaction to last week's Password Haystacks episode, so it's about 50/50: some great comments, some things that I sort of forgot to mention that people brought up, some very valid criticism saying, you know, padding's not the same as entropy. So we've got some good things to cover, and not too much news today, but some interesting news. So I think overall a great podcast for everybody.

Leo: I can't wait. All right, Steve. Happy IPv6 Day. What the hell does that mean?

Steve: Well, I noted months ago that this date, June 8, would fall on a Wednesday. And so I thought, hey, cool, we'll actually be talking to each other, we'll be recording a podcast on IPv6 Day. My biggest disappointment, though, is there's no fancy Google logo for it.

Leo: Oh, you'd think they'd have a Google...

Steve: Oh, there's Google logos for, like, when people discovered navel lint. I mean...

Leo: Yeah, and there's nothing here, nothing.

Steve: It's just ridiculous.

Leo: Just a blank old Google.

Steve: I went there, and I thought, okay, come on, give us something really cool, like Rube Goldberg plumbing or something. But no. We got nothing.

Leo: And I imagine Google is fairly involved in IPv6 Day. This is in their interest, isn't it.

Steve: They're one of the big people, and Facebook, and Yahoo!, and a number of other big ones. First of all, it means really nothing for end users. The idea is for major websites to bring up IPv6 services, sort of as a dry run, sort of as a test. And the idea is to sort of test the infrastructure, let the engineers see it in use, see if anything happens. I'll have more news next week when I've been able to aggregate the reactions from what's going on today. I've seen some indications that some small percentage, like 0.02 percent of the 'Net might have connectivity problems if, as a consequence of this being activated, their traffic happens to go over IPv6 through really no effort of their own except that their infrastructure or the extra structure is using IPv6, and if there are problems. So...

Leo: Where would the problem lie? Is it my router that I'm going to have a problem with? Is it...

Steve: Well, here's the deal. The entire Internet today is IPv4, which means every single piece of equipment between you and me, Leo, or you sending out this real-time stream, and every single one of the listeners knows IPv4. That is to say that that's the protocol that wraps the data packets which move from point to point. Today it is not the case that even the majority of the Internet's plumbing equipment knows IPv6. Which is to say, for most equipment on the 'Net, if an IPv6 packet came to it, it wouldn't know what to do with it. It would just ignore it because there's a version that's, I think it's a byte in the IP header that says this is my version. All of them in the world say "4." Every packet out there has a 4 in there. And that's one of the things the router hopefully checks as the data's coming in, it checks to make sure that it's 4. Well, when packets arrive and it's 6, the router's going to go, huh? Because, I mean, everything's different, the 128-bit IP, for example...

Leo: You hope it checks because if it doesn't check it would be worse than "huh." It

would barf; right?

Steve: Yeah, and, I mean, we know that the way things are engineered, since everything has always been 4, it might very well be that there is equipment out there that doesn't check. Which means it might crash, it might stumble, it might hang, it might drop the packet. But the key is, for it to work, for IPv6 natively to work, every single piece of equipment, and there's a lot of it between any two IPv6 endpoints, has to know IPv6.

Now, there are tunneling solutions where you can chop up bigger IPv6 packets into multiple IPv4, hide them in IPv4 so that the existing infrastructure still works, like if it had to pass through a realm of the Internet that couldn't transit, for whatever reason, or wasn't yet able to, IPv6. You could have tunneling endpoints that rewrap the incompatible IPv6 in IPv4. It goes where it needs to go, then it gets unwrapped again. So, I mean, there are those kinds of kludge-y solutions. But those are no better than, like, NAT routing that we have because we're running out of IP space.

So ultimately what the goal is, everything will be running IPv6 and still know about IPv4. But we're an unknown length away from that. I mean, even companies which have their own IPv6 big monster allocations, their traffic will be converted to IPv4 not very far after it gets out of their facility because that's all the 'Net actually works on today. It doesn't actually work on IPv6 yet except there are some sites that are, today, on June 8th, giving it a shot, sort of like offering their services there to see how it goes.

Leo: So you have to request an IPv6 packet. It won't just spit it out at you.

Steve: Well, the way it works is you have - we've talked about TCP/IP stacks. And every TCP/IP stack for several generations of our operating systems has been IPv6 capable. So, for example, when you enable IPv6 on XP, and there's like a command you can do to turn it on in XP, when you do that, then it will attempt to issue traffic over IPv6. And when that fails, as it invariably does today, it falls back to IPv4. So eventually, when you connect that machine to something that does know about IPv6, then it'll work. And it'll just sort of work seamlessly. So Mac OS X has had it for several versions. I just saw this morning where, like at what point they added it. It might have been Leopard. Was that a while ago?

Leo: Yeah, Leopard. Now, Snow Leopard's the current. I bet you they did in Leopard. That would make sense.

Steve: Okay, Leopard, yeah. And UNIX has had it since 2002. And Microsoft has been experimenting with it. You can, like, install an experimental IPv6 stack. Even as far back as Windows 2000 the Microsoft Research folks had that working. So it's in all the current operating systems. But right now there's no way - and if you were to plug two IPv6 machines directly into each other with a crossover cable, they'd happily talk IPv6. But it's the connectivity problem. It's all the equipment in between. Every single piece of it has to be able to handle IPv6 or translate to IPv4, tunnel, and then translate back.

But we're just not there yet. I mean, we're not even close. I'm having to fight to get IPv6 service on my T1s. And it's like I had to go through a big conference call with Level 3 to get it at GRC. So it's not, like, just happening. And it won't really be until - it's just like

raising the debt ceiling on the U.S. It won't be until it absolutely has to be done that somebody will say, okay, fine. And then people will scurry around and...

Leo: IP chicken, we're playing. So Dr. Mom asks, in the chatroom, and I'm sure everybody wants to know, is there anything we need to do as end users?

Steve: At some point our routers will become an interesting issue. That is, if cable suppliers need to provide their customers with more IP space because the cable supplier itself is out of IPv4 for its own use, then they would upgrade their cable modems and/or their associated routers. It's conceivable that at some point an ISP could say, we're discontinuing IPv4 support to you, Customer X, on a certain date. And it will be a ways out in the future. In which case Customer X would need to go to Staples or Fry's or Amazon, wherever, and upgrade their router. Maybe flash the version of the kernel on their router to a new version, that's possible. But probably just get a new router because the old one's going to be old anyway. And then that router would be able to understand IPv6 on the public side.

Now, if it passed IPv6 through on the private side, then, finally, all of these machines that have been sitting around, our operating systems, for the last couple years waiting, would suddenly be able to use IPv6. I mean, it doesn't do anything really more for us. We'll be talking about it in detail. It's got some security IPSEC protocol built into the definition rather than being layered on top. So there are a few nice things about it. Mostly it's massive address expansion and also has a bunch of features that make routing tables for the Internet, which are already a problem because they're just so fractured and fragmented, it brings a new architecture there that allows the Internet's infrastructure to work a little bit more efficiently.

But what I'm betting is any home router, any SOHO router that people buy will always be offering IPv4 and DHCP so that our older appliances, our operating systems that we've had for the last couple years, will always be able to talk to the router with IPv4, and then it will do the translation to IPv6 to get out onto the public Wide Area Network, on the WAN side. But this is still years away.

Leo: There's also been a lot of speculation that what'll happen is that there will be ISP-level NAT translation, and you won't even see it. Won't even happen.

Steve: Precisely. I wouldn't be at all surprised.

Leo: I bet you, I mean, that seems like the - nobody's going to - you're going to ask my mom to upgrade her router? The ISP would get all these calls.

Steve: There is mess at every level. I mean, this is why there's the inertia and the reluctance is everyone's, like, well, if we don't have to do it, we really don't want to. Maybe it'll just go away.

Leo: Well, it won't go away.

Steve: It's not going to go away.

Leo: We know that.

Steve: Yeah.

Leo: It's just a question of how we deal with it.

Steve: So I rebooted my machine the other day. I don't remember why. I don't do it often. But I got the notice from Adobe that I was getting a new version of Flash, which...

Leo: That's why you should reboot every day.

Steve: ...we've been waiting for. We talked about how there was a new cross-site scripting vulnerability which was affecting Flash players and how Adobe might as well just give up this notion of quarterly updates. And so what they did was an out-of-band fix for this zero-day vulnerability that we have been speaking of in the last couple weeks. Google's Chrome browser got it fast and first, and there is now a new version for Windows, Mac, Linux, Solaris, and Android. So sort of across the board that's been fixed. So that's good.

The big news, the other shoe to drop, essentially, on the RSA fiasco from 90 days ago - I looked back at my blog posting, and that was March 17 when I went out on a little bit of a limb, and I said, okay, the only way to read what RSA is not telling us is they lost the keys. And, I mean, there just isn't any other way to interpret it. Well, they have 'fessed up.

Leo: They lost the keys.

Steve: They lost the keys. And we're talking 40 million of them. And in an open letter to RSA's SecurID customers, the chairman, Art Coviello, said in an interview that RSA, well, he said in an interview separate from the open letter, he said: "RSA is offering to provide security monitoring or replace SecurID tokens for virtually every customer we have."

Leo: Every one of them.

Steve: Every one.

Leo: So they really did get compromised.

Steve: They did.

Leo: Wow.

Steve: And Fox News also reported that Northrop Grumman was another defense contractor that had been attacked using compromised RSA credentials. So they have formally admitted that they lost the keys. And now people are saying, okay, I mean, they're saying what you and I were saying three months ago, why were they on their network? I mean, why could those have been compromisable? And the answer is always the same: convenience. It's easier. So that's what we did. Well, they've certainly learned a lesson the hard way. And I hope so many other people are paying attention because we don't want everyone to have to learn the lesson the hard way. Be nice if people could - if CEOs or chairmen of the board could say to their CIOs, hey, tell me how this cannot happen to us. Prove to me that it cannot happen to us. So I hope that's happening.

We have early feedback from the really cool new Microsoft Safety Scanner, which remember is the malware and specifically rootkit scanner which you boot and check your systems. I've seen a number of tweets from people who follow me, mentioning that they had found malware on their machine. Microsoft reports that it's been downloaded, at the time of this report, which is pretty recent, 420,000 times, and has found and removed malware from 20,000 machines. Looking at what it was that it found - it provides feedback to them so they're able to get a sense for what it's doing - seven of the top threats that were found to install malware, the malware that this thing found and removed, were Java-based exploits. Remember, Java, not JavaScript. I want to make sure - in email that I receive and in tweets that I see, I see that mistake being made all the time. And so I want to make sure people understand Java is not JavaScript and vice versa. So these are Java-based exploits, not in this case JavaScript-based. However, the SANS Institute editor Eugene Schultz made a good comment when SANS was reporting this. He said, "I believe Microsoft's reported infection rate is too low. Users who do not have a clue concerning how to secure their systems almost certainly have much higher infection rates." And the story did say 5 percent. So he says, "...almost certainly have much higher infection rates."

Leo: And they ain't running any Microsoft Scanner software.

Steve: That's exactly his point. He says, "These users are not aware of Microsoft's Safety Scanner, let alone of how to download and run this tool; but more sophisticated and security-aware users are. Microsoft's statistics thus in all likelihood apply almost entirely to the latter group." So that's a very good point is that, I mean, anyone who's running this is already savvy. They're listening to this podcast. They're able to burn bootable CDs or USBs and reboot systems. This is not your typical user. So it says something that 5 percent of those users are being caught out by seven out of ten are Java-based exploits and finding rootkits on their machine that were otherwise hidden from them.

Leo: Although, as we noted, when you go to that site, it says, "You've been sent here by Microsoft Support." Presumably, of the people sent by Microsoft Support, it's like 90 percent have rootkits. So that may skew the numbers also. It's hard, I think you just can't tell from the 5 percent. It's not meaningful.

Steve: Yeah. I went to docs.google.com an hour ago in order to put these documents up for you, Leo, and it was over HTTPS. Certificate Patrol, which is the add-on I have

mentioned a number of times and have been liking, popped up and said, "Certificate exchanged. Mostly harmless." And then in the far right it said docs.google.com. This is the first time I have received that. And I was tickled that it did this.

So what this was doing was - this is exactly what it's for. And so far all it's been doing is popping up every time I go to a site that I haven't visited since I installed Certificate Patrol. It would give me a dialogue saying, hey, here's a new certificate that I'm in the process of putting into my cache. And so that's not very informative, although that's how I learned, like, for example, that Facebook is using DigiCert, I think it was. And I looked at them, and their certificates are far less expensive than VeriSign's. And if Facebook's using it, good enough for me, too. So I'll be switching and saving myself a lot of money because I've been using VeriSign from day one.

In this case, Certificate Patrol did what's really cool, which is it noted a change in SSL certificates when I revisited a site that I had already visited. In this case, I don't know what Google's doing, but the old certificate, that is, the prior one that it had, was only issued 28 days ago, and the new one was issued 13 days ago. And what's nice is it provides you with a side-by-side display where you're able to see, sort of like allow your eyes to just scan down and see what parts have been changed.

And everything was the same except of course the serial number, which is a hash that's always going to be different. That was different on the two. And then some MD5s and other stuff toward the bottom were different. But and then of course the issue date and the expiration date and how much longer that the certificates have. And it commented that this certificate was exchanged, even though the prior one wasn't near expiring. So it does some nice little interpretation for you to sort of help you understand what's going on.

So I just wanted to raise it to our listeners because it's the first time I've had a certificate change, and it works. And of course what this does is this alerts you to, if there were a man-in-the-middle attack, if your employer or your school district or somebody were changing certificates on you and using a different cert in order to filter your SSL traffic, this would pick it up. And there's no way you could be fooled because the certificate would change, even if the issued name were the same, for example, if a government was going to play this game, and we talked about a story recently where some governments were trying to use fraudulent certificates, presumably to monitor their citizens, even though they were over SSL connections. So this prevents that, or at least alerts you that something fishy is going on, and then also helps to interpret what it is. So it's very cool.

Leo: Very interesting, yeah.

Steve: And this has got to be the best thing I've seen in a long time. In our Attacks & Breaches section, my subhead was "Endless Sony Breaches."

Leo: Oh, it just won't stop.

Steve: I know. We're at 13 now, different breaches of Sony. And my friend on Twitter, Simon Zerafa, sent me a link this morning that's got to be the funniest and best website I've seen. The URL is hassonybeenhackedthisweek.com. And, I mean, it's a legitimate website. And I went there, and there's a big red "Yes" with a frowny face.

Leo: He probably doesn't have to update this site that often.

Steve: Hassonybeenhackedthisweek.com now exists.

Leo: Then it says "Latest hack," so you can always see when the latest hack is. That is a hoot.

Steve: Yup. And in fact - isn't that great? Hassonybeenhackedthisweek.com. I mean, that's the kind of website you really don't want to have your name in. But Sony does. Attack #13 occurred two days ago on June 6th, which was Monday. And it was Sony's European website for professional broadcasting equipment. The attacker of the website said that he just used standard injection techniques for SQL - now they're standard, Leo. They're not even a big deal. It's like, yeah, yeah.

Leo: We go get them at the SQL injection store.

Steve: Oh, absolutely - in order to access the database where he got information of usernames, plaintext passwords, mobile numbers, and emails for around 120 users - not 35 million, like we've seen recently, but 120 - in another attack Sony had previously experienced. So apparently there was some overlap. The person to blame seems to be a Lebanese hacker known by the name of Idahc. And he is the same one behind the attacks on the Canadian Sony Ericsson site that took place last month, in May. And he was quoted as saying, "Yeah, I was bored, and I play the game of the year, Hacker vs. Sony."

Leo: And now this makes me wonder if it's really Sony's fault at this point because now they're really a target. I mean, it's obviously their fault. But how many of us targeted by this many hackers would have a similar problem, I guess is the question.

Steve: Well, and as I said last week when we were talking about it, it's a matter of hardening. And hardening is hard. And Sony obviously had never hardened their sites. They're learning now. And again, I hope everyone, I hope the industry is paying attention, and chairmen of the board are saying, I mean, making their CIOs prove to them that they're not responsible. SQL is a real problem because it mixes commands in with the data, which makes it very easy to produce back-end database-driven websites. But this is the consequence.

Leo: Well, if you don't sanitize your inputs. I mean, which is very - it's a very straightforward, very well-known thing, and this is kind of shocking. Also sometimes these MySQL injection attacks take advantage of flaws, patched flaws in my SQL, and so you've got to keep MySQL up to date.

Steve: And also sometimes many people are using canned packages, which they drop into an existing server, and they customize it.

Leo: Oh, yeah. These guys are script kiddies that are hacking this. This is all stuff that's well known, and these canned packages are wide open.

Steve: Yup, exactly. There was one more site was breached, which was PBS had their site breached. It caught my eye, I got a kick out of it because PBS themselves said that hackers broke into their website and posted a phony story on the PBS website falsely claiming that deceased rapper Tupac was alive and well and living in New Zealand.

Leo: Of course he is.

Steve: So apparently the group Lulz Security, LulzSec, is believed to have attacked PBS due to them being upset about the WikiLeaks story. So here again it seems that, well, if you want to attack a site, the site's just there waiting for you to do so. Crazy.

And I got three little blurbs from the Twitterverse that I want to share. @andronicus, whose actual name is Andrew Skretvedt - I'm sorry, Andrew. S-k-r-e-t-v-e-d-t.

Leo: Skretvedt.

Steve: Skretvedt, thank you, Leo.

Leo: I don't know. I'm making it up.

Steve: He says, "So I'm staring at 26 USD/BTC." So...

Leo: No.

Steve: ...at the time that he tweeted this, we're now at $26 per bitcoin.

Leo: No.

Steve: He says, "If you're still holding your 50, how does it feel to have $1300 of value?" He said, "I spent mine back at 35 cents.)"

Leo: Do you still have your 50?

Steve: Yeah.

Leo: See, the only reason it's inflating at this rate is because people are using it for

money laundering. It has to be. It has to be. Right? How could it possibly be worth $26 a bitcoin?

Steve: Well, it's scarce. It's hard to make them.

Leo: And it's useless.

Steve: You get government employees knocking on your door wondering if you're growing weed.

Leo: My fingernail clippings are scarce. It doesn't mean they're worth $26 each.

Steve: No, I've got fingernail clippings, too, Leo.

Leo: But mine are...

Steve: Yours are not that scarce.

Leo: I'm just saying scarcity alone - there's no inherent value. I'm very puzzled by this. I think it's money laundering.

Steve: It could be scarce and abandoned. And I've seen some interesting criticism recently, people talking about how this doesn't make economic sense. I mean, to me, all of these arguments sound a little evangelical, I mean, like they've got a cross to bear. All I'm talking about, all we did on the podcast was talk about the very cool technology, and I analyzed it from a crypto standpoint. And it's just bulletproof. I mean, it was beautifully designed for what it is. So, yeah.

Leo: Actually I think this is what's going on. There's speculation going on because there are some people thinking, yeah, maybe I should buy up some bitcoins because it's possible this will become a nongovernmental currency, a valid nongovernmental currency. And down the road, as you said, there is a limited number of bitcoin. This could be a very valuable scarce resource. It's got to be speculation at this point.

Steve: I would agree. I think it has to be speculative.

Leo: Crazy.

Steve: @muoncapture, whose real name is Michael Boleman in Mobile, Alabama, he said, "Tried the MS Sweeper rootkit detector/removal tool mentioned on last SN episode. Doesn't work on a TrueCrypted hard drive." And I meant to bring that up, which is why I

wanted to say it now. Many people have tweeted that fact, that it is not TrueCrypt compatible. And that's of course true because TrueCrypt, remember, has to boot from the special boot sector that it has, which then enables its boot time encryption and then basically installs a short-term driver to get the OS going at which time the OS takes over with a TrueCrypt driver in the OS in a seamless handoff.

But what that means is, and this of course is why you have TrueCrypt, is to the outside world it's just gibberish. It's random, literally pseudorandom noise. So unfortunately, the only way to run it on your system would be to go through the time-intensive process of removing TrueCrypt from the hard drive, then running the scanner on it, and then re-True encrypting it, which I think few people are likely to do. But I did want to acknowledge all the people that have made the comment that, whoops, this thing won't work if you've got your drive encrypted. And that's absolutely the case.

Leo: Hey, here's a...

Steve: Go ahead.

Leo: Just a side note on the Bitcoin story. Today Senator Charles Schumer of New York and Joe Manchin of West Virginia wrote to Attorney General Eric Holder a letter expressing concerns about Silk Road, which is a bitcoin exchange site, and the use of bitcoins to make purchases there. The senators have asked the attorney general and the DEA to shut down Bitcoin.

Steve: Well, they can't.

Leo: They can't.

Steve: No.

Leo: This is interesting. The Senate is now aware of Bitcoin, is a little concerned.

Steve: Yeah.

Leo: What if the government closed the bank accounts associated with Bitcoin?

Steve: Well, they can attack the U.S. domestic exchanges. That they could get because they have a known clear public presence. They cannot do anything about exchanges offshore, outside the U.S. And they certainly can't do anything about the network itself.

Leo: It's a peer-to-peer service. There's no central place to shut it down.

Steve: Right.

Leo: But they could close the membrane that allows bitcoins to get turned into real currency.

Steve: Correct. Well, they could close it domestically.

Leo: Locally, right.

Steve: So it would just be the U.S. that didn't have access to it. Or unless we used a non-U.S. provider to do that. So, yes, technology meets the Senate once again.

Leo: Although in this case they say they feel it's a drug issue, a drug-laundering issue.

Steve: It's just money. I mean, sure, once upon a time the Internet was only porn. That's all it had. Now it's way more than that. So fortunately the Internet survived the porn stage, just as VHS did and DVD did. So, I mean, I just don't blame, I never blame the technology. The technology is neutral. It's ethically and morally neutral. It just provides a capability.

Leo: You're right, yup.

Steve: So people will be people. @xino, who's Joe in Wisconsin, asked, "What was the max range of the portable dog killer?" Now, there's going to be the return of the portable dog killer.

Leo: You're going to make one? You're going to make a new one?

Steve: I have to. I spent a really annoyed afternoon with a friend of mine barbecuing on Sunday with this little yappy dog next door that will not stop. And Mark has done - he's been so patient with the neighbors. He texts them, and then they apologize, but they're not at home, and then they come back, and they bring the dog in. And this has been a year now he's been putting up with this little yappy dog. Now, many people in response to the portable dog killer story - and if anyone listening to this doesn't know what we're talking about, we're not talking about something that kills dogs. That's the whimsical name for something I built when I was 16.

Leo: It does not kill dogs. It annoys them.

Steve: 38 years ago. Yes, it trains them. It's an acoustical trainer. So many people have asked for plans for the portable dog killer. And many people have said, hey, Steve, here's a link to one. Well, I've purchased, out of curiosity, about six of these things and given them to Mark to try to use. And they don't work. They're just useless. So I will be recreating - this is a background project. I don't know what the timeline will be. But I will

document it. I know that it will be based on a 1600-watt tweeter, which ought to do the job. And probably a piece of tuned PVC piping in order to provide aiming. And we'll set up a standing wave in the pipe in order to maximize the effect.

Leo: Oh. Aiming and amplification, mmm.

Steve: And amplification, and directionality. So it'll be fun...

Leo: Will you sell this for bitcoin, if people want to buy it?

Steve: I'm not going to sell it. But I'll fully document the project with plans and everything that I went through, and it'll be on the website at some point in the future.

Leo: You should tweet about that tweeter, I think.

Steve: I saw this, and I thought, okay, well, I ought to tell people what's going on, that there will be a return of the portable dog killer. And anybody who needs a - this is going to be overkill, probably. But if you're going to make something that's loud, you might as well make it as loud as you can. So he's got some problems with birds up in the trees, too. This ought to just blow them right out of the tree, so...

Leo: Geez Louise. We're going to have to get the ASPCA to monitor this podcast pretty soon.

Steve: And I do have a nice bit of SpinRite feedback from Mark Botner, who wrote on June 7th, so just yesterday. He said, "Dear Steve, a co-worker and friend of mine recently told me about Security Now! Episode 291, which is all about the Stuxnet worm. As I listened to more episodes of Security Now!, I figured out that you were indeed the inventor of the Gibson Light Pen, which I fondly recall from my early days of learning to program computers."

He said, "Also as I listened to Security Now! I became curious about SpinRite because I frequently troubleshoot, diagnose, and repair PCs for my family, extended family, and friends. I did not purchase SpinRite immediately but planned to use it the next time I had a problem with a hard disk. Well, as luck would have, shortly after listening to Episode 291 the hard disk in my teenage son's PC started making a strange noise, and the system ran very slowly. Normally I just replace disks at the first instant they start acting unexpectedly because that's usually an early indicator of imminent failure. This time I purchased SpinRite instead and ran it on my son's system in Level 4 mode.

"SpinRite reported no problems, but the system has worked perfectly ever since, and I have been saved the cost of purchasing a new drive. By my calculations, SpinRite has now paid for itself and is essentially free for all my future use. Thanks for a great product. Mark Botner, Little Rock, Arkansas." And he says, "P.S.: I'm now listening to all of the Security Now! podcasts, starting with Episode #1, and am currently up to 64. They are fantastic." So thank you very much, Mark.

Leo: Awesome. All right, Steve. I've got questions for you. Are you ready?

Steve: You betcha.

Leo: Number one comes from Shaul in Israel, a third-year geography student at Ben-Gurion University, and an avid Ubuntu user. He says: Steve, I just watched the last Security Now!. In the next show, can you please talk about how the new quantum computing will affect cryptography and passwords? I think you got this one from Twitter because he's got all the hash tags in it.

Steve: Yup, he tweeted this. And I thought, okay, how can I best describe the problem that quantum computing will represent to cryptography? And the analogy that first popped into my mind was it'll be very similar to what will happen to bank vaults when we have teleportation.

Leo: [Laughing] You're right, Steve. That could be a problem. Beam me in, Scotty.

Steve: Yes.

Leo: So what you're saying, I think, is that teleportation and quantum computing are equally likely in our future.

Steve: Exactly. That's why the analogy works on so many levels. Teleportation, of course, completely destroys the utility of bank vaults. And Fort Knox would be...

Leo: However, highly unlikely.

Steve: Fort Knox would be in trouble, yes. And it's extremely unlikely to happen in any time soon.

Leo: So you're skeptical of the claims people have made for quantum computing.

Steve: Okay. Here's why. The way a true quantum computer would function, if anyone knew how to make it, and if it was of sufficient complexity to be a threat to crypto, is it would instantly be able to instantly try every possible key. That's what a quantum computer does.

Leo: Oh.

Steve: It tries all of them at once.

Leo: It's massively parallel.

Steve: It's the end of crypto as we know it. It's back to smoke signals.

Leo: So just as a recap, of course everybody's listened to your fantastic explanation of crypto going back way back when to early episodes. But modern crypto relies on the difficulty of factoring large primes.

Steve: Yes. And so there are a number of ways a quantum computer could end crypto as we know it. One is that it could simply say, oh, here's the factor of this big prime. One of the ways that crypto works is that, for example, public key crypto relies on that we don't have, no matter, I mean, we've tried and tried and tried and tried, all the best minds in the world have tried, the math guys, to come up with a way of determining what two factors are of a really big prime, and they can't. Or, I mean, what two prime factors are of a large composite, basically to perform a prime factorization. And there's just no solution for it that we've found.

Presumably, a quantum computer, that's one of the first things you ask it, is when you hatch it or grow it or whatever you do, is do this. The other possibility is that you could set up a problem with any kind of a key where it's literally able to brute force the key instantly. It would try all bit combinations at once. That's what a quantum computer can do. So it's the end of life as we, well, crypto life as we know it. And again, we're nowhere near it happening. It needs to have a level of complexity that can handle that. And it's purely theoretical at this point in the first place. And when we start having them, they won't be able to be that complex. And in order to be complex enough, the challenge just scales exponentially.

Leo: You're basically saying it's as likely as an Einstein-Rosen bridge from here to Mars.

Steve: Yeah. It's not something that we need to worry about today. And we'll have plenty of notice.

Leo: It seems to me, maybe I'm wrong, it's like cold fusion. There are people who claim to have kind of done this or something like it. Or is that not the case? Is it purely speculative?

Steve: Well, it's absolutely in the lab, where you've got to wear goggles. And there's cool, like, smoke moving around the floor. And you're in a university where you don't pay your own bills. And, I mean, it's way out. And they're able to say, oh, look...

Leo: There's this company in, I seem to remember, in Vancouver that claims to have made one. I remember, you know, people, I was doing this TV show up there, said, oh, we've got to interview him. And I had the same reaction, I said, well, no, we don't. They say they've sold one to Lockheed Martin.

Steve: As long as they don't give one to the NSA, we're fine.

Leo: Well, I'm sure the Lockheed Martin is just a front for the NSA.

Steve: They sold one to Lockheed Martin. Which may be the reason that they were cracked with RSA SecurID. Actually Lockheed Martin wouldn't need to have SecurID. They'd just crack it instantly.

Leo: Hmm. Moving right along to Question #2. Daniel Summers, also from a Twitter question, @DanJSum in Albuquerque, New Mexico: Your "hash on the client" comment got me thinking. Is there a good way to have secret salt on the client? Would secret matter? You'd better untwist this tweet.

Steve: Yeah. I made just an offhand comment, threw it out there at one point last week, saying that, if when we're logging into websites, the website we're logging into were to locally hash our password, then all of these problems that we've been discussing disappear. That is, the client, the browser, would hash it, turning whatever we use, I mean, it could be "Hi Mom." Actually that's probably a bad example. But, I mean, it could be something very simple. The browser would hash it so that only the hash would go over the line, the wire, over to the server. So the plaintext hash, the plaintext password would never be available to be stolen and so easily reused elsewhere, which is what we've been seeing. And we're complaining that, like, Sony apparently never hashed a password in its life. Sony just stores them all in plaintext. All the ones that get out are in plaintext.

Leo: I think they said they hashed them. Didn't they say they hashed them?

Steve: Well, I mean, hackers are posting plaintext passwords from Sony.

Leo: If it's an unsalted hash, can you solve it with rainbow tables?

Steve: Well, you can solve it with a rainbow table if the hash is up to a certain size. And I think I've seen where at, like, six, or maybe seven characters - because remember, as my own Haystacks page shows, if you're solving for a large character set, the number of possible combinations goes up very fast. And that means the rainbow table size goes up very fast. The whole point of a rainbow table is that it's a sorted list of all the hashes that came out of putting things in. Which means you've got to store them somewhere. And yes, mass storage is coming down in price. But, I mean, we're talking escalation. Add one character, and now you need 95 times as much storage as you had before.

Leo: So it's straightforward, trivial, and should have been done to hash it sufficiently so that you couldn't crack it.

Steve: Yes. I think that what we're seeing is, this is another symptom of us still being in the wild frontier stage of the Internet, where anyone is storing a non-hashed password.

And I wouldn't be at all surprised, with JavaScript as prevalent as it is now, I mean, it's virtually ubiquitous, that it's a great idea to hash the password in the client so that it's always obscured. So the user sees something friendly, and before it goes anywhere it's turned into something that looks like pure gibberish, all high entropy, and then that's what gets sent to the server.

Leo: Sergey Romanov, whom one would think is one of the last members of the Russian royalty, but in fact lives in Minneapolis, suggests he's got a solution - he must have sent this before last week - that he knows what your password secret is. I got it. He says: I think I already know the solution for stronger passwords that you are going to talk about next week, and I agree this will change many passwords of Security Now! listeners. I've been listening for more than a year now. And just as you are, I like assembly language. I used to practice my programming skills many years ago on my 8088 processor, an MK-88 with 256K of RAM. He is Russian. He says that's the Belarus version of the IBM PC is the MK-88. Wow.

Since then, with the invention of - I should read like this [Dracula accent]. Since then, with invention of virtual memory addressing in 386 and advancement of scripting languages, I've lost interest in programming as a profession. So he says - here's his technique. He says: Let me put it in one sentence. Use the whole range, all 255 ASCII characters when creating your passwords. For example, in a Windows-based PC, do so by pressing ALT + 137. That'll give you a . This way, instead of entropy limited only to the number of characters available on the keyboard, which is roughly 26+26+10+32, those are the typeable characters - 26 without any modifier keys, then shift, alt, and so forth. In other words, 94 keys. One can use all 8 possible bits of 255 ASCII character range and thus achieve a maximum entropy per character entered.

Now, I don't know how you - on the Mac it's very different. You can't use ALT 137. You have to press option, shift, and it's a modifier key combination. So he says: Try to guess, for instance, }%, I don't know what that is, a little angstrom mark, then another , then an , then an em-dash, then a <, and then a cross, and then a smart left quote - anyway, you get the idea. First one seems to be harder, don't you agree? But using this password scheme represents a small difficulty for smart phones - yeah - and Linux because ASCII characters are not simply typed using such operating systems, but I'm sure apps are written to do it. Did I guess right, Steve? If I did, can I have a copy of SpinRite signed by the author? Signed, Sergey Romanov, Minneapolis.

Steve: So actually I noticed that, in his little sample, I wonder if he's trying to claim it as a registered trademark because he's got the last character in there is a registration, the circle R. Okay. So the problem, first of all, yes. If you had an operating system that made it practical to enter those - and I would go even further and say Unicode, which is 16 bits, more even than 8 bits, if there were some way to do that. And if the web form you were entering it into accepted it, and if the website at the other end accepted it...

Leo: Oh, that's a bigger if.

Steve: And that's the big if. I mean, we're seeing - I've had a lot of feedback from people saying, gee, Steve, I tried to use the haystacks approach and put one of each type of character in explicitly, and nothing I could do worked. And so the big problem is

that the recipients of - well, there's two big ones. The recipients need to be able to handle any kind of character you could throw at them. But also you need to be able to input it reliably. I mean, yes, we all know, all us old IBMers, the ALT something. I mean, I use ALT 7 to give me a bullet on my PC all the time. Or is that CTRL 7? No, it's ALT. And, like, 145. Anyway, I have a bunch of those codes memorized that I've used through the years.

Leo: Yeah, me, too, yeah, yeah. There's no standard way to enter those.

Steve: Precisely. And so if you're anywhere else, on a different device or in a different location where you don't have access to what you entered before, you're out of luck. So I think it's very important - and this is the reason my own Haystacks page only focused on that base set of 95 characters, because I counted space also, 33 special cases. I don't know whether space is an alphabetic or a symbol. But anyway, I counted it there. The only reason I did that was to say, okay, here's the lowest common denominator. Sure, if your particular use case would allow you to use a wacky character, and that means that the recipient of it understands the wackiness also, by all means go for it. I mean, that instantly means that you're off the reservation completely, and that thing is never going to be found. The downside is you're in trouble if you need to log in anywhere else that may make it much more difficult to enter those wacky characters.

Leo: Well, nice try. But no SpinRite for you.

Steve: Not quite yet.

Leo: Thank you, Sergey. Jerod Lycett in Duncannon, Pennsylvania asks: Please keep Perfect Passwords. It's still useful, even if it's not passwords. He says: There is a use for the Perfect Password page that Steve has at GRC.com. I don't think you've taken it down, hope you haven't taken it down.

Steve: I won't.

Leo: No. He says you may have overlooked this, but I use it for salt. If you use SHA512 10,000 times, it still won't prevent a determined hacker. The important thing is, of course, an unguessable salt. So he's saying it's one thing to hash it, but another thing to hash it and have a really strong salt key. If you use a known cleartext such as "turtle," as a password, and then gain access to the hashes, you'd simply have to figure out how many times they hashed, what the salt is. If the number of times they hash is known - and companies for some reason like to brag about how many times they hash, big mistake - then all you have to do is run an attack to find out the salt. I use Perfect Passwords to generate the salt for everything I do. Even if you plan on removing it, please leave the salt generator up for everyone.

Steve: And the page will stay, GRC.com/passwords.

Leo: Thank you. No, I would keep that up. That's a useful page. Not everybody wants to use Haystacks, for a variety of reasons. Question #5, Dave Anderson, Grass Valley, California. He says: USB prophylactic achieved.

Steve: Only on this show, Leo, do we have USB prophylaxis.

Leo: This is in response to the listener who was concerned about using a flash drive after it was connected to an infected computer. And we said, yeah, that's kind of risky because flash drives could be written to. He said: This got me wondering if there are any USB flash drives with write-protect switches. Turns out there are, though not in great proliferation. I found this site which seems to be trying to collect a list of them. Fencepost.net has the post. I'll put a link in our show notes. Seems like a great tool for the purpose of having a maintainable toolkit which is protected from infections and fits in your pocket, unlike a CD or DVD. And actually I meant to ask you about this when you brought this up. We said use a CD because that's for sure write-only, I mean read-only. But does the hardware protection on a flash drive, is that reliable? Can that be overwritten by software?

Steve: No, it is enforced at the hardware level inside the drive. And for any listeners who don't have ready access, I checked. And if you Google "USB flash drives with hardware write-protection," which is the tail of this URL, USB flash drives with hardware write protection, the first link that Google finds is this post. And it is a nice page that by make and model runs through, I mean, that's what it is, obviously, is a page listing all that. So I just thought that might be of use to our listeners. However, I then encountered something very cool which is - it's our last question of the podcast, and it's one of the two Hot Tips of the Week, which I just completely overlooked.

Leo: Ahh. Well, we'll get to that in a moment.

Steve: We will.

Leo: Yes, yes. Question #6 from Lynne in Maine. Lynne has thought about the best passwords ever: Steve, first the pleasantries. I've been listening to Security Now! for several years and have gradually ratcheted up my security per your suggestions - LastPass, NoScript, Certificate Patrol, to name a few. I really appreciate your going into the nuts and bolts of how things work so I can make informed decisions about security. Now, I just finished listening to 302 and the password revelation which you would be sharing with us next week. I am now very curious if it's at all like what I have been doing for years. I'm a network engineer for a major cable ISP, and I'm responsible for coming up with the enable passwords for routers, switches, et cetera.

Here's what I do: First I come up with an 8-12 word phrase I can easily remember. Then I take the first letters of the words of that phrase. Then I make various substitutions such as zero for the letter "O," 3 for "e," $ for "S," et cetera. Then I mix up the capitalization. Then I add punctuation. This has the advantage of being memorable, although some of my co-workers may disagree. But I dare any brute-force attack to succeed in anything like a reasonable amount of time. Just curious if

this will be anything like what you've come up with. Can't wait till 303.

And I have to say I actually, and have mentioned before that I use that. I take song lyrics because I can easily remember the song lyric. And then I will use the initials of the first, say, 10 or 12 words of the song lyric, capitalize, punctuate. I don't like using a zero for "O" and a 3 for "E" because that's leetspeak, and it's probably the first thing anybody would try.

Steve: And that was exactly the comment I was going to make, Leo. I didn't mention it last week, and I thought I should because I've run across comments a number of times. And you're right, it is a well-known, those simple sort of visually approximate substitutions are known by attackers and are in their dictionary attack arsenal. So that's the one thing that won't get you a lot more safety. Certainly mixing up capitalization, we know why that works because you have to have an exact match. And adding punctuation, we know why that works because you have to have an exact match. So those are all good things. But I did want to mention not to depend upon changing letters for numbers or symbols, characters that look similar, because that's really not providing the kind of protection that someone would hope.

Leo: Yeah. I use the punctuation. And I have an algorithm which I won't tell you for uppercase/lowercase that is non-obvious. And then I guess you could, if you wanted to put numbers in for letters, if you had your own algorithm, you've replaced all E's with twos, that's not going to be likely guessed. I mean, maybe that's going too far. I don't know.

Steve: I did want to mention that I've updated the Haystack page with a bunch of links at the bottom of password-related web pages that I think anyone who's interested in passwords will find interesting. I've found, I mean, you could imagine, since last week's podcast I've had a whole ton of interest. And there have been people posting about GPUs, Graphics Processing Units, that are able to crack through passwords that are too short; a bunch of interesting analysis of the recent password databases that have been breached, things showing percentages of what - that's where I know that your prior, your old favorite password of "monkey" was #14 on one database. Which means that everyone was choosing "monkey" for some reason, which I just...

Leo: It just, I don't know, it's just something about monkeys. Isn't that funny? I mean, I certainly wasn't choosing it because I thought everybody else was.

Steve: No.

Leo: I gave that up, by the way, using that, in 2004. Just so if anybody wants to try to crack my system, it's been some time since I used "monkey."

Javi Harris in Iowa wonders how to get started as a kid: Hey, guys, let me start with this. When I first started listening I had no clue what any of this security stuff was. I almost gave up, but iTunes insisted that I had subscribed to your podcast, so with a great deal of patience I've slowly started to really understand all the topics you

cover. Javi, well done. That's great. What really got me interested is how easy it is to be safe and even easier to be unsafe. I didn't realize that a simple change of your password could make a huge change. All of this crazy hashing and salting passwords still sometimes makes me dizzy, but I think I have the overall concept. I will be forever a listener, and I hope to learn a lot more. Wow, that is so cool.

Oh, before my question, did I mention I was 16? Now my question: What can we as teens do to make our computers more secure? Yeah, it really is teens that seem to be the people who are getting their parents in trouble most of all, as you've spoken about in the past. We don't have a bunch of money or the resources most adults do. Do you have anything I could start looking into to learn more about security? As the future to computer security I think it would be nice to know what we can do, my generation can do.

Well, I'll let you guys get back to saving the world. Before I do, one last thought. I realize I've been going on for a while. Maybe you could have a little segment on the show where you teach something new to beginner security experts or something along those lines. Just an idea. Thank you so much for having an amazing show, and keep on getting the word out to people. I've changed my perspective on Internet security, and I hope many more kids like me can, too. See you guys. Wow, what a great letter.

Steve: So, okay. Two things. The bad news, Javi, is behavior is the biggest problem with security. In today's world, it is unsafe behavior which gets people into trouble more than anything else. I mean, we'd like to have some technological solution, you could download something and install it and run it and so forth. And if we had that, that would be great. That doesn't exist. It's behavior. Which I know is like the last thing you want to hear because you want to do what you want to do. But unfortunately, it's doing that which causes the problems. So clicking on links, questionable stuff, unknown stuff that your friends send you. Or going to areas of the Internet that are perhaps unsavory and not as safe. People who stay with Google and Amazon and Yahoo! and MSNBC and CNET, they're not typically getting themselves in trouble.

Leo: And they're not teenagers, either.

Steve: And they're not teenagers. Exactly. So there does seem to be, like, a demographic sort of bias in that direction. So the first thing I could say is unfortunately there aren't any shortcuts. And the thing you want to hear the least is behavior is really the way to be safe, basically not do some of the things that you're currently having fun doing with the Internet, which might not be safe. And the second is, relative to segments to teach beginners, I don't know when you got hip to the podcast, but the very start of the podcast, five and a half years ago, Episode #1 and moving forward, we spent a lot of time laying down some foundational stuff, which if you haven't heard it, frankly, I recommend doing that more than anything else, is go back to Episode #1 and work your way at whatever speed you're comfortable with forward because we really have covered a huge amount of content in a very carefully, this builds on that, that builds on this, that builds on that.

So, for example, when I'm talking about hashing and salt, I use the term now assuming our listeners have enough familiarity with it to be comfortable with it. But there was a podcast where we did nothing but talk about that. I mean, really carefully, what is a

hash, what is salt. So all that still exists, and it's available. GRC.com/sn for Security Now! will take you to that page where the entire archive is there. And Leo, I think people can get it on TWiT.tv, too; right?

Leo: Yeah, what we do, just so people know, is if you go to TWiT.tv/sn, every show is there. It's kind of a pain. We're redesigning the website, by the way. And roughly the same time the new studio opens we will have a new website that'll be much easier to navigate. But right now you can go previous, previous, previous. But if you know our naming convention for episodes, it's really easy. It's TWiT.tv/sn and then the episode number. So #1 is sn1, #2 is sn2, #3 is sn3. So you can go back to any arbitrary episode, even start with #1 and go forward, if you know that naming convention. I apologize. You shouldn't have to know a URL naming convention to find stuff.

And the truth is you can go - there's search, and you can go back and stuff. But it's all there. We also - you have the transcripts, which I think are great, at GRC.com. And we also have show notes on our TWiT wiki. Almost always, most of the time, I don't know how far back those show notes go, but for current shows anyway I'll make sure that your notes get entered into our wiki.twit.tv, so you can search for any show note there, as well. So that's another place.

And actually there's a link, if you look at the show notes, there's always a link to the transcript and other information there. So that's another place to go, wiki.twit.tv. And if you are a listener, and you find that useful, and you want to help out, please, don't hesitate to sign up for an account. It's free on the wiki. We do ask you to attach an email address to your name before you start editing, just so we know who you are. But everybody is encouraged to edit this. This is a community project, the wiki, and I think very useful.

Returning to our questions - by the way, I'm just - thank you. It's great to know you're listening, Javi. And keep up the good work. The truth is, if one in a hundred 16 year olds listen and then you take it on yourself to spread this information among your peer group, this would be a much safer place. You can be the leader here, Javi.

Thomas Kingston in Longmont, Colorado. He said he loved last week's Haystack episode, your new password technology. I've been listening to Security Now! since Episode #1. Crazy, isn't it. And I must say Episode 303, Password Haystacks, is by far my favorite. While I agree 100 percent that password length plus a combination of all four character types - lower and uppercase, number, and special character - in a padded style is far superior in password strength than sheer entropy, there do appear to be some examples in which your Brute Force Password "Search Space" Calculator don't match up with this logic.

For instance, many websites will give you a limit on the number of characters your password can be. For example, a website that allows 32 characters, filling in 32 a's in your calculator yields an offline attack scenario of 6.29 trillion trillion centuries. Right? Seems unreasonable to me. So I figured it best to pass along the finding - oh, I guess because if you were doing a brute-force, all lowercase a's might be one of the very first things you do. And you didn't take that into account, I guess. So he said I figured I'd pass along the findings so people can be encouraged to use something like this after seeing the calculator's results. So in case people are encouraged. Anyway, great job as always, and I really appreciate all that you and Leo do. Thom. And I think you were very clear not to use the calculator as a way to

test passwords. It's not a password test.

Steve: Right. There has been some confusion. And immediately after, right below the calculator, I say what this is not. And it is not a password strength meter. And it can easily, because it can so easily be mistaken for one, I wanted to make it very clear. Then I drew the example that the word "password" the little Haystack meter ranks as very good, but of course it's very bad. So this wasn't attempting to analyze the word for its probability of being in a dictionary or anything. It was just to say, what character set does the word occupy? And based on its length, how long would it take a brute-force search to find it?

And the common theme that I've received in feedback from people who took exception to this whole notion is they've said, wait a minute, but you could design other brute-force attacks which attack padding. And it's like, absolutely, I mean, and their point was there is no brute-force strategy for pure entropy. And they're very right about that, too. If your password is 24 characters of gibberish that's really random, there is no strategy for finding it. And so the purists were arguing that any padding that involves any pattern necessarily means that there would be a strategy for finding it. And I don't discount that, either.

So really the core takeaway was to appreciate that length really does matter in a brute-force search. So you wouldn't want to take, you couldn't securely take something really weak and add padding that was like obvious padding and feel super comfortable with that. I would say take a password like you're already using and just extend it because you lose nothing by making it longer. And even if it's simple padding, that's easy to remember. You don't have to pad, or the whole password doesn't have to be crazy complex.

So that was really the point was, yes, pure entropy has value because there's no - you cannot design a strategy to crack it. Any padding, any lowering of entropy, means by definition that there could be a cracking strategy designed to find it. I'm not sure that's such a problem because don't use all dots, don't use all a's, be clever, come up with something. And Leo, just as you're not telling us what your capitalization strategy is, don't share what your padding strategy is. Don't rely on the padding. But it's an inexpensive bonus for making existing passwords stronger.

Leo: Yeah, so if you go to a site that allows 32-character passwords, and your normal nominal password would be 12, then you just add 18 a's, and that would be good.

Steve: Yes.

Leo: That's not going to show...

Steve: Because it's going to be as good, absolutely as good as what you had before, and inexpensive to pad, exactly, and make it stronger.

Leo: Brian Drake in Gallatin, Tennessee also wonders about Password Haystacks. He says: As someone who has been using leetspeak - oh, we were just talking about that - for a couple of years now to create passwords, I was happy to learn by using your Haystack tool that this yields passwords that would take 1.66 hundred centuries to crack. Problem I've run into using this method, however, is that a disturbing number of websites simply will not allow you to use anything other than numbers and letters for your password, and many of these same sites have a restriction on how long your password can be. This kind of makes it difficult to use your method. And in some cases there's no choice in using a different company if you discover that they only allow you to have weak passwords. For instance, my university. I'm not going to change schools even though they allow weak passwords only.

Do you have any suggestions as to what one can do in those cases? Or how we can get organizations to adjust their password requirements? And while we're at it, would it be too much trouble to get people to put in the password requirements, put them beside every box where you have to enter them in? It's a pain to punch in a password and have the system reject it for some unspecified reason so you keep trying it until you realize it doesn't like the special characters you've been typing in. That's a pet peeve of mine, too. I hate that. It happens to me all the time because, if you use strong passwords, you will frequently run into sites that say, oh, by the way, you did this, or you did - that's annoying. Tell me upfront, please.

Steve: Yeah. I posted this for that reason. I thought Brian made some very good points. I'm sure all of us who are trying to be strong with our passwords are constantly, just as you are, Leo, running across sites where they don't make it clear upfront. It's only after you give them one, then they say, oh, we forgot to tell you, here's the following criteria for passwords. Well, the fact that they have any is purely customer service. I mean, their reps don't know what that "a" with a circle around it, it's called an "at sign," or "back tick," or circumflex. If you don't know the names of these things, it would be hard to explain it to somebody.

But I don't know, it's just, again, Brian and listeners, we're clearly in a frontier era still. And I think things with huge, high publicity problems like Sony is having, like RSA is having, it's got to be the case that companies are going to be looking at making sure this doesn't happen to them. And putting pressure on the company, when Brian asks is there anything we can do, I would say yeah. If you find yourself stymied by a company that won't let you use a strong password, complain. Everyone's got a support link. And they may blow you off and ignore it. But it's worth doing. If nothing else, you'll have the peace of mind of saying, well, I tried to give them some feedback. Please remove the restrictions from your passwords because I want strong passwords.

Leo: Well, and as you've mentioned in the past, a fixed-length password or a maximum-length password means that they're not hashing their passwords.

Steve: It certainly implies it. We don't know for sure, but it's a good tipoff because, if they're hashing, they don't need to have a length restriction.

Leo: Right. So that's a bad sign.

Steve: Yeah.

Leo: Sony. Let's see. Is it Jonathan next? Yeah, Jonathan Simon is concerned about long but low-entropy passwords: How robust are long but low-entropy passwords against new brute-force attacks that reorder the guessing so that it is not in alphanumeric order but rather in low-entropy to high-entropy order? That is, if the algorithm tries passwords that would, for example, be compressible to smaller passwords or that contain dictionary words, earlier than true high entropy passwords in the Shannon sense. Jonathan.

Steve: Well, I just answered that. I forgot that I had this question here. I wanted to make sure that I covered...

Leo: In other words, aaaaaaaaaaaaaaaa, which is low entropy, but long.

Steve: Yes. I want to make sure that everyone understood that I get it, and that I shared with everyone that I recognize that the gold standard is a long and high-entropy password because you can't brute-force it, nor could there be a theoretical strategy for finding it sooner than doing pure brute force; and that it's absolutely true, if you lower the entropy, then that implies - and he mentioned Shannon, who is of course the famous scientist...

Leo: Claude Shannon, yeah.

Steve: ...that talked about information theory, that if you lower the entropy, then it means that there's something theoretically attackable. So, I mean, I wanted to let everyone know, I wanted to give them the satisfaction of knowing, those who felt that they were correcting me, that I get it. I understand that. But I would still argue, take something good and pad it to make it better.

And I will also say that I'm not going to tease again, like I did two weeks ago, but I had a really good session at Starbucks last Thursday, the morning after I recorded last week's podcast. And it's looking good for I have one more really cool new, really new thing to add to the whole password bag of tricks that everyone's going to get a big kick out of. Again, I need to do some more work on it, so I don't know when. I'm going to drop the issue until I have an announcement. And then we're going to have a lot of fun.

Leo: Cool. Can't wait. Curtis in Sayreville, New Jersey needs a secure hard drive eraser: Steve, I have a few hard drives that I need to erase. I mean erase erase. I've heard there are programs out there that will write over the existing data and do a set amount of passes. What program, if any, do you use or recommend for this? Thanks for an awesome podcast. Curtis.

Steve: It was such an important question, and a quickie, that I just threw it in here, even though it extended the length. We like Darik's Boot And Nuke, also known as its acronym, DBAN. You can download the ISO, burn it to a CD or stick it on a USB drive. You just boot a machine that's got the evil hard drive you want wiped, and it'll do it. I

mean, and it does it really thoroughly. I think it's overkill. I believe, like, three or four pseudorandom passes ought to wipe any hard drive for reasons we've talked about before. I know there are some that do 33, and it's like, oh, boy, okay...

Leo: I think two is probably enough. But anyway, yeah.

Steve: Yeah. The key is, the idea is you can always read back what you just wrote. That's what the hard drive is designed to do. But then if you subtract out the big signal that you know was written, that's going to leave the signal that it overwrote. And so if you subtract it out, then you get the signals that it overwrote, which might have been what was stored before. And that's probably the limit of what we're able to do. So my theory is you record pseudorandom data so that you don't know what it was you recorded, and then do it a second time just so that the top two layers are just absolute noise. And that's going to do a really good job at wiping out anything that was stored magnetically beforehand.

Leo: Just Google DBAN, that's it.

Steve: DBAN.

Leo: Finally, Question #12 comes to us from Mark Hull in Charlotte, North Carolina. It's the Double-Header TIPS of the Week: I've been a listener since the beginning. I just want to comment on the easy way to remember the PIE (which we've replaced PEE with PIE, by the way), Pre-Internet Encryption, acronym. When sending things to the cloud or sky, it's easy to remember "PIE in the sky." In other words, encrypt before you PIE, or you sky. PIE before you sky. Also, when doing troubleshooting on machines that might be infected, I use a USB-to-SD adapter, with the SD's "lock" - most SD flash have lock.

Steve: I think by definition, in the spec, it has to.

Leo: Okay. And so he does it, he sets it to read-only on the SD card, puts it in a card reader. That turns it into a flash key that's write-protected.

Steve: And that's what I loved. I thought that was such a great tip, Leo, because here we're like, oh, I mean, I've got all these thumb drives around. They don't have write-protect on any of them. But I also have SD cards for different things.

Leo: They all have write-protect.

Steve: And every single one has a write-protect on it. So that's a perfect solution. I mean, and now you can get them huge, SD cards. You can put all kinds of stuff on it. But write-protecting an SD card and then putting it in an SD-to-USB adapter, you've got the best of both worlds.

Leo: I'm still wondering if that - so you're saying that's a hardware lock that prevents - you just cannot write. You can't go around it. Software can't end-around it or anything.

Steve: Correct. There is, at the hardware level, is a write-request line in the actual hardware spec which you raise this wire in order to enable writing, and then it latches the data at the connector into the chip. This switch disables that. It breaks that connection so that it just - it is dead to writing. You cannot write to it. Now, I remember back in the antivirus days, there were people that were getting - I mean sneakernet, pre-Internet, when we had floppy disks. The floppy drives had a notch on the side, the larger - was it 8.5 and 5.25? Or just 8-inch, the big 8-inch floppies and the 5.25s. And then we got the 3.5s, and they had the little slider.

It's surprising back then how many write-protect switches back then didn't work. That is, the disks didn't have them. They just had a window that was open or closed. But there was a little micro switch in the drive somewhere that you were depending on to provide you with write protection. And it was - I remember being really surprised the percentage that were broken, and most people never knew because you would only know if you attempted to write to a protected disk, and you succeeded, which is normally not something you do. So the one thing I would say to people is, when you lock your SD, try to write to it, just to make sure the switch really does work.

Leo: Good idea.

Steve: So a little double-check.

Leo: Steve, we are done, 12 questions good and true. And you have answered them all to the best of your ability and knowledge. Congratulations. Do you have a plan for next week?

Steve: I don't. We'll see what brings. I've got a list of things I want to get to. Something could happen between now and then. If not, I'll pull something from the list, and we'll have a great podcast. That I can guarantee.

Leo: Steve, thank you for a great show. Everybody should visit Steve's website, GRC.com. That's where you'll find SpinRite, the world's finest hard drive and maintenance utility. You can also find all the previous shows there. You could find Password Haystacks, even Perfect Passwords, everything is there. There's a great menuing system. And if you've got a question for Steve's next Q&A episode, GRC.com/feedback is the place to go. We do this show every Wednesday, 11:00 a.m. Pacific, 2:00 p.m. Eastern at live.twit.tv. That's 1800 UTC. So please stop by and watch the live show, but you can always subscribe after the fact. It's on iTunes, it's on the Zune Marketplace, everywhere podcasts are. Or find it on GRC.com or TWiT.tv/sn. And there we have it. Thank you, Steve.

Steve: Thanks, Leo.

Leo: Great to see you, and we'll see you all next time on Security Now!.