SECURITY NOW!

Transcript of Episode #442

Listener Feedback #183

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-442.mp3
Quarter size (16 kbps) mp3 audio file URL: http://media.GRC.com/sn/sn-442-lq.mp3

SHOW TEASE: It's another Patch Tuesday. Steve Gibson has the latest from Microsoft. And of course we'll answer your questions, 10 great questions, a lot of password conversations and more. Coming up next, on Security Now!.

Leo Laporte: This is Security Now! with Steve Gibson, Episode 442, recorded February 11, 2014: Your questions, Steve's answers, #183.

It's time for Security Now!, the show that protects you, your loved ones, your close personal friends and pets, online and off.

Steve Gibson: And significant others.

Leo: And those. Here he is, Steve Gibson, the king of Security Now! from the Gibson Research Corporation, our security guru.

Steve: That big corporation in the sky, the Gibson Research Corporation. Gibson Research.

Leo: GRC. What was the peak size of GRC?

Steve: 23 people.

Leo: That's as big as TWiT. And you had a GRC building?

Steve: We had a building. We had everyone's cars washed every week. We won't tell your people that.

Leo: What?

Steve: Yeah. Because I was having mine washed, and I thought, well, why should I have this, and they shouldn't't?

Leo: That's kind of cool.

Steve: Our guy came out and spent the entire day washing everyone's cars on Friday. That way they were all clean for the weekend, for social events and things. It didn't go very well because people began bringing other people's cars to have them washed. It's like, okay. It's one of the lessons that I learned about giving people too much freedom, they will take it.

Leo: It is the strangest thing. We've had a little bit of that experience, as well. All of our employees are really great. But I think what happens, I'm convinced what happens is they forget that it's a person. It becomes a company, and nobody has any qualms about taking advantage of a company. Myself included.

Steve: Yes. That, of course, is the problem with corporations that inherently lack a conscience. If the people running the company feel that their obligation is to the shareholders and have no conscience, no corporate conscience - and this is, of course, where Google's "We will do no evil" came from. It was their deliberate attempt to say we're not going down that path. We've seen it time and time again, and we're not doing that.

Leo: Well, we're glad that you have pared it down to, what, three people now. Two people.

Steve: Yes, Greg, Sue, and myself. And as long as I keep them, I'm in business. I can't do this by myself because I just - the IRS would come and drag me off. They'd say, okay, Steve.

Leo: I'm reading a book about Jeff Bezos's business philosophies and stuff - it's a fascinating book - the founder and CEO of Amazon. And, very famously, he considers one of the core corporate values frugality, I guess that's the word, which he got from Sam Walton, reading Sam Walton's book. So in the early days of - forget washing cars. In the early days of Amazon, they wouldn't even pay for your parking. You had to pay for your parking in the Amazon parking structure.

Steve: Ooh, boy. In the corporate parking structure.

Leo: Yeah. So but now, I think, there's the range right there. Free car washes; you pay for your parking. Right there, that's the range. But that's not what we're here to discuss. This is a Q&A episode. We have lots of questions for you, Steve. But before we do that, we usually like to take a look at what's going on in the world of security.

Steve: Well, yeah. Because this podcast follows last week, I realized this was really, after I looked at the mailbag, this was the post-password principles and policies podcast.

Leo: PPP.

Steve: Yeah, the PPPPP.

Leo: PPPPP.

Steve: Because everybody wanted to talk about password policies. And arguably, that's very important. Passwords are what we're using today as our means for authenticating ourselves on the Internet. As we know, I'm actively working as hard as I can to change that, as are other people. The FIDO Alliance today released the documentation for their work. And I think I did a better job than they have. But we'll let the market sort that out.

This is Patch Tuesday. This is the EFF's Day We Fight Back. This is the Global Safer Internet Day. Actually all on today. It's weird that finally, instead of things happening the day after the podcast, actually they all happened on the day of the podcast. We got some more details about Target's point-of-purchase breach. Comcast is doing something a little chilling and unsettling that we'll talk about where they're turning people's home WiFi routers also into public hotspots.

Leo: Yeah. Do you have to give them permission to do that?

Steve: No. It's been happening. In fact, I don't know that you can even prevent them from doing it. I mean, they're really selling this. They're marketing it hard. I want to talk a little bit about the progress with SQRL. And of course we've got 10 largely password-related questions to discuss today.

Leo: It's a busy, busy day. Leo Laporte, Steve Gibson, Security Now!. We're talking about all the security news. It's been a very big week for stuff.

Steve: Yeah. This is, of course, second Tuesday of February. So Microsoft has been putting together charts for a while. And I just - I looked at this chart for today, and I thought, well, okay, there it is. That's really all we need to know. I've got it here in the show notes, Leo, if you wanted to put it up on the screen.

Leo: I shall.

Steve: This is where they're doing this thing we've talked about before, their so-called "deployment priority," where they say, we realize that there's some burden associated with installing patches, typically aimed at the corporate overworked IT department. These are the ones that you really don't want to wait on. For example, they are remote code execution. That's at the very far end on the right there, it says "RCE," that's Remote Code Execution, which has the so-called "maximum impact." And of course you could remove "max" because, in fact, that's what these things do is remote code execution. They're not going to do anything else.

So there are three patches, one that's in the Direct2D component, one in Internet Explorer, and one in VBScript, of all things. They were all disclosed privately to Microsoft. They're all critical. And those are the patches you want to deploy first. Then they had some in the middle range, so-called "important" patches, that were like, okay, still you want to do it, but you're not going to be in huge danger if you don't. Three patches, one for XML, one for Forefront, which is their sort of firewall network appliance product, and then .NET also, which are information disclosure, remote code execution, and elevation of privilege impacts, respectively, so not such a big deal.

Also I should mention that the other thing that we've talked is the exploitability index, which is Microsoft's own appraisal of how exploitable this is, like, okay, do you have to stand on one foot and do crazy things at the same time to make this thing work? Or does it look like bad guys are going to be able to do it? And pretty much they've either got ones or threes here. Red is yes, this is exploitable; green is, eh, it's really unlikely to happen, but let's get this fixed anyway. So not a huge amount of news. A total of seven different bulletins covering these patches, critical down through important.

Leo: Now, here's the interesting question. How many of these are XP?

Steve: Actually, they all run back to XP SP3. As I was running through them, I thought, yeah, it goes all the way back there.

Leo: So you've got a couple more months.

Steve: Fifty-five days, my friend. Not that I'm counting; but, yes, 55 days.

Leo: The last Patch Tuesday will be April 8th, and that will be the last Patch Tuesday. And what people have pointed out, you've pointed out is that, from then on, future patches are essentially beacons, signals to hackers: Hey, here's something that's not going to be patched in XP.

Steve: Yup, yup. We have a couple questions in today's mailbag, in our Q&A, about the impact of XP being cut loose, essentially, so I won't step on that yet. We will talk about that when we get to it.

Leo: But I think for the next couple of months you're going to want to pay attention because it's going to give you some idea of what we can expect because, well, there's three, no, five critical exploits. I thought only three of the five were XP, but it doesn't really matter, it's more than none.

Steve: Oh, yeah, yeah, yeah.

Leo: XP's not fixed. It's not done.

Steve: You're right. It may not be all of them. I remember seeing that the...

Leo: The first two were not, and then I think subsequent ones were.

Steve: ...high priority ones are XP, the remote code execution. So I don't really have much to say about Safer Internet Day except that there is a site, www.SaferInternet.org.

Leo: Who's not against that? I mean, who's not for that? I mean...

Steve: I think it sounds like a good idea.

Leo: Good idea to me.

Steve: And I looked at Google. Google's there. If you just bring up the Google home page today there's a little link there to Safer Internet. And it takes you to Google.com/safetycenter and suggests that you flush your browser history. It's like...

Leo: Really? That's how we stay safe, huh?

Steve: Yeah, I was wondering about that, too. It's like, if you've got Android, do this. If you've got a browser, do that, and if you've got so and so. So four little things, different colors, and pretty icons. It's like, okay. So I guess this is just Internet Safety Awareness Day, and it happened to be on February 11th, podcast day. So there you go. And it is also the EFF's Day We Fight Back against mass surveillance.

Leo: Now, this is the same - this is Aaron Swartz's, what it is, Progress Now, Democracy Now, Progress something, anyway [DemandProgress.org]. And Reddit, Alexis Ohanian and Reddit. But remember, two years ago we did this. Everybody's sites went black.

Steve: Now, that was SOPA.

Leo: That was SOPA and PIPA. Sites went black. We went black and white for the day. You don't see that kind of mass participation this year.

Steve: I think there's some fatigue, actually, on the part of people.

Leo: We've gone to that well that once too often.

Steve: Yeah. So anyway, EFF is reporting that, thanks to their work, 5,000 people an hour are calling Congress. You can put your phone number into a page they provide. Now, as I understand it, they call you back and give you a script to read so that - exactly, right there on the screen, you're showing it. And so they really do make it pretty easy to make your annoyance known. If you're annoyed, then this is an opportunity to focus all of this annoyance on one day and pretty much overwhelm Congress and show them that you're really not happy.

Leo: You can see the stats on that page, 55,000 calls were placed, 116,000 emails. Emails are worthless, by the way, in my opinion. You really want to call. It's easy to call.

Steve: Oh, I agree. An email's just going to get filtered off. Someone's not - it's like, okay, how many did we get? Oh, that's nice. But they're not really...

Leo: They can't, for instance, validate that it was from a constituent. So I've always been told by legislators we want snail mail because we look at the postmark, or we want you to call us. Anyway, it's easy, and it's free to do, so do it.

Steve: Yeah.

Leo: Who's for Internet surveillance, after all. Not I.

Steve: Exactly. Exactly. Now, Comcast came up with a weird idea. And I don't know whose this was, how this happened. But they decided that they themselves would take it on to essentially create WiFi Everywhere. And I've got two diagrams that I assembled that are there in the show notes which are really interesting. One is a close-up of the L.A. metropolitan area showing Comcast hotspot availability created by Comcast-provided routers in residential settings.

Leo: Hey, good news. You're helping.

Steve: Exactly.

Leo: Comcast customers.

Steve: As this has come to light, individual people are saying, wait a minute. You're telling me that I've got people I don't know connecting to my home router...

Leo: Yes, you do.

Steve: ...without my knowledge or permission, behind my back? And the answer is uh-huh, yes.

Leo: Now, I can give you a little background if you want.

Steve: Yeah, please.

Leo: So I've been a Comcast customer for a long time. At least, I would say, about a year ago, Comcast released an app that you could put on your phone that would find a nearby free Comcast WiFi hotspot. And you would have to be a Comcast customer to use it. Now, this is, by the way, the benefit to Comcast. This is a customer benefit. And it turned out, when I first launched it, they were all over Petaluma. And I said, well, where are these - they're not in businesses. Where are they doing this from? So they've been doing this for a while. And they've really, as you can see, they've really increased the map.

Steve: Ooh, baby, yeah.

Leo: And it isn't altruistic. These aren't free WiFi hotspots for everybody. You have to be a Comcast customer.

Steve: Right.

Leo: It's very frustrating to me. I think AT&T is doing something similar.

Steve: And so they say on their page: "Over 500,000 hotspots. Find one near you." And it's funny, when I put my zip code in, because my neighborhood is all Cox Cable, there are none near me. But if you scroll, like, northwest, suddenly it's, wham. I mean, there's like a dividing line at a freeway. And on one side of the so-called Newport Freeway, the 55, it's like all red. It's like, okay, that's clearly Comcast territory. And then they also say - they have an animated Flash thing on their page, very pretty looking. Graphic design is nice. But I stopped it and had to go back and freeze it so I could copy the text off it.

It says, "With XFINITY WiFi Home Hotspot," so that's what they're calling it. They're also calling it "CableWiFi." "With XFINITY WiFi Home Hotspot, you'll have two WiFi networks - one for you, and one for your guests." Well, of course, and everybody else driving around outside your home. "Now, you can give visitors WiFi access in your home without sharing your wireless password." And moreover, we're giving all of our customers access to your residential router, which we have provided.

Leo: Now, it's secure; right? I mean, I don't have to worry about...

Steve: See, that's the problem. We've just been talking about major firmware problems. Remember this backdoor trojan port 32764 which was discovered. I mean, we don't know this is secure. We just know that now people are connecting to - considering this a feature of Comcast that they can find a hotspot nearby. Well, yeah, it might be your home.

Leo: And it doesn't count against your bandwidth, obviously. They know the difference, I would presume.

Steve: True, it absolutely doesn't, yup.

Leo: I guess it should be opt-in. I don't think it should be automatic. It's not opt-in. And in fact, I don't think there's a way to opt-out, is there?

Steve: Nope. They provide this to you, and it's like, here you go. Essentially, they consider this as their cable, their bandwidth, their router. They're providing this router to, the Comcast cable router. And so they're allowing you to use, under their terms, their bandwidth and their appliance. And, oh, by the way, they're going to be using it, too.

Leo: Worst company in America, ladies and gentlemen. By the way, I don't know if you saw this. A guy named Matt Vukas has, I think, very effectively demonstrated that Comcast is now throttling Netflix. Comcast launched recently its own online streaming video service, directly competing with [Netflix], XFINITY online streaming. It happened right after the court overturned the FCC's Net Neutrality regulation.

Steve: They were sitting there waiting.

Leo: Yup. And the way he showed it is very interesting. He used a VPN. Now, normally you'd assume the VPN would suck. But it does have the advantage of hiding from Comcast the kind of traffic going over the VPN. And he found, this is on Comcast, he's getting a bitrate of 235Kb per second, which is, by the way, not enough to watch anything at anything better than 320x240. He did this for five minutes, let it sit there, didn't get better. Buffered a lot. Then he went through his VPN and got 3,000Kb/second, same connection.

Steve: Three megabits.

Leo: Three megabits, even with VPN overhead.

Steve: Wow.

Leo: So that's because Comcast said, in this case, oh, you're using Netflix; in this case couldn't tell. Now, that's just appalling.

Steve: Yeah, it is.

Leo: But he points out it's effectively a monopoly. You don't really have a choice.

Steve: Yes, yes. Exactly. Here, as I said, that map of Comcast hotspots is completely empty anywhere within the region where I am. But all you have to do is scroll the map to the northwest, and wham, there it is. So if you look at the map, you can scroll around and see where there's Comcast as the provider and not. And if there were choice, then you'd expect there would just be random choices being made, and there'd be a mixture. There'd be, like, maybe some variation in population density, but not an all or nothing. And so that absolutely demonstrates the fact that there is no user choice. I have no choice. I'm a Cox Cable subscriber. That's my sole option. And that's the problem. Where these meetings are being held in Congress, and the providers are saying, oh, well, consumers have choice. No, we actually don't have any choice.

Leo: No. In most cases it's a duopoly. You have DSL and cable, and I guess you could use satellite or dial-up, but that's usually not something you'd want to use.

Steve: Yeah. Just ask Elaine how she likes her satellite.

Leo: Now, Father Robert Ballecer on This Week in Enterprise Tech this week talked about this and said you can call Comcast, and they can dial in and disable it on your router, if you want.

Steve: Oh, nice.

Leo: Everybody, please do that. Good luck getting a human at Comcast. But that's the best way to vote.

Steve: So you're able to say "I object to having people connecting to my router. I want this disabled."

Leo: Yeah.

Steve: And so they're able to connect to it and disable it.

Leo: My suspicion is that legally they have the right to do this, so that currently as a customer service they're giving you the chance to turn it off.

Steve: I absolutely guarantee that in the fine print of something you had to check and say yes as part of establishing it, it gives them that right, sure.

Leo: But for the time being for customer relations they'll turn it off. But at any point they could say, no, we can't turn that off. That's built into the router. Wow. It's the worst company in America. Thank you, Comcast.

Steve: We're getting some creeping details. Not creepy details. Well, maybe they're a little creepy. But creeping. Information is slowly coming to light about what was behind the Target point-of-sale terminal breach. And Brian Krebs has been on this and doing some great reporting. He reports that sources close to the investigation said the attackers first broke into the retailer's network, and we now have a date, middle of November, November 15th of last year, 2013, using - and this is the new information - network credentials stolen from Fazio Mechanical Services, a Sharpsburg, Pennsylvania-based provider of refrigeration and HVAC systems.

Fazio's President, Ross Fazio, confirmed that the U.S. Secret Service visited his company's offices in connection with the Target investigation, but said he was not present when the visit occurred. Apparently the VP, Daniel Mitsch, declined to answer questions about the visit. He was there, but he says "I'm not talking." And according to the company's homepage, Brian reports, Fazio Mechanical has also performed refrigeration and HVAC work for specific Trader Joe's, Whole Foods, BJ's Wholesale Club locations in Pennsylvania, Maryland, Ohio, Virginia and West Virginia. So they're a commercial HVAC vendor. And in this case Target was one of their clients.

And Brian wrote, he said: "It's not immediately clear why Target would have given an HVAC company external network access, or why that access would not be cordoned off from Target's payment system network. But according to a cybersecurity expert at a large retailer who asked not to be named because he did not have permission to speak on the record, it is common for large retail operations to have a team that routinely monitors energy consumption and temperatures in stores to save costs, particularly at night, and to alert store managers if temperatures in the stores fluctuate outside of an acceptable range that could prevent customers from shopping at the store."

And then, quoting this unnamed source, the guy said: "To support this solution, vendors need to be able to remote into the system in order to do maintenance - updates, patches, et cetera - or to troubleshoot glitches and connectivity issues with the software. This feeds into the topic of cost savings, with so many solutions in a given organization. And to save on headcount, it is sometimes beneficial to allow a vendor to support versus train or hire extra people." So the idea being that major corporations like Target essentially subcontract out that aspect of maintenance, and in return will give a subcontractor credentials on their network that allows the subcontractor to do the job that would normally be performed in-house. And so, again, we don't exactly understand how this is being done.

Fazio has since released a statement that is divulging more information, saying essentially, pushing back on this somewhat, saying that their contractual relationship, that is, contracts and purchase orders and things, is their connection into Target's network. That's why they've got access to inside of Target's facilities. So it's like a business-to-business relationship as opposed to something specific to connecting to HVAC systems and monitoring. And I saw some other conversation on the 'Net that was saying that, like it or not, Java is the machine-independent technology that a lot of these HVAC systems were built on. And of course we know how that goes.

I should mention that I recently installed Java. Java was complaining that it wasn't able to update itself. And a few weeks ago I just got tired of that, and so I removed all of the Java from this system. And as happens often, there was something I needed to do that needed Java. So I downloaded it cleanly, 7.051 or something, I think it was, or Version 7, Release 51 from Oracle. And what I appreciated was I installed it on my system, and when it was finished, it came up with a dialogue, and it said, "Java is disabled in all browsers." And I thought, what? And sure enough, Oracle is proactively, when you install it, it is not available to browsers. You've got to go into the Java security settings and deliberately turn it on to make it available to browsers.

Leo: That's great.

Steve: It really is. I mean, it's sad that - how many years did you say we've been doing this podcast, Leo?

Leo: Took a little while.

Steve: I mean, a decade. And I got of course the 3 billion - Java is in 3 billion devices, whether you want it there or not. It's like, okay. The path I took was just putting Oracle Java into the search bar of Firefox, and it took me right to the Oracle page. I accepted the license agreement, downloaded the EXE, 27MB of download, and ran it, and it installed this with this notice. So it's like, okay, well, yes, we're beginning to make some progress here. Untold amount of damage has been done, but to the degree that people will be updating - of course, remember, the problem is that old versions of Java didn't have auto update technology, and so they have no way of fixing themselves. And they're sitting in systems all over the place with those vulnerabilities exposed. But for new downloads, for new installations, Java knows better than to make itself available to the browser, which is a huge, huge improvement.

Leo: That's great, yeah.

Steve: And I did just see a story that crossed my path in the last week that was sort of sad, but hardly surprising. And that is, an entire law firm had its entire cache of client files, all of its work product, encrypted by CryptoLocker. Someone in the firm - apparently this was a voicemail message or something. It looked like email that was sent from their voicemail system. An employee clicked on it, didn't realize that installed CryptoLocker on that machine. And then the law firm's network server had a drive letter mapped onto that local system. So CryptoLocker was able to enumerate the drive letters. It went into the server and encrypted all the files. They then waited, for reasons that weren't clear, more than 72 hours and lost the ability to decrypt their files. And that was everything that they had.

Leo: Unbelievable.

Steve: So I mention this only because this really does continue to scare me. Simon Zerafa, our friend and frequent contributor from Wales, sent me a link to yet a newer version of CryptoLocker, which at this point this one was about half seen. I think it's, like, 27 out of the 47 AV tools that are aggregated at VirusTotal spotted it, as opposed to not. But nothing like 42 out of 47. It was a little over half. And those other two that I had downloaded and posted on GRC's malware resource page, at the time I downloaded, one of them was like, four out of 47 were detecting it. And so this just worries me because certainly our audience is aware of it. But I just, boy, this thing is nasty. And the problem is it is making these guys so much money that we're never going to get rid of it.

Leo: Did they pay? I presume they did.

Steve: I don't know if it's possible past the 73 hours.

Leo: It's not supposed to be.

Steve: Yeah. Or 72 hours.

Leo: So they didn't respond quickly?

Steve: I think they got themselves hosed, unfortunately.

Leo: That's terrible. And they didn't have a backup. Come on, guys. Or it was a hot backup.

Steve: It was a hot backup, and it went in and wiped that out, too. Yikes.

Leo: Hey, before - I know you have some other miscellany.

Steve: Yeah.

Leo: But one thing I did want to ask you about, last week NBC - and we've talked a little bit about this, a lot of us...

Steve: Oh, Leo, I know what you're going to say. Go ahead.

Leo: As part of their Olympic coverage on the Nightly News, Richard Engel, their international correspondent...

Steve: Richard I like so much. He does such good reporting. I've followed him through the trenches of the Middle East and, wow. Anyway, go ahead.

Leo: Yeah, you know, obviously he's not a very technical guy. So they hired - this is their first mistake - an antivirus company to set up a test to see how hackable - the premise, and I blame Richard for this, or perhaps his producer, they went in with the premise that, if you use a device in Sochi, you're at risk. And they went in to prove it; right? And this is, of course, the wrong way to do news. You don't do news with the story in your mind and then go out and say let's get the evidence. You're investigating. And this was not an investigation. They brought in a guy from Trend Micro who brought in a Windows machine and a Mac, opened it up - you saw Richard Engel tearing open that Mac box. That was painful.

Steve: Oh, I was just going to say, it was hard to watch because it was just this gorgeous Apple box that we all are aware of, we just lift the lid off and sort of air seeps in, and so it's...

Leo: He tore it open like a FedEx box.

Steve: Oh, my god.

Leo: But that wasn't really the crime. The crime was editing because what they did is they - first of all, they weren't in Sochi. They were in Moscow, a thousand-plus miles away. They went into a coffee bar. He had an Android phone and these two devices. They did not patch any of the devices. What they didn't show is - but we found out later because the Trend Micro consultant felt guilty, I think, or was worried, because he posted a blog post explaining what they'd actually done.

Steve: The security industry went nuts over this because it was just so irresponsible. I mean, it got so much coverage, it freaked everybody out. The message was you go and turn anything on, and it's immediately taken over. I mean, that's what we were told. But the fact is they clicked on links. I mean, they proactively...

Leo: And, by the way, did nothing that you couldn't have done here in the studio.

Steve: Yes, yes.

Leo: They went to bad websites, malicious websites. Neither machine was updated. In fact, and this is not completely clear, the Trend Micro blog post said that the Apple machine had Flash and Java installed. But at the same time he said that they hadn't done any updates or installs on the machines. He actually installed two known bad vectors that don't come on Apple machines for that reason. So he installed Java, installed Flash on that machine; did not do Windows Update or Apple Update. These are out of the box except for putting crap on the machine. Then surfed to malicious websites. And lo and behold, after running...

Steve: And clicked on links and said...

Leo: And ran software. And said okay, okay, okay. They opened Word docs. They did all the stuff. And then, and they didn't show this, this was even more offensive to me, they go into the Android phone, intentionally disable the setting that says you can't download third-party apps, and went and downloaded malware. They didn't show any of this in the edit. They merely showed, look, my Android phone has been hacked. After you intentionally download malware, I would expect that.

Steve: I know.

Leo: So I think the security guy from Trend got cold feet and blogged and revealed what had happened. But it was obvious, if anybody watched the piece, it was a crap piece. NBC has not yet apologized. It is reprehensible. It's poor journalism. It's scare tactics. And it's a lie. And by the way, NBC's owned by Comcast. I don't have to say anymore. The worst company in the United States.

Steve: I am so glad you reminded me because I meant to talk about it today. I mean, it generated a huge backlash in the security industry because everyone looked at it and said, wait a minute. And I was frankly, I mean, I like Richard Engel. But this was, as you said...

Leo: Same as rigging a Pinto to explode. No different. It's a lie. And, sadly, there was a good opportunity to help people understand how you get infected and what not to do. But they didn't take that.

Steve: The only flipside is, if it caused people to leave their stuff at home, that's probably better, too. The takeaway was try not to bring anything with you that you don't actually need. And it's like, well, okay, that's not bad advice.

Leo: I guess. But it's no more dangerous there than it is here.

Steve: That's a very good point. There was nothing about being there that was - I mean, they talked about all these, it was like poised hackers who are, like, peering around the corners and looking at you as you walk down the sidewalk and, like, zeroing in on you. I mean, it really was quite a fright piece.

Leo: It was so disappointing to me. I used to work for NBC. And this is why, by the way, I don't work for mainstream media. They are appalling. Anyway, I knew you probably saw something about that...

Steve: I'm so glad you didn't...

Leo: ...but I thought I'd mention it. All right.

Steve: I'm so glad you didn't forget to bring it up.

Leo: Enough said, yeah.

Steve: So this is completely random, but it came from the podcast mailbag in miscellanea. Phil in South Florida, I saw the subject, it said "Your EV Certs." And I thought, what? It caught my eye. So he sent this on the 3rd of February. He said, "I was using your SSL fingerprint site and noticed that your EV is set to expire in about 10 days. Letting you know so you don't lose any Yabba-Dabbas." So first of all, I really appreciated that. I was watching this date approach, and I could have done it sooner, but I finally found some time.

And I just wanted to say I had another perfect DigiCert experience. I love these guys. And I've had a lot of people in the intervening two years, because EV certs are only allowed to last two years, so it was two years ago February 13th, I think it is. Oh, yeah, because this was on the 3rd, and it was 13th. So two days from now was when, two years ago, I essentially went to - I dropped VeriSign and switched to DigiCert. And oh, my lord, it was wonderful. This was the same way. It's funny, too, because not only have I had email reminding me that certs were expiring, I got a phone call from them saying, you know, you only have five days left. And I said, no, actually, I renewed that cert.

What happened was I had originally - I had separate certificates for www.grc.com and just GRC.com. And then I added media.grc.com when I wanted to go to full HTTPS Everywhere, or STS, Strict Transport Security, so that I would absolutely - so, for example, Google could have it built into Chrome, as it is, that never attempt to contact GRC except over SSL, which is now built into Chrome. So that meant I could no longer use redirects to take people from insecure over to secure. I had to be able to support that directly.

So the certificate I purchased with DigiCert allowed me to have, for the same price, an additional domain name. You also can't do wildcards in EV, so you have to enumerate the domains that you're going to have. So I had GRC.com, www.grc.com, and media.grc.com. Anyway, the media.grc.com was a separate certificate. And so when I renewed my existing EV certs, I amalgamated them so that media.grc.com was now bound into a single cert, which is cooler and the way I should have done it from the beginning, if I had known to do that. So the point was that I was being warned that I hadn't updated media.grc.com, when in fact I had bundled it into the one that I had updated.

So anyway, I just wanted to say, for those who have heard me talk about them before, these are my certificate guys. And I just had another - it took me all of 10 minutes, I'm not kidding, for the entire process. I had the certificate back in my hands, renewed for two years, plus a free month, thank you very much, just because why not? Because they're DigiCert. VeriSign never gave me any months. So I'm just 100% bullish on these guys, DigiCert.com.

SQRL, the Secure Quick Reliable Login project that everybody knows I'm working on. I wanted to talk about what's been happening with the design briefly, which is, as I have mentioned, I am now focused on the user interface. And I understand that, if this isn't obvious to use, simple and straightforward, it's just going to be a dud. It will have been a really intriguing cryptographic exercise, but it's not going to get off the ground unless it is really easy to use. And we designed the technology on the back end to be very powerful. But it wasn't until I started looking at, okay, how do I describe this, in checkboxes and radio buttons and click okay or click cancel format, to end users? And several times now I've had to go back and change the design, the technology, in order to accommodate the user interface. And I've made a number of small changes over the last couple weeks. And essentially what's happening is I am iterating.

And this reminded me of the time when I was working on the longest repeating strings process where I had an idea of how to do this. I wrote the code to find the longest repeating strings in a large corpus. And by the time I was done, and then watched it work, I thought, oh, I know how to do that better now. And so I scrapped it and wrote it again. And that happened, like, three or four times, scrapped it and wrote it again, until actually finally there was a breakthrough in my thinking where it's just like, oh, my goodness. And I have yet to describe this to everyone because immediately after finishing that I switched to something else. I don't remember what it was, and I never did the podcast that I promised about it. But I'm going to have to, if I can figure out how to describe it in the podcast.

So what's interesting to me is that we're seeing that same sort of thing happening where the needs of the user interface is feeding back into the technology, which I thought was done, but turns out it wasn't. And as a consequence, the technology is changing to suit the needs of the user, which is entirely appropriate. I mean, I'm really happy with the way this has happened. Actually this morning at one point I read - I was sort of catching up over in the SQRL development newsgroup. And we had a guy who's been a very useful contributor ask, he said, any estimate on when this might be? Because I was talking about how, once I got everything finished, I'll go back and catch up on the - many of the pages of documentation are, as a consequence, obsolete. They describe the way I thought it was going to turn out in the beginning, which is not how it's turned out.

And he said, "Any estimate on when this might be? I'm currently working on a bachelor thesis evaluating SQRL, and it's quite hard to keep track of all the ongoing discussions. Are there any other major aspects prone to be changed in the near future?" And so then I wrote back, and I said, "On today's podcast I'm going to talk a bit about the nature of iterating over a design. We did this during the development of the Longest Repeated Strings technology, where each iteration substantially improved upon the one preceding, until finally there was actually a breakthrough that was clearly the end of the design."

And it's interesting, too, because I remember when that happened. It was like, okay, now we're done. This cannot be any better. And I think we're there now with SQRL. And I said, though, I said the trouble is - and this is the key. "The trouble is, this is what happens with unbounded development where a higher value is placed upon eventually arriving at the best result possible than is placed upon what are effectively arbitrary deadlines. True creativity isn't something that can be demanded by management, given a timeline budget, and placed onto a PERT chart. As for where we are today? I had no idea when I switched over to thinking about the user interface that it was going to feed backwards and force significant changes to the design of the technology. But that's what makes this entire effort interesting and, I think, worthwhile."

And finally I said, "I think with yesterday's redesign I'm finally happy with the management of the crypto keying," which is what I've changed. So the good news is I am starting to work on the UI. And when people say when's it going to be done, it's like, I don't know. I just - I don't know. It's going to be done when it's done. I did write, I said, "Basically, this is all I'm working on. I'm working on this, eating, sleeping, and maintaining a girlfriend." So that's my life right now.

Leo: And that's why there's no show notes so far. People in the chatroom are wondering where the show notes are for this week.

Steve: Oh, that's right, I forgot to - you're right, I forgot to post them and to link to them, as I normally do.

Leo: Don't worry about it. We'll do it. When I'm doing the ad you could do it.

Steve: And finally, I wanted to share a field report. Matt in Atlanta, who sent this on Friday, February 7th, the subject was "SpinRite is the hero; I get the credit." He said: "Steve, I started listening to Security Now! 1.5 years ago and have been hooked ever since. During one of the episodes, you described how SpinRite worked, and I decided to buy a copy to put it into my IT 'toolbox' on the off chance I ever needed it. Well, tonight was the night. I just took a graduate assistant position, doing IT support for a department in my school, and was setting up a KVM switch for a faculty member. Simple job, right? Computers were shut down, cables were hooked up, and power was restored. But one computer was in a blue-screen-of-death loop, and Windows recovery wasn't working.

I thought for a second, got a smile on my face, and pulled out my SpinRite CD. Off it went on Level 2. After it finished, one sector wasn't recovered, so I crossed my fingers and rebooted the computer. The Windows 7 startup sound never sounded so good. It is finally my chance to say thank you for this awesome product. It will, and should have, come first in line to fix my problems. Thanks." And he said: "Side note, I decided to run SpinRite on the drive a second time, and the one unrecovered sector was good. I presume that was just SpinRite doing its thing."

And that's exactly right. What SpinRite does is, once it finally decides that, no matter how hard it tries, it is absolutely unable to recover any additional data from a sector - and one of the unique aspects of SpinRite, it's able to do partial recovery of sectors. It will then rewrite what it has finally been able to recover back onto that sector, fixing the previously unreadableness of it, making it readable. And so the second time you run it, you see a perfectly readable hard drive. And then things worked that didn't work before. So, yeah, that's part of what it does.

Leo: Speaking of things working, I'm watching John King on CNN. I apologize. It's on a monitor. Apparently his touchscreen isn't working. They're trying to show the vote going on right now in the House over the debt ceiling limit, and they're trying to show the results. And he keeps tapping, it's so funny, he keeps tapping the screen and nothing happens. And he says, well, I guess I can't show you that. So they moved on to an ad. All right, Steve. I've got questions. I know you've been thinking hard about the answers.

Steve: Well, the show notes are now posted.

Leo: Thank you.

Steve: So people can go to GRC.com/sn. I did not update the page, but the format of the numbering is the same. So if you look at the link for the show notes of 441 and just change it to 442, you'll get it. Or GRC.com/sn/sn-442-notes.pdf.

Leo: Thank you, thank you.

Steve: And you'll get them.

Leo: And it's a measure of how much people love these notes, by the way. They were in the chatroom, where's the notes? Where's the notes? I want to see the graphs. I want to see the notes. It's nice that you have that.

Steve: I've been getting lots of great feedback about those.

Leo: Thank you for doing that.

Steve: Glad we're doing that.

Leo: All right.

Steve: Yes, questions.

Leo: Questions und answers von Steven. Question 1 comes from Paul Green, near Boston, Massachusetts. He writes: Steve, I believe the habit of not echoing passwords back - we talked about this. And you know what, I am now going to launch a campaign, before we get into the email. This we talked about last week. And you know what, thank you, Steve. This whole thing of putting dots on the screen instead of the password as you enter it, antiquated, stupid, useless.

Steve: Yup.

Leo: And especially on mobile devices or on your game console. It just gets in the way because it's so hard as it is to enter it correctly. It accomplishes nothing in terms of security. He says that this goes back to the days of printing terminals and terminal "pool" rooms. In the 1960s and '70s, terminals were as expensive as small cars, as in, for example, a Beetle, the VW. Only a big shot had a private terminal. A big shot with no hearing because they were noisy, too. The rest of us walked to a nearby pool room and sat down at an unused terminal. Because these were printing terminals, and because the ability to suppress echoing of characters varied among devices, the usual method of hiding the password was to print a row of Xs or random characters, and backspace the printing mechanism so that, as the user typed his or her password, the terminal printed the password over an obfuscated background. On terminals where printing could be suppressed, the software would simply not echo the password back to the terminal so it would not appear on the paper. But that was a long time ago.

Steve: I thought that was interesting. And I know that you'll remember, and the old-timers among us will remember the term "half duplex."

Leo: Right.

Steve: And "full duplex." The idea was essentially, to cut down on the delay when you type something, a half-duplex system would do local echoing. So when you hit a key, it was immediately connected to the printer and also sent out. But that meant that the far end did not echo that key back to you. And in fact, if it did, if it was set up believing that your terminal was full duplex, but in fact your terminal was half duplex, you'd hit a key, and you'd get a double. You'd get a double key because you'd get the one because your terminal was running in half duplex.

So you'd hit "A." Your terminal would type "A." And then the "A" would go off to the remote machine, and then it, thinking that you were running a full duplex terminal, would echo it for you so that you could see what you were typing, which unfortunately in this case you'd already seen, and you'd get a double "A." So many times people would be typing, and everything they typed came out double. But I thought that was just a cool hack that Paul shared where, if it was time to put in your password, it would print a bunch of - I actually remember pound signs being the character for using to obscure something. So it'd print a bunch of pound signs, then back up. And then you would type your password sort of underneath the pound sign and that way not be seen.

Leo: Here we go. We're going to call in on our terminal, just to get it going here so we can log into GEnie.

Steve: And that's an actual teletype teletype.

Leo: Yup, there you go.

Steve: Remember that rectangular brick of individual slugs?

Leo: You bet. Now, look, here's the login. And there's the login, and it's going to do the...

Steve: Oh, Leo, you're making me nostalgic. Ohhhhh.

Leo: Yeah, AT&T UNIX there, from Bell Labs.

Steve: Some time later...

Leo: Some extensive time later. Oh, my, my, my. The good old days. Yeah, so that's - but that was a long time ago, my friends, like 40 years ago. Can we now get rid of the dots?

Steve: Yeah. I just - I think what people need to understand is that the only thing happening - I had some people respond to my rant about this last week, saying that, oh, no, you're typing into the web browser, and it's secure. I said no, no, no. All that's happening is the form has the password mode set, so it shows dots instead of the actual character shape. But they're there. And there's all kinds of, like, password revealer tools and things you can get that will, like, oh, look, magically it's decrypted it. No, it hasn't. It's just showing the real character rather than pretending it's not there.

Leo: The only argument I guess would be that it's hiding it from somebody looking over your shoulder. But golly, it just doesn't - it doesn't help.

Steve: I have designed the password dialogue for SQRL. And there's an empty field, and then I've got two little words on either side. At the front of it, it says "Clear," if you just want to, like, clear the password and start over. And the other side it says "Show." And so the default, in honor of - I don't want to freak people out. So by default it will do the character obfuscation. But it's quite happy right there where you're typing. If you just want to click on "Show," it'll let them be seen because it matters not at all to security.

Leo: Matters not at all to security. There you go. You heard it from Steve. It's one of those things we just do because we've always done it that way.

Steve: Yup, yup.

Leo: And it would scare people if you didn't. I mean, people would go, wha-a-a-a.

Steve: Exactly.

Leo: I'm not supposed to be able to see this.

Steve: It's like, okay, well, you're typing it. Why can't you make sure you've typed what you thought you typed?

Leo: Somebody should be brave here.

Steve: Just no sense at all.

Leo: Do something better. From Ian W. in Ottawa: Why do you use link shorteners that hide the final URL? Somebody could easily typo-squat a bit.ly link that you provide on the podcast via audio, then point it at their evil site. Surely a Steve-managed system at GRC.com redirecting links would be super easy to implement. Actually, we have one at TWiT, if you want to use it. I know you have to be selective with your time, but in this case are you sure you've struck the right balance between convenience and security? It is, after all, an audio podcast about security. This has always been the issue with URL shorteners is they can be used to obscure where you're going.

Steve: Well, there is nothing I would like more than to write my own, of course. I have the domain, GRC.sc, for shortcut. The problem is I ask, would people rather I spent time on that or worked on finishing SQRL? And once SQRL is done, would they rather I spent time working on that or got back to SpinRite 6.1? And I know everyone would rather have SpinRite 6.1. And the problem is nothing for me is quick because I'm not going to want to write a link shortener. I'm going to want to write THE link shortener.

Leo: That's the problem.

Steve: I mean, like the galaxy's last word in link shortening, whatever that is. Whether I'm going to have comments, and the ability to browse all the other links that have been defined, I mean, I very much want my own, for exactly the reasons that Ian says. But I just haven't been able to get to it. But...

Leo: Mainly we use bit.ly, and bit.ly does offer branded URLs. You see, there's The New York Times. And that's shortened by bit.ly. But I know you wouldn't want to use bit.ly.

Steve: I'm not using somebody else's. It's like, I'm not having YouTube host my videos on GRC. It's like, no, because after one of those plays you get this random stuff that comes up, and it's like taking people to other unrelated videos. So, no.

Leo: Bit.ly's a good company. But okay, I understand. This is...

Steve: And bit.ly, as everyone knows, I use bit.ly. That's the shortener I prefer. And for now that's what I'm going to do. But maybe, if it bubbles to the top of the things I need to do after SpinRite 6.1 is finished, I'd love to write a GRC link shortener, absolutely.

Leo: As I said, I think we have tw.im. Actually, I'd love to get tw.it. I guess we'd have to talk to Silvio Berlusconi to do that. But we have tw.im as a shortener. But we rarely use it.

Steve: Yeah, tw.it, that's a kick. That's wonderful.

Leo: Yeah. If we can get the domain name, and I think we just go to Italy, and we can get it, then it's easy enough to patch. But we use bit.ly, so we don't...

Steve: But I'm sure "tw" is taken. "Tw" has got to be taken, Leo.

Leo: Tw.it, yeah, yeah.

Steve: No, no. Yeah, I mean, "tw" in Italy.

Leo: I wonder. Do we use bit.ly, Bear? Or do we use something else for our domain shortener? I'll have to ask the engineering department. And there is, Kabusi [ph] points out, an open source, well, is it open source? Yeah, it's PHP Scripts, a URL shortener.

Steve: Gag [elaborate gagging].

Leo: I can't imagine Steve really - there you have it. I think that's Steve's vote. He's disappeared. Joseph in Maryland. He's got a question about temporary versus permanent password lockout. Enjoyed Security Now! Password Policy podcast episode. One of the important features you mentioned, password lockout after, let's say, four or 10 unsuccessful attempts. Why do these passwords have to require manual reset, though? Isn't a temporary timeout, like an hour, enough to dissuade automated brute-force attacks? Why do you have to call customer service?

Steve: You know, it's a great question, and it's just a function, I think, of policy. You can reason it out yourself. The presumption is that, if someone has hit the limit of the number of opportunities, that that's not you. That's the whole point. There would be no reason to lock it out if it wasn't you. So the assumption is you will be able to log in within that ceiling of attempts. And if someone can't, it's not you. So then the question is, if that's the case, what would you actually want? Would you want it to expire, the lockout to expire, and make it again available for someone to try again? Or would you like to be notified? And of course that's what's going to happen. You're going to try to log in and not be able to, and then have to call in order to get yourself fixed. And they'll say, oh, well, we locked your account because somebody, apparently not you, was trying to get in.

Leo: So what's the answer to your secret question?

Steve: Exactly. Now might be the time to change your password from "monkey" to something that they're not going to have a chance of guessing.

Leo: Get LastPass. Get LastPass.

Steve: Yeah, exactly. Use lots of gibberish.

Leo: And a password rememberer so you don't - by the way, did you see that Microsoft put out a paper that estimated that about one in five, 20% of all Microsoft account passwords were hacked, were out there on the Internet, not because they were actually hacked, but because people used the same password on Outlook.com and Microsoft.com that they used elsewhere, say your Target account or whatever, your Yahoo! account, those passwords have been leaked out. So they did an estimate, they scanned all the passwords, all those Yahoo! passwords are online and so forth. And they said we think about one in five of all of Microsoft account passwords are now in the hands of hackers.

Steve: So that is showing the rate or the level of reuse, of password reuse.

Leo: Right.

Steve: Yeah, Leo, I just...

Leo: Just get LastPass. Generate a new password for every time you need a password. Yes, it's long and complicated, although I am discovering, as relevant to last week's conversation, a lot of places don't want more than 16 characters. And a lot of them don't allow special characters. So I used to do, my standard for password generation was 20 characters, mix of everything. And in a lot of places that breaks, Comcast included. Oh, that's another Comcast story. There's a bad exploit that allows - if you have a Comcast account, you might want to change your password. Did I mention they're the worst company in America lately? This has been a known exploit for some months. And finally people just released it. They just said, just forget it, you know. And Comcast is not telling people to reset their passwords.

Steve: Right, I did see that go by.

Leo: Yup. We just don't want to pile on Comcast or anything.

Steve: Yeah. There's something I want to talk about. Maybe I'll make some time next week. Because we've been...

Leo: Oh, there they are [trash trucks].

Steve: We've been going - yeah. We've been going through some discussion of passwords relative to the way SQRL's going to operate because there's going to be a user-assignable password, which is one that they use for authenticating to their phone, but also something that we call an "access code," which the system will generate because it absolutely has to be ultra-high entropy. And I don't trust my mom to do that herself. She just doesn't know what that means. And then the question is, should it be upper and lowercase, special characters, digits also, blah blah blah blah blah, the things we've always talked about.

Well, what's interesting is there's a strong argument to be made, and in fact a good friend of mine first voiced this in email and got me thinking about it, and then we were like, right there in the SQRL project at the same time, that when you add a bit, say that you had, just for the sake of discussion, a 32-character alphabet, which is easy to do - 26 alpha plus, what, six digits, and so now you've got 32 characters. Well, we know that that's five bits. That is, you can represent a 32-character alphabet in five bits. Well, so it's five bits of entropy per character. If you want to add a bit, every bit you add requires the character set to be doubled in size, right, because you've got to have characters that you can represent with all of those.

Anyway, the point was that the question is, is it, from a user convenience standpoint, does it make more sense to use a larger alphabet and fewer characters, or a smaller, more convenient alphabet and more characters? And my thinking has come around, in fact, to the second, to the latter case. This access code which SQRL will generate will be all lowercase alpha because that's the easier thing to enter on a mobile device. And that was the other point that my...

Leo: Thank you.

Steve: Yes. That was the other point that my friend John was making, was that getting to special characters is really burdensome on a typical touchscreen. In some cases, at least in iOS, you've got to do two different shifts in order to get to special characters.

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

Steve: And as opposed to lowercase that are just sitting there, asking you to press the button.

Leo: It does reduce entropy, randomness.

Steve: But actually the all-lowercase, that is, 26 characters, is 4.7 bits of entropy per character. And we're in the process of still deciding how many characters we want. But say that it was 16, four groups of four, all lowercase, easily recognizable. You don't have to remove confusing characters like "O" and zero, or lowercase "l" and numeric "1," which often are indistinguishable in some typefaces, and so on. Somebody else mentioned capital "K" and lowercase "k," not very different-looking in some typefaces. And so it's all lowercase, easy to type. It doesn't look as technical with all kinds of curlicues and funky sharp edges pointing out of it like some super-secret passcode. But we want it to be user-friendly.

And so we're going to end up with all lowercase, and just a few more of them. But what's really interesting is not that many more. I think it was, shoot, I did the math a couple days ago, and now I've forgotten. But it was like a few more. We only had to, if we had a maximum entropy, all upper and lowercase, digits, and special characters alphabet, or all lowercase, we only had to add, like, four more characters for the same amount of entropy. And it was just vastly easier to enter that.

Leo: Good. Just make it a little longer. It's easy.

Steve: And of course that's the haystack message, is make it longer. Length is what matters.

Leo: Jason wonders about his Chase Bank password policy. Thanks for the great podcast. Really enjoyed the episode last week on eCommerce retailers' password policy. I'd love to see a similar study performed on online banking password policies, and I'd like to see another test vector added: case-insensitive accepting of passwords. I have an Amazon credit card that's managed through Chase Bank, and I've discovered that it doesn't matter if it's upper or lowercase. They just take it. Doesn't this mean they're either storing my password in the clear or they're modifying my password to remove the case before hashing?

Steve: So, yes. It probably means, I mean, let's give them the benefit of the doubt. All they have to be doing...

Leo: Just simple JavaScript would do it.

Steve: ...is lower, well, they could be lowercasing it in the browser, or they could be lowercasing it when they receive it. But they're probably removing case. And then hopefully they're hashing it in order to create something that is safe. And hopefully they're multiply hashing it and maybe running it a thousand times and using PBKDF2 with a large number of iterations to make it difficult for a bad guy to run through the hash in a forward direction, essentially, in order to determine what the password is. And they're using per-password salt, so they're not able to build a single dictionary that runs on all the passwords in the site. But all they have to do is remove what they want to ignore from the password on input, and then perform the hashing.

And I did say something last week, and I wanted to remind myself, or I wanted to remember to mention. It came up, we were talking about a password being - maybe it was two weeks ago - a password being too similar. Oh, it was Yahoo!. They were changing their Yahoo! password, and Yahoo! said, hey, this password looks too much like the last one. Did that mean that they were storing it in the clear? And my immediate response was, well, yeah, of course. And then someone mentioned, he said, you know, Steve, that typically when you are changing your password, you're rendering your old one first, to get permission to change it to your new one.

So all they have to be doing, they could still be hashing it for storage, but they're just holding the one you've just entered. They hash that to verify you're you. But now they keep that, that you've just entered, and verify that it's you, and then do a comparison to the one you're replacing it with. And if it's too similar, then they say, you know, make it a little more different, please. Just don't change the number of monkeys from one to multiple in your password.

Leo: Right, that makes sense.

Steve: So it was a good point.

Leo: Yeah. Earl J. in Dallas/Fort Worth: Relevant to the Target and others' compromises, how could Chip and PIN prevent a successful attack, if the point-of-sale device is owned? By the way, Chip and PIN is coming, we probably mentioned this before, to the U.S. by the end of next year.

Steve: Yay.

Leo: All the major credit cards are going to start adopting this. They're replacing the swipe method with a method that involves a microprocessor in your credit card that, I would guess, stores the information that's on a swipe strip plus a PIN.

Steve: Yeah. Well, it doesn't really even have to do that. What the EU saw when they adopted Chip and PIN was a reduction in credit card fraud: 80%.

Leo: Yikes.

Steve: Dramatic. It cut out four out of five instances of fraud. And the way to address Earl's question is, well, if the point-of-sale device is owned, what difference does it make? The key is that it's passive versus active. If we have a passive card, as we do now in the U.S., every one in my wallet is passive, then all it is, is when you swipe it, it's saying, "Here's all my information." It's divulging everything it has to give. There it is. With a chip in the card, everything changes because now you query the card, and it never needs to divulge its secret.

If you have a simple mag strip card, when you swipe it, its secret is now completely read by the device. But that's in a passive technology. With a chipped card, there is a secret which is burned into the chip, which never leaves it. And it doesn't relate it. It doesn't give it up. What happens is it's a so-called "challenge/response" paradigm, where a challenge is given to the chip in the card for it to respond to. And then technology in the authentication end - which is probably not on the point-of-sale device. The point-of-sale device receives the challenge. It asks for a challenge, receives the challenge, gives it to the chip in the card. The chip responds. The point-of-sale device sends a response back to a central server that verifies that the card has responded appropriately to the challenge. And then that server says to the point-of-sale device, yes, this is authenticated, go. So it's a substantially more complex, but consequently vastly more secure solution.

And, by the way, that's exactly how SQRL works. SQRL does the same thing. The little QR code on the website or the one that's embedded in the link that you click on, it is a challenge from the web server. Then your little SQRL client responds by signing that. And only if it has the private key corresponding to the public key for that site is the signature valid. So it's the same kind of thing. You can eavesdrop on this conversation, and you lose no security from SQRL login, which is one of the several benefits of it.

Leo: Paul B. in New South Wales, Australia suggests of password retries, he says: Has the view of the forest been obscured by all those darn trees? I fear that in discussion of limiting password retries, the quite critical risk of a trivially implemented denial of service attack has been myopically missed. This came to my notice when I accidentally made three bad tries to log into my bank, due to a persistent bug in Linux's implementation of typewriter mode, as I prefer the use of the Caps Lock key. I then had to contact the bank and was frustratingly required to change my lovely little password.

All this was annoying enough as it was, but then I thought, okay. So all somebody has to do to take down the entire Internet banking function is to write a script to log on successively to random account numbers with four successive wrong passwords. Obviously, this will be readily extensible to more, like 10, if necessary. The damage would be pretty much complete: a deluge to their phone support as well as physical branches; a PR disaster; a dent in the share price, and so a good time to buy. Of course, one would not perform this from home, probably not even via Tor. Only a few bots would be required, hardly an onerous task. How do you defend? Blocking IPs? Well, with whole user blocks NAT-ed, that in itself would take out customers en masse. In short, it seems to me, restricting password retries is, for an enterprise on the Internet, a terrible risk in itself. What about that? Somebody could intentionally block your account easily enough.

Steve: Yup. And I think Paul is absolutely right. It wouldn't be blocking our account, it would be everybody's account. That is, if somebody decided they wanted to cause, essentially, a form of denial of service attack on a banking institution, they could write scripts for a bunch of bots on a botnet to log in and run through every possible account number, and essentially guess the password until the system locks them out, and then go to the next account. And that would shut down all online banking for that banking facility. I mean, he's absolutely right. I don't see any way around that being a major denial of service attack on a financial institution.

Leo: Strikes me that this only works if you use account numbers.

Steve: True. Exactly. You wouldn't be able to use a username and email address and then an account number.

Leo: Right. My bank, for instances, uses, well, it could be a name, could be anything. It uses alphabetics, which would be harder to do.

Steve: Right.

Leo: Craig Naples in Edinburgh, Scotland shares some thoughts about enforced password composition. He says: Listening to the last episode I was struck by the fact that you recommended online retailers should insist on a mix of letters, numbers, and cases in passwords. However, if they're hashing these properly and using end-to-end encryption, how can they know what the entered characters are? I mean, really. Surely it's better to have a system that's password agnostic, apart from minimum length, because otherwise you'd introduce a weakness hackers can exploit to guess the domain size of possible passwords. Insist on at least one number, and most people will include just one, meaning brute force attacks will try and probably succeed by putting just one number in all the positions, for example. I always thought a password system that has no idea what you have entered would be better than one that asks for specific characters, or is this something that is checked for locally as you enter them without the server's knowledge?

Steve: Well, it doesn't really matter where the check is performed. It can be performed in the browser. And in fact, where we see this being performed in the browser is with contemporary password strength meters where, as the user's typing, the JavaScript running in the browser is looking at what they've done and evaluating the strength of their password as they're designing it. And in that case, you've got code running in the browser. My guess, though, is that when they submit it, it probably goes, as sent, to the remote end, at which point the technology there decides where they agree that it's a qualifying password or not.

But I certainly take Craig's point. And this has been something that's been well understood. When specific policies are enforced, what that does is give hackers a leg up because they know, I mean, and we've seen hackers, we absolutely know hackers who are trying to do brute-forcing will absolutely use their knowledge of the specific password policies of a site to design their attacks for that site. It absolutely would make no sense to waste any time at all on passwords, for example, with no digits, if they know that the policy requires one or more. So it is absolutely true that, again, the problem is this is just a fuzzy gray area where, in trying to strengthen passwords and enforce behavior, that same enforced behavior can be leveraged by the bad guys, as Craig suggests, to further typify or characterize the domain that the passwords are going to be operating within. It would be interesting, actually, to have the policy vary as a function of user somehow.

Leo: That's interesting.

Steve: Yeah, I never thought of that before. I'm just saying this is like, yeah...

Leo: Randomize it, yeah.

Steve: Yeah, to, like, maybe take some aspect of their email address and come up with different requirements per user. But then, of course, now you've got a secret that you have to keep. And of course we know that secrets are notoriously impossible to keep, which is why they're never a good thing to require. So, eh, not such a good idea, no.

Leo: Jim Hyslop in Toronto: Include - this is good. He's got a little C-style include in here: #include <long_time_listener_etc.h>. In SN-439 you were lamenting that Microsoft wasn't going to continue applying security patches to XP. While I agree with your assessment that many of the vulnerabilities that are announced affect multiple versions of the Windows OS, and it is possible to apply the patches to all versions of Windows, I can understand why Microsoft decided not to continue doing it to XP. Microsoft's insistence on pushing people toward the latest and greatest is just rooted in a desire to sell more software, and there is a very real and practical reason to minimize the number of versions you need to support. I'm sorry, is NOT just rooted. I'm sorry. I misstated it. Not just rooted in a desire to sell more software. They just don't want to support all these different versions. And I think that's true.

Each version of a software product that you have to support adds to the complexity of software development, configuration management, and testing. The increase in complexity is not linear. I've never taken the time to analyze the added complexity, but it's more like a logarithmic or possibly exponential increase in complexity - I'm starting to really like the tweet questions, you know, the 140-character ones. The human brain has an amazing capacity of wrapping up extremely complex concepts and packing them with a simple label. Do I have to read all this? Is there anything you want in here? Your version management system will track this change, and once it changes [muttering]. Be nice if Microsoft - basically, you get the premise. By the way, he says, I have a copy of SpinRite. The premise is there's good reason. It's complicated to support multiple operating systems, especially after, let's say, 15 years.

Steve: Yeah. And I chose this, not because it was long, but because Jim really does raise a valid point. And that is, we see Microsoft, any company, I mean, and Apple is notorious for this. Apple has upset developers through time by discontinuing technologies which you could argue were still relevant, but Apple just said, no, we're not going to move that forward. And so I do appreciate the fact that, for example, with Windows 7 Microsoft finally said, okay, no more 16-bit support. 16-bit apps, I mean, if you absolutely have to have them, well, we'll give you XP in a virtual machine in Windows 7. But really we don't want to have that code in the underlying OS any longer. So that, I really do understand that. And, yes, at some point I think it does make sense for Microsoft to say, okay, it's been, as Leo, you said 15 years. That's long enough.

Leo: There's a time. There comes a time.

Steve: Yes. I only wish that what they were giving us was value as they move forward. As far as I'm concerned, there's nothing of any use in Windows 8 that Windows 7 has. It's just a disaster. It's like, why force me to go somewhere I don't want to go? It's not like there's great value there, unless I've missed something somewhere, Leo.

Leo: There's improvements.

Steve: Okay.

Leo: But the other issue, which is actually to me the significant issue, which is addressed by our next question, which is there are machines with XP on the

Internet. What is Microsoft's responsibility to the greater Internet? Because those machines are going to become a threat to all of us. And that gets us to the question from Simon Smith, Dublin, Ireland. He says: "I have two servers for educational purposes. One runs XP. It's a pretty old machine, doesn't have a lot of hardware in it. I don't really want to load it with Windows 7. Would keeping it be a security hole in my network? It isn't directly on the Internet. I have a VPN to get into it remotely. Well, hmm. Sounds like it is on the Internet, isn't it. But it would still have some outbound connectivity from time to time. Do I have to purge XP here?

Steve: You know, you hear me urging people to install security patches every week. But those are systems that users are, without complete control, using. And you also just heard me saying, you know, worrying about CryptoLocker because someone is going to click on something that they shouldn't and get themselves infected, and it's not just like, oh, look, I've got a search bar that I don't want on my browser. Now it's oh, my god, all of the files within reach of that machine are irreversibly encrypted without paying ransom, and we'd better take this very, very seriously. I was running GRC's server, until I replaced it only last year, remember, over the holidays last year, I was running Windows 2000. That's what I was running. And there were people, like, [gasping] oh, my god.

Well, there were some rough patches in the beginning where IIS had some problems with directory traversal mistakes and so forth. But no users were surfing on that. It was just a web server and a file server that had filters that prevented anyone from accessing it except my IP block here at home. And it was, as far as I know, absolutely secure. One of the benefits I had of moving up to Server 2008 R2 was that I got all of the new SSL and TLS protocols, because this box was so old that it didn't support any of those things. And so people were complaining that SSL Labs was giving me a D or whatever. I mean, still, arguably, absolutely secure, but not the latest and greatest.

So I really do, I mean, I think Simon's question is very good. I would say it's probably fine. To understand that it's not going to have the latest patches, but if it's being used in a careful and responsible way, I think it's probably fine. I'm looking at my own XP system here in front of me, and I think of, like, over the years, all of the tunings and tweakings and things I have done. The investment that I have in its configuration is irreplaceable. And when I move myself to 7, as I eventually will, it's going to be like, oh, god, I forgot I did this, and I forgot I did that. And it's just, I mean, it's going to take months to bring a Windows 7 machine up to where I have this thing now.

So I could really understand someone making a conscious decision, in a fixed-use, fixed-application environment, just leave it be. Obviously, XP Embedded in point-of-sale terminals, it's having problems. But that's happening while it's still fully supported. So it's not like that's the cause. And it's also not like systems just crumble mysteriously by themselves. It'll be an event which causes a problem. And if those events don't happen, then problems won't be caused. So, yes, machines that users who you don't control are in front of, clicking on links and surfing around the Internet, I would say you really want to stay current with security. Machines running in the back that have a specific application and purpose, if they're stable and happy, and they're not in danger from that kind of exploit, I really think you're going to be fine.

Leo: It would also behoove us to remember this in future: When you install a proprietary operating system, you cannot count on it working forever. And I don't think that that's a reasonable expectation. So consider, if you want to have something that runs decades, you might want to use something that you can guarantee support for, such as an open source operating system.

Steve: Yeah.

Leo: Philip in Central Virginia, he says - this, I think, is our last question.

Steve: Yeah.

Leo: The minimum password length is not the only number that needs to be tracked. The maximum length needs to be included to have a complete picture of security, along with complexity allowed in the password.

Steve: And this is the point you were making.

Leo: That's what I was saying, yeah.

Steve: Yes.

Leo: He also uses LastPass. He says: I've tried to go up above 15 characters and have been rebuffed by limits from some of the same websites listed here. My bank, for instance. Comcast wouldn't let me use 16. My bank wouldn't let me use more than 16. There are websites - this is Leo talking. Back to the email. There are websites not necessarily on the show's list that have a maximum of eight characters. Maximum. Some websites only allow letters and numbers, but no special characters. Until sites allow both longer passwords and special passwords by default, there is a greater chance of being hacked by the dark side. Long-time user of SpinRite, long-time listener. Looking forward to the next version. Philip Taylor.

Steve: Yeah, again, he's absolutely right. This wraps up our coverage of passwords. I think we've pretty much beaten this thing to death. I'm just, as a consequence of where my focus is now, I'm just thinking, oh, let's just abandon this entire model, this ridiculous, lowest common denominator, this is what we've been doing since terminals were typing hash pound signs and then backspacing over them so that no one could see what you were typing, to obscure your typed-in password. And you know, it's funny, Leo, I do remember, like, holding paper printed up to the light and, like, looking through the obviously known obscuring character and looking at what was behind it because you could do it. It wasn't that tricky.

Leo: Not that hard.

Steve: No, not that hard.

Leo: You hacker, you.

Steve: Well, yeah. I just, you know, we just need to abandon this. I'll bet you 10 years from now the outlook is entirely different. We've got SQRL happening. We've got the FIDO Alliance happening. We've got the notion of multifactor. This change happens slowly. But we're just ramping up the pressure on this change. I think we're not that far from it.

Leo: I haven't used a password to log into my SSH servers on my website in years. No password. I just login, leo@leoville.com, and it goes, boom, okay, you're in. You know why? Because it's a much more secure system. I use public key crypto. And so it has my key. It recognizes the key and lets me in. And believe me, that's more secure than any password; right?

Steve: I have the same thing on my VPN, on my OpenVPN servers. I built my own keys, and I've got keys in my clients that I'm roaming with. And it's like, okay. Just solved that problem.

Leo: SSH has that authorized keys database, and you can put authorized keys in. And we further restrict it to IP addresses. So I have to be on an approved IP address using an authorized key, and boom, I'm in. And that actually is the one area where I'd say, where we normally say there's a balance between convenience and security, where it is much more convenient AND much more secure. It's one of the few exceptions to that rule. It's so easy. So easy. And I don't have to worry about the Chinese hackers anymore.

Mr. Steve Gibson, you are the greatest. Go to GRC.com. The show notes are now there, along with 16Kb versions of the audio. A full transcript will be there in a couple of days, thanks to Elaine Farris. Steve provides all that at GRC.com/sn. But there's lots of other things there. Browse around. Steve's health guidelines, yes, and they're good. Steve's password recommendations. Lots of freebies. Lots of free software. And of course his bread and butter, SpinRite, the world's best hard drive maintenance utility. You've got to have it if you've got hard drives.

And Steve will be back here next Tuesday. We do the show at 1:00 p.m. Pacific, 4:00 p.m. Eastern time, 21:00 UTC on TWiT.tv, so you can watch live. But of course he has audio. I have high-quality audio and video at our site, TWiT.tv/sn. And if you subscribe on iTunes or some other netcast client, you'll be able to get every episode, every week, the minute it comes out. Thanks, Steve. Thanks for joining us. Thanks to you for listening, folks. We'll see you next time on Security Now!.

Steve: Thanks, Leo.