SECURITY NOW!

Transcript of Episode #473

Google vs. SHA-1

Description: After we catch up with interesting security news of the past week, Steve and Leo examine Google's surprising, controversial, and unilateral decision to suddenly and significantly deprecate ALL web server certificates signed by SHA-1 that will be valid past 2016 - even though 92% of certificates (with lives of at least two years) signed in January 2014 were SHA-1.

High quality (64 kbps) mp3 audio file URL: http://media.GRC.com/sn/SN-473.mp3
Quarter size (16 kbps) mp3 audio file URL: http://media.GRC.com/sn/sn-473-lq.mp3

SHOW TEASE: It's time for Security Now!. Steve Gibson is here, and he's ready to rip Google a new one. Seems like Google's policy on the expiration of SHA-1 generated certificates is premature folly. We'll find out exactly what Steve's all het up about, next. Stay tuned.

Leo Laporte: This is Security Now! #473, recorded September 16th, 2014: Google vs. SHA-1.

It's time for Security Now!, the show that protects you and your loved ones online, your privacy. This is the guy in charge, our king of security, Mr. Steven "Tiberius" Gibson. And he is at GRC.com, the Gibson Research Corporation, creator of SpinRite, but also a well-known security expert because, not only does he talk about it, he does it. He lives it. He breathes it. Hi, Steve.

Steve Gibson: Hey, Leo. It's great to be with you again. Speaking of which, I got a notice from my ISP, Cogent, who supplies the bandwidth for my T1s, saying that I have been participating in a UDP reflection attack against some poor victim.

Leo: What?

Steve: And, yeah, it turns out that it was - it's my Cisco router at this end of my T1s. It was responding to SNMP queries. And someone found it and was bouncing packets off of it. So of course that was a little minor annoyance. It's like, oh, well, didn't know that was happening. So I turned that off.

Leo: But that's the right thing for Cogent to do, and any ISP.

Steve: Oh, absolutely.

Leo: Because they'll [indiscernible] that traffic coming out of you.

Steve: Yup. Yup. Actually, what happened was the victim, who was being flooded with this, they were - I was receiving spoofed IPs and bouncing SNMP replies to, I mean, unwittingly, of course, to the target, along with lord knows how many other similar open SNMP servers. And so the victim complained to Cogent that an IP within the Cogent block, they had no idea who...

Leo: They don't know it's you, right.

Steve: They don't know it's me. But they said this IP. And then Cogent looked it up and said, oh, that's - oh, my god, he's got 64 IPs? What's he doing with all those?

Leo: So Cogent didn't observe the oddball traffic, I guess because it's not that oddball, it's just an SNMP...

Steve: Well, and remember, that's a T1. I'm basically blowing through a straw and trying to create a whirlwind. I mean, and so it's like [demonstrating].

Leo: Well, that's what's interesting about amplification attacks because they do allow a single person with even dialup bandwidth capabilities to initiate an attack that's sufficient to DDoS even big servers.

Steve: Well, yes. But my point was, I was only able possibly to contribute...

Leo: Yeah, you were a small portion of it.

Steve: ...1.5Mb because they were - the router is here at this end of the T1s. And so I'm sure they, well, lord knows how many other people they had...

Leo: One of a small group.

Steve: One of a small group. Or a small piece of a large group, I think is probably the case.

Leo: Yeah, yeah, yeah. Interesting.

Steve: So we have some interesting news. And today's topic is Google vs. SHA-1. And when I decided to defer discussing this and give it its own episode, I was not completely sure there was enough to talk about. But I am now. It's really interesting. The industry has reacted very negatively and, I mean, with a lot of pushback against Google because, first of all, it was very short notice, and it's quite heavy-handed. So that's the topic that we'll get to after we cover the news.

We're going to talk about Comcast vs. Tor. There was a flurry of is Comcast disconnecting people using Tor? I want to briefly talk - many people asked about Tim Cook's appearance on Charlie Rose and what I thought about what he had to say. LinkedIn was discovered to have made a mistake. There's a bad, just today, pre-KitKat problem with Android. And LastPass has an iOS 8 announcement. We've got a little bit of telephone, and we already talked about iPhone stuff, so, I mean, not telephone, television miscellany, and then a great topic.

Leo: Awesome. A busy day, as always, here at Security Now! Headquarters. And onward we go.

Steve: Okay. Now, first...

Leo: Yes.

Steve: Yesterday's Triangulation.

Leo: Oh, did you enjoy that?

Steve: Oh, my goodness, Leo.

Leo: Isn't he great? I love Lawrence Krauss.

Steve: I wanted to make sure all of our listeners knew of this fabulous Dr. Lawrence Krauss interview you did. He was just delightful.

Leo: Yeah, I agree.

Steve: I mean, a lot of energy. Very literate. He's a physicist and a cosmologist. And, but, I mean, just compelling.

Leo: He's, you know, it's funny, because he's one of the best science writers out there today. He's written a number of great books. And my son Henry is a huge fan of his.

Steve: No kidding.

Leo: Yeah, because he got Henry into physics. He read his books, and Eric Greene's books, and fell in love with physics. And that's where Lawrence Krauss deserves a lot of credit. Of course he's an advanced experimental physics theorist. He works on dark energy. But a great communicator, yeah.

Steve: Oh, I just, I mean, so I think I didn't say this well. Everybody listening to this podcast has to watch yesterday's Triangulation.

Leo: Well, I couldn't agree more.

Steve: It was fabulous. I mean, it really was. I love that he said, I mean, it was so pithy. There were just so many things he had. I was thinking, wow, you know, his students are lucky people. He was talking about like how he's a physicist, and he deals with the excited states of matter. And he said the two most exciting states to be in are "confused" and "wrong." And he really understands the essence of the scientific method, that it's not about being proven right, but the only time you're learning is when you realize, oh, crap, what I thought is completely wrong. And that's an exciting thing. That's not to be ashamed of.

Leo: Didn't you love that? Yeah, yeah.

Steve: Yes, yeah. Oh, it just, I mean, and in fact I was hoping that Audible was going to be a sponsor this week because I took the trouble to verify that his most recent book, titled "A Universe From Nothing: Why There Is Something Rather Than Nothing," is on Audible.

Leo: Yeah, yeah.

Steve: So I think Jenny's going to get it and listen to it and/or "read" it, as people say. And it just sounds - oh, actually I think she downloaded it on her Kindle because she said she thought she would be better reading it than listening to it in order to not have it just go by, so that she can focus on it. But, and I may do the same thing if I can find some time because I just loved yesterday's interview. So I want to make sure that our Security Now! listeners who didn't know about it will take the time to go listen to it. It's absolutely worthwhile.

Leo: Episode 167 of Triangulation. You can find it at TWiT.tv/tri167. And of course it's on everywhere else, you know, iTunes and all of that stuff. Yeah, he's great. And

I was actually quite pleased because at the end of it he said, "Hey, this wasn't bad at all." I think he was expecting the worst.

Steve: Well, it was interesting, too. I was watching him. Whatever he was doing, he was studying something to get himself ready for it.

Leo: Yeah. He was reading the foreword - so the premise for getting him on was he'd written the foreword to a new science fiction short story collection, which he'd written more than a year ago, so he was rereading his foreword so he knew what he said. Because he thought, and I think this was the problem, he thought this was going to be all about the book. And it wasn't. That was the excuse for getting Lawrence Krauss on the air. I just, you know, because I love his stuff. He was on a cruise that I was on, it was a Scientific American cruise for the Australian total eclipse a couple of years ago.

Steve: I remember, when it was overcast.

Leo: Yeah, he was so peeved.

Steve: You came back talking about how, well, I'm sure there was an eclipse up there somewhere.

Leo: Yeah. He was mad because it felt like the captain just didn't want to get to the eclipse, he was going to stay in the clouds. But I had seen him speak, and I was very, very impressed. I knew of his books. So we just - it was just an excuse to get him on. And I think he was happy to talk about that.

Steve: Well, I want to make sure our listeners know. I loved it. And I can't imagine anyone who enjoys this podcast would not equally enjoy this guy because he was just - he was terrific. I mean, it was just - it was as fast as you could go, he was coming out with really fun observations.

Leo: That's the best part about the show, which you've been on twice. Triangulation is an interview show, and it's fun to get smart people. And to have an hour to talk to them is such a pleasure for me. I just love the opportunity. And we use it really as an excuse to get people I want to talk to on.

Steve: Yeah. It was just great.

Leo: Like you.

Steve: Well, thank you. You get me every week, like it or not. So Comcast vs. Tor. What I think this was is another example of just sort of the lumbering size of Comcast. And you can imagine a culture where the support people are not super technical. Maybe there's a few geeks who sort of tend to run the herd and like to sound like know-it-alls. Because it sounds like some of the customer support people were sort of, like, going complete off script, not really understanding what they're talking about.

But the reporting from DeepDotWeb was that - and this is just a couple days ago - that Comcast had been repeatedly asking, well, first of all, Comcast knew that their customers were using Tor and then were contacting them and repeatedly asking which sites they were visiting with Tor. So DeepDotWeb wrote: "Reports have surfaced that Comcast agents have contacted customers using Tor and instructed them to stop using the browser or risk termination of service. A Comcast agent named Jeremy allegedly called Tor a, quote, 'illegal service,' unquote. The Comcast agent told its customer that such activity is against usage policies. The Comcast agent then repeatedly asked the customer to tell him what sites he was accessing through the Tor browser. The customer [thankfully] refused to answer.

"The next day the customer called Comcast and spoke to a different agent named Kelly, who reiterated that Comcast does not want its customers using Tor. The Comcast agent then allegedly told the customer" - and I had a quote here, and it looks like it didn't make it into my notes. But it was essentially it's an illegal service, and it's against our policies, and people only who are only doing illegal things are using Tor, and therefore you should not be using it.

So after about a day went by, one of the VPs, Jason Livingood, posted an official Comcast response titled "Setting the Record Straight on Tor," flatly denying all of this, denying that these conversations occurred, citing them as being from a chatroom where apparently they weren't. And again, I think it sounds to me like this group of Comcast support people just were sort of off script, that this - maybe somebody said it, and the legend grew. Who knows. But anyway, so for what it's worth...

Leo: It wouldn't be the first time Comcast support people have gotten a little heat for their off scriptness.

Steve: Right. And on one of your podcasts recently, one of your bright people said something like, "Never ascribe to malice what is probably explainable as a mistake," sort of thing.

Leo: Yup, yup.

Steve: It's like, I don't really - this just didn't feel like they were perpetrating a policy. It's just the simpler answer is normally the right one, and it's just the Comcast support people are absolutely clueless.

Leo: I can't think of any reason Comcast wouldn't want you to use Tor unless they were afraid there was illegal activity going on in their network, and you're using Tor to hide that. But that's the same thing with BitTorrent. You can't assume that because somebody's using a legitimate protocol that they're doing it illegitimately.

Steve: Right. And ask the EFF how they feel about any of these things.

Leo: Right.

Steve: I mean, they've been sponsors of that, and for a while stopped taking bitcoin, then decided, oh, we were wrong, we're going to take it again. So, okay. So Tim Cook was on Charlie Rose. And because of the deep dive we did into Apple iOS 7 security, we've absolutely verified that, while Apple may not have the keys for decrypting iMessages, because they are managing the keys in a way that is completely transparent to their users, they have the absolute capability of reading iMessages, if they want to.

And so what Tim Cook was quoted as - what Tim Cook said, and I watched the interview, was "We are not reading your email. We are not reading your iMessages. If the government laid a subpoena on us to get your iMessages, we can't provide it. It's encrypted, and we don't have the key." Okay. So I think that is factually accurate, but it's not effectively accurate. It's not operationally accurate. They have the ability, we know they have the ability to have the keys. I mean, they actually have the keys. They may not be using them, but he says "We don't have the key."

Well, yes, you do have the key. You're providing them to each of us so that we can cross-encrypt our iMessages to each other. So by definition you have to have the key. Still, he's in the top floor of the building, and you've got to go about halfway down probably until you actually get to people who understand the technology of this. And so I...

Leo: That's probably the case, yeah.

Steve: Yeah, I'm sure it's not deliberate. I'm sure this is what he's been told. And the reason I got all this feedback from our listeners is, wait a minute, who's right here? And it's like, well, I'm sure he believes he's right. And you can see it as a way of being right. But I think technically he's not. He also did say, "Our customers are not our product." He carried on about that for a while. He says, "These gadgets here" - and he had laid them out on the table in front of Charlie. He says, you know, "These gadgets are what we're selling." We're not selling our customers' data, our customers' lives, our customers' experiences. And, I mean, he went to some effort to sort of demonstrate, I think, or differentiate himself from - clearly the elephant in the room there was Google that he was referring to.

Leo: And the NSA, and Facebook, and...

Steve: Yeah. And he said, "We go out of our way not to have our customers' data. We don't want it." And the fact is, our own analysis of the design of their system, one of the conclusions I came to after the three episodes we did on iOS 7 was exactly that. It was evident to me from a technology standpoint they were working to have as little responsibility as they could while still delivering the experience, the "it just works" experience, which we know is difficult to offer for security because that is oftentimes at odds with convenience.

Leo: I would love it if, I mean, I understand why you give Charlie Rose an interview, because it's very prestigious, he's on Bloomberg, he's on PBS. But Charlie is not known for his technical acumen, nor is he known for his hard-hitting questions. And that's another reason perhaps that they chose him for that interview. But I would love, I mean, I would love Tim Cook, maybe if he wants to bring along some expert technical people with him, to come on our network or some similar network where you'd have actual expertise, so that we can ask him some a little bit more challenging questions about this stuff.

Steve: Yeah. And I wonder if he can answer those, Leo.

Leo: No, well, that's why I said maybe he'll bring in somebody who does. Of course he can't. He's the CEO. I don't interview CEOs. You notice I very rarely do CEO interviews. They don't know nothin'. And they're not going to say that they do.

Steve: It's like the U.S. cyber czar guy, where he boasts that he doesn't know what end to plug in. And it's like, oh.

Leo: And I don't need to.

Steve: Okay, yeah.

Leo: I just sign the checks.

Steve: Yeah. And if I had that job, just shoot me right now. I'm sure I'd be screaming and then tied up in a straightjacket in about a week because of just - it's all about politics and bureaucracy. It's not about technology.

Leo: Right, right. And he's never going to come on a challenging venue and try to answer these questions.

Steve: No. Why?

Leo: He should, but he doesn't need to.

Steve: How does that benefit Apple?

Leo: Yeah. Well, that's the exact right question. How does that benefit Apple?

Steve: So LinkedIn made a mistake. And this is not a big breach. It's gotten some bad press, I think. I mean, it's not - it's a goof. Some clever security guys, the two founders of Rhino Security Labs in Seattle, Benjamin Caudill and Brian Seely, they leveraged an unintended side effect of a LinkedIn feature. LinkedIn has a feature - because it's all about getting you linked in. When you are setting up an account, LinkedIn says, hey, we'd like to help you connect to other people on LinkedIn who you know in your other social networking environments, like your Gmail address book or your Yahoo! contact list. So can we have access to that? And you say, oh, that sounds handy, so, yeah. And so LinkedIn goes and sucks out all of your contacts in this other facility and then does a cross-reference of the email addresses of the people you claim to know - thus the problem - against LinkedIn, the addresses LinkedIn knows everybody in LinkedIn has, and then populates your connections list in LinkedIn.

Leo: That's not a bug. That's a feature.

Steve: Yeah. Except that it makes it rather automatable. And that's what these guys did. I'm trying to remember the guy, I'm not remembering his name, someone famous who's gotten into technology...

Leo: Ashton Kutcher?

Steve: No. Shoot.

Leo: Charlie Rose?

Steve: Anyway, it doesn't matter.

Leo: All right.

Steve: It'll probably come. So what they did was they just put into their address book a ton of email guesses, just guessing famous people's email addresses, all kinds of permutations, their names and various ways and so forth. Oh, Cuban, Marc Cuban.

Leo: Yeah.

Steve: I knew it would come. So, and it turns out LinkedIn filtered through all of those on their behalf and said, oh, yeah, here's Cuban's email address. And they didn't...

Leo: But didn't they have to have his address to begin with? I don't understand.

Steve: No, no.

Leo: Just his name. You just put Marc Cuban in there.

Steve: No. What they did was they created fake address book entries in Gmail or Yahoo!.

Leo: So how much information had to be in that entry for LinkedIn to give them the email address?

Steve: I think they had to have the email address correct. But the point was they were able to build a huge fake contact list that was all of these guesses, and LinkedIn did the cross-referencing.

Leo: Oh, it confirmed one.

Steve: Yes.

Leo: Got it.

Steve: Precisely. And then they actually leveraged this into an interesting social engineering deal. Cuban has got some new social networking app...

Leo: Yeah, Cyber Dust, yeah.

Steve: That's it. And so they waited for a tweet to appear, they made a comment of that, then they used the fact that they had his email address in order to get - basically in order to get some work from Marc, saying, hey, would you like us to check your system?

Leo: You've got a little problem, Marc.

Steve: Yeah, because we just hacked LinkedIn. So anyway, that's what that's about. It's not - it was being billed as like a major flaw in LinkedIn. It's like, well, okay. I'm not sure how you offer this facility without opening it to abuse except, well, I just don't know because you could certainly have valid email addresses which LinkedIn doesn't know. So a bunch of non-collisions are not going to disqualify you or demonstrate that you're trying to hack that facility. And one of them is going to match if you guess a lot.

Leo: And apparently having that email address is the first step towards a larger social engineering hack, getting more information. The Apple example, the iCloud hack, they had to have a correct email address to begin.

Steve: Right, right. They had to have some foothold to get into the system.

Leo: Right, yeah.

Steve: Well, there is, however, a bad problem that was just recently discovered. And this affects all pre-KitKat versions of Android, KitKat 4.4, which is just now, as of like a couple days ago, at 24.5% share. So three quarters of Android devices are still pre-KitKat, that is, this affects them. What was found was an incredibly easy-to-exploit bypass of the same-origin policy that protects browsers.

So to remind everybody, same-origin policy is a crucial, essentially firewall feature of browsers which restrains what code received from a website can do based on the domain name, the idea being that it is free to, like, ask for other assets and do other things, as long as they are within the same origin. That is, if you receive code from GRC.com, that code can interact further with GRC.com. But it can't on your behalf go do something with Amazon because that would be bad. I mean, it could on your behalf make a query to Amazon. Your browser would give Amazon your Amazon cookie, your session cookie, because that's what browsers do, and the JavaScript could capture that, and you'd have a serious problem.

Well, that's what exists on the default Android browser, the so-called AOSP platform browser. AOSP stands for the Android Open Source Platform browser, which is installed by default on all of these many phones where the user hasn't switched to any of the alternative browsers. And it is as simple as prepending a unicode null in front of the URL of the JavaScript.

And so what this means, for example, is that, if a user using that browser on a pre-4.4 - it was fixed in KitKat quietly. But a user using one of those vulnerable browsers visits a website which doesn't have their best interests at heart. It's able to present them with a page containing JavaScript that gains access to all the pages they have open in the browser. It's able essentially to crawl the so-called DOM, the Document Object Model hierarchy of all of the pages and, for example, collect the session cookies which they may have current, and even issue requests on behalf of them to other sites that are popular to which they may be logged in and acquire their session cookies and thus hijack their current session and current credentials.

So I've got a link here in the show notes to - Rafay Baloch is the researcher who found this. And the proof of concept is as simple as a couple lines of HTML. I mean, it's just trivial. You do a /u0000, which is a unicode zero, and then the URL you want. And you're able to demonstrate that this is happening. So the takeaway is for our listeners to, if you are on a pre-4.4 Android, stay away from the default Android browser. Chrome and Firefox and others should be fine. And you can use this URL to verify that you either are or are not vulnerable. He discovered it on what is apparently his own phone, which was - I had it in the notes here, and I'm not seeing it in front of me. Oh, a Qmobile...

Leo: A weird one, a Qmobile Noir A20, which I've never even heard of.

Steve: Yes. But he also verified it on the Galaxy S3, the HTC Wildfire, Sony Xperia, Motorola, and others. A very simple proof of concept, it is now - it's already made it into all versions of Metasploit. So the bad guys have it and may be getting up to some mischief with it. So again, if this affects you, be advised that you want to stay away from that browser on the pre-KitKat versions of Android.

And some good news, anticipated. You were already anticipating it on MacBreak Weekly, the podcast before this, and that is that LastPass just announced that they will be shortly, but apparently not at launch, but soon after, adding support for iOS 8 and doing everything we want. They said in their posting: "Following Apple's announcement of iOS

8 in June" - oh, right, the announcement in June - "we've been hard at work to bring the platform's new security and authentication features to the LastPass mobile experience. Now, with the impending release of the platform, we're thrilled to announce the LastPass app will be available for iOS 8 with Touch ID integration and a Safari extension for automated web logins. This marks a tremendous shift," they wrote, "in our ability to bring a seamless login experience to LastPass users on iOS."

And all of us who are LastPass users on iOS are saying, yay. No more need - I was talking about it a couple weeks ago, about having to switch over to the LastPass tab or the LastPass app, get the password, copy it, then, you know, and I mentioned how I copy the password first, then my username second, so that it overwrites the password in the clipboard and so forth. I mean, things like that, that all goes away.

So the third-party app extension for Safari allows them to integrate into Safari so it'll just be able to populate the fields the way it does on desktop browsers. And then with Touch ID integration, they'll be able to ask for a fresh thumbprint, essentially, or fingerprint from iOS. iOS will then prompt the user to, on the fly in real-time, provide an authentication thumbprint for that instance of login. And then the return comes back to LastPass. Applications get nothing but a yay or nay. Like, yes, the person successfully authenticated just now. And then LastPass will proceed to do its work for us. So, and they said available soon, no specific date. But that's super good news.

Miscellaneousness: We already talked about iPhone stuff that I had in my notes because I wanted to briefly talk with you about that. I saw last night at Mark Thompson's recommendation - Mark Thompson, a buddy of mine at AnalogX.com and friend of ours. He said, "Steve, you've got to go look at the special about Fox's forthcoming series 'Gotham.'" And so I made some time last night. It's 22 minutes long and is available now on the Fox site. And my Fox affiliate, I guess, is airing it on Saturday, so it may also - you may be able to get it on regular Fox TV.

I wanted to just give a heads-up to our listeners, it looks really good. I was initially not that excited about it. It's like, oh, okay, maybe. But wow. It's a fabulous-looking cast. It tells the story of the evolution of the DC Comics bad guys from early - yeah. "The Legend Reborn" it's called. And it just really looks great. So I just wanted to give our listeners a quick heads-up. Go check it out if it sounds like you might be interested. Twenty-two minutes, and if you're not hooked or convinced by then, then it's probably not for you. But it just - it really looks good.

And of course many people, I tweeted my excitement about this last night, and I got a bunch of people saying, yeah, Steve, now that Fox knows you love it, they'll cancel it, like they did "Firefly" and "Almost Human."

Leo: Or even better, run it out of order.

Steve: Exactly.

Leo: The mayor vows to find the Wayne murderer, and then the murder is the next episode. That'd make a lot of sense.

Steve: That's right.

Leo: You know, this is an endless, really, isn't it, an endless well of inspiration is this DC Comic stuff.

Steve: Boy, yeah, it sure is.

Leo: How many times have Batman's parents been killed? Many, many, many times.

Steve: Yeah, you can keep turning the clock back and reinterpret.

Leo: Yeah, yeah.

Steve: We're going to reinterpret. "Batman Returns." "Batman Begins." "Batman Tries It Again."

Leo: You know what's fun, there's a - of course they were seeing a play, or an event. They were going to something downtown in Gotham, and then the young Batman with his mom and his dad walk out of the theater and get robbed at gunpoint, and the parents get shot. And what's changed is what they were going to see.

Steve: Oh.

Leo: And, yeah, there's a history. Let me see if I can find Bruce Wayne's parents. Because in the original DC Comic, Batman No. 1, it was something really ancient.

Steve: Right.

Leo: So I'll find it, and you can keep on, and I don't want to slow you down.

Steve: Yeah, so it was interesting, I was just going to say while you were looking that, in this special, we're seeing it's all, like, shot behind the scenes. So they show how this is all being shot in New York because they want that gritty real city feel. And they have some wipes where they show what the cameras actually see and then how they've been reimagined. And I don't know how they do any of this. But this wipe goes across, and you see, like, some gargoyles appear. But mostly the buildings are still kind of the same, but the facades have changed.

Leo: Yeah.

Steve: It's just cool technology. It's like, wow. Yeah, so, really neat.

Leo: So in "Batman Begins" they were seeing "Mefistofele," an opera.

Steve: Okay, yup.

Leo: In the original Batman I think it's the...

Steve: No wonder the young Bruce - or actually you like opera.

Leo: I do.

Steve: But I was going to say no wonder the young Bruce Wayne wanted to leave.

Leo: Mom and Dad, let's get out of here. Well, in the original DC Comic they were seeing "The Mark of Zorro" starring Douglas Fairbanks, the 1920 version.

Steve: I love the Internet.

Leo: Yeah, I mean, it's fascinating. Somewhere somebody has probably published a list of all of the different shows that they were at.

Steve: No doubt. No doubt. So speaking of cool technology, last Thursday I turned over the UI complete functioning client of SQRL to the guys who have been - I don't think there are any gals, so I'll say "guys" - over in the newsgroup who've been pounding on it. And it's holding up very well. I've now been pounding on Wine because there's some weird display anomalies that I've been chasing down because I would like to be able to have it run under Linux and on the Mac, also, using Wine as the interface. So there's some display and some printing problems which I've been working on. But so another step forward. Once that gets nailed down, and then I think there are a couple UI things people found, then I implement the online protocol, and we will have an easy-to-use, free, robust, secure, and anonymous identity system for the web. So getting, each week, getting another step closer.

And since I've been talking about SpinRite and RAID recently, I got a nice piece of email that I ran across last week for the Q&A when I was going through the email, that I saved to this week, from a Patrick in Pittsburgh, Pennsylvania who wanted to share his experience of SpinRite saving his RAID array. So even, yes, having a RAID means that a drive can die, and you have redundancy problems, it's not complete protection, as we will see.

He wrote: "Steve, I heard you discussing SpinRite and RAID arrays a few episodes back and thought I'd share a success story. I moved across the country" - okay, so he's in Pittsburgh now. Maybe he was out here before. "I moved across the country about a year ago; and, during the move, a Synology disk array fell out of the back of the moving truck." And then he says, parens...

Leo: Literally fell off a truck.

Steve: Literally. He says, "fell off while it was parked," he said, "but it was still bad. When I plugged it back in, all sorts of audible beeps, amber lights, and console messages indicated that one of the disks was past the point of repair and needed to be replaced." So here was a situation where he was in trouble. He said, "I took the damaged disk out of the array and put it into an old desktop, then let SpinRite do its thing. When SpinRite was finished, I slipped the disk back into the array, and it's been humming along for over a year now without any trouble. That's the second time SpinRite has saved me. Thank you for an amazing product and an amazing podcast."

Leo: Yay.

Steve: And Patrick, than you for an amazing success story. I appreciate it.

Leo: SpinRite saves the day. All right, Steve. First of all, what is SHA-1?

Steve: So we've talked about certificates a lot on the show because they are the meat and potatoes of Internet security. The way, just a quick refresher, the way the system works is that somebody running a server wants to be able to assert their own identity to someone connecting to the server. So they generate a certificate that has their information, their domain name and other information, the name of their company and where they're located and so forth. They generate the certificate. They create a hash of the certificate. And they get the hash signed by the certificate authority that essentially is saying, okay, all of the information that is in the certificate is valid. So the user connecting to the server receives the certificate and verifies that the hash is correct, and that the signing of the hash is correct. So both of those things have to be true. The hash needs to match the contents of the certificate and be validly signed.

Through history, as our understanding of cryptography has gotten stronger and evolved and, significantly, as the power of our computers has just been exponential with no apparent end, we've moved through different technologies of hashing. And we discussed on this podcast essentially the death of MD5. MD5, Message Digest 5, was the previously very popular hash which all of the Internet was using a decade ago.

And you'll remember, Leo, when we talked about this because the academic guys who did this, they used a wall of PlayStation 3's in order to create a fraudulent certificate that had the company name MD5 Collisions, Inc. And what they did was they watched the rate at which, I think it was RapidSSL was the certificate authority. And it turns out that RapidSSL was, like, almost all the certificates that RapidSSL was issuing were MD5s then. And this was in a period where the first chinks were discovered in the armor of MD5, back in '95. And it took 13 years to go from "we see some problems" to a successful attack.

So it's one thing to find some problems, and an entirely 'nother scale to turn those into an effective attack. And even so, they needed to anticipate the details of the certificate that would be issued by RapidSSL in the future because it took so long for them to crunch, to essentially crack the hash, to create, to solve the problem of creating a second hash, or a second certificate that hashed to the same value as the first certificate. They managed to do it. There was a tiny window of opportunity, and they then pounded RapidSSL with a bunch of certificate requests when they thought that the one that they were anticipating coming along would, and they had the correct solution. They got the certificate issued. And as I remember it was an intermediate.

So they had - they'd created their own certificate authority, an intermediate certificate, which could be used to then sign any other certificates they wanted. So this was a horrifying failure of the crypto system. But they weren't bad guys. They told everybody. And the industry collectively shrieked. All the browsers immediately blocked that fraudulent certificate, which they freely confessed to having created and never used it to do anything else bad, and the browsers all blocked it so they never could. And everyone quickly stopped using MD5. Which was fine because SHA-1 was the successor to that. And SHA-1 seemed strong and was already well supported.

So it was just - and this is one of the issues which Google has used to motivate them for what they decided to do a few weeks ago, or decided to announce a few weeks ago and will be doing in November. And that is that, because everyone was using MD5, and MD5 seemed fine, everyone kept using it. There was no big hurry, no emergency to move off of MD5. And Google's position is that we're in the same place now with SHA-1, the successor to MD5; that, in the future, computing power is going to continue to get stronger.

And what that means in the real world is the practical cost of cracking the hash to create a fraudulent certificate, which is the single thing we're worried about - we're worried that some actor could create a fraudulent certificate that would be accepted by all the web browsers, and they would be able to impersonate websites as a consequence. To do that, they have to solve the SHA-1 hash challenge, essentially. They need to modify a certificate without modifying its signature. And that's the key. That way the signature stays valid and trusted by an existing certificate authority, yet they've changed the contents.

And that is specifically what hashing is designed to prevent. The idea is it is supposed to be computationally infeasible to create a given hash signature from content that you want to hash. And it's the computational infeasibility which no one is arguing over time is incrementally becoming less infeasible, which is to say more feasible. The question is, when?

And some of the dialogue I've seen on the 'Net I had to sort of chuckle at because people have said, oh, in this post-Odin - post-Odin - post-Snowden era, no one any longer doubts the interest of governments in cracking crypto, which is completely ridiculous because in the post-Snowden era does anyone imagine that the NSA can't just simply create any certificate they want? Of course they can. One of these certificate authorities that we all trust is a front, or they have employees deeply planted in a certificate authority, or who imagines that the Hong Kong Post Office that our browsers trust won't do whatever the Chinese government wants them to? Of course they will.

So, again, remember that you always attack the weakest link in the chain of links which is security. And the weakest link is not the crypto. The weakest link is almost never the crypto. It's something else. And so people don't really bother attacking the crypto because that's hard. Hitting somebody over the head, that's easy, and getting them to tell you your password, or having NSA-sympathetic employees in a company, or just there's no way, I mean, the Department of Defense is a certificate authority. So I imagine they can pretty much make whatever certs they want to. So I would argue that's worrying about the wrong thing.

So what happened is on August 19th, less than a month ago, out of the blue - oh, first I should say back last November, so nearly a year ago, November of 2013, Microsoft - and we talked about it on the podcast - Microsoft announced the formal sunsetting of SHA-1 in 2017. So four years in advance, Microsoft said, you know, it's time. We're going to, in 2017, we're going to stop Windows, Windows servers, Windows clients, browsers, anything from us, we're saying night-night, sunsetting SHA-1. Nobody had a problem with that. I mean, it just was sort of a blip. It was like, oh, okay. 2017, not a problem. Four years, plenty of time.

And remember that non-EV certs are issued for either two or three years, EV certs only for two years. So setting that deadline out four years in the future didn't upset anybody because that told everybody well in advance to plan not to try to be using SHA-1, if you care about Microsoft at all, and it's not possible not to care about Microsoft, in 2017. Just, you know, in 2015, when you make your two-year cert, or in 2014 this year if you make a three-year cert, any cert that would cross into '17, you're now responsible for choosing the next generation in the secure hash algorithm, which is what SHA stands for, which is it's technically SHA-2, except that SHA-2, unlike SHA-1, is a family of hashes with different length outputs. SHA-256, 220 - is it 226? Yeah, I think there's a 226, 256, I think 384, and 512.

And, for example, SQRL uses both SHA-256 and SHA-512 for different things. I use SHA-512 as part of SQRL's entropy generation system, where I'm just hashing all this noise coming in from all these different chaotic sources and state of the processor and things that are unknowable to attackers, and that all gets poured into a big SHA-512 to generate enough entropy to create identities and for SQRL's various rather modest needs for entropy. So it's SHA-256 is where we're going next from SHA-1. So it was fine that Microsoft said this. And I think everyone's like, okay, yeah, you're right, by then it'll be time. Plenty of notice.

Out of the blue, three weeks ago, Ryan Sleevi, who's one of the security guys at Google, on August 19, makes a blog entry: "Intent to Deprecate: SHA-1 Certificates." And what was the source of concern, the reason the industry reacted very differently to this announcement from Google was this was not in 2017. This was in November that this would happen. That is, three months, 90 days later, Google was saying, "We've decided we're going to create," and this is their words, "a slow-motion emergency. We've been unimpressed with the way the industry has reacted in the past." They looked at MD5, and they use MD5 as an example. They think that was, you know - nobody, they're saying, will move until they have to. So we're going to make them.

Now, the problem is, as far as I could see looking at everything, the only problem is notice. The reason Microsoft's decision didn't upset anybody is that was plenty of notice, I mean, for people to plan. Big organizations need to plan. For example, Nick Sullivan, who we talked about back in the - I want to say Backblaze. Was it Backblaze? He's at CloudFlare. And he was really upset. He wrote in response to Ryan's posting: "This timeline for deprecating SHA-1 is very aggressive and puts us (CloudFlare) in a bind. Currently we provide certificates to our customers signed by GlobalSign's SHA-1 intermediate, with a three-year expiration. All customer certificates from this year are valid for part of 2017. We still use these SHA-1 certificates because a large percentage of the web visitors of our customers are using Windows XP SP2, which does not support SHA-256. It's therefore not realistic to expect everyone to be able to upgrade to SP3.

"This change" - and he wrote this three weeks ago, so this is current. "This change will force our customers to have to choose between having a mixed content warning" - and I'll talk about what Chrome is going to do in a second - "on Chrome or cutting off a large portion of their visitor base. Can we please delay this move until at least next year?"

So he's writing this in 2014. So here's what Chrome does. Chrome is - and this is starting in November. I think it's Chrome 39, I didn't write it down, but that's the number that sticks in my head. Chrome 39, which is expected to come out in November, its behavior changes if it sees that the certificate it is receiving from web servers was signed by SHA-1. And it's worth noting that today, today, at the day that this is looming over us, 92% of the web is signed by SHA-1. That is, there's been...

Leo: Wow.

Steve: Yes.

Leo: Including some of Google's servers.

Steve: Look at that chart, Leo, on the first page of the show notes. I put that there so you could display it. That shows SHA-1 and SHA-256. SHA-1 has just started to come down off of 100%. SHA-256 just started to come up from 0%. So, I mean, this is, there's no question, this is a premature move on Google's part. And as I said at the beginning of this, and this is a bit of a double standard, yes, there it is on the screen, that's the current relationship that you're showing now between SHA-1, just peeling off from 100% off the top at the beginning of this year, versus SHA-256 just beginning to come up from the bottom. So 92% today is SHA-1 certs, all of which Google isn't - it announced three weeks ago they're going to start flagging as insecure in November on Chrome. And so...

Leo: So it's in Canary now, which is in a very, very early pre, it's almost pre-beta because there's a Chrome Beta and then Chrome Canary.

Steve: Right.

Leo: But you're saying they're going to move this into the full Chrome by the end of the year.

Steve: November, yes.

Leo: Wow.

Steve: I know. It is extremely aggressive.

Leo: So that means that 92% of the web will - what will happen when we go there? We'll get a warning?

Steve: Exactly. So here's the way it works. They've designed this to penalize certificates that will be living - certificates seen today which will be living at two different points in the future. The first point is in 2016. So at and after January 1st of 2016. If a certificate today has an expiration in 2016, their user interface declares it insecure today, although it isn't. And that's what's wrong with this is they're saying this certificate is insecure today. What they're trying to do is they're trying to put artificial pressure on the industry to move today in order to prevent Chrome from complaining about certificates which, if you don't move, will eventually be insecure.

I mean, and again, this is arbitrary. There's no problem with SHA-1. It's like, from what I saw, it was 2021 where, if things continued to go exponential, and people got crazy hashing chips, and if we continued on this exponential rise, 2021 was where the cost to crack made it begin to be feasible for nongovernmental agencies. Because, again, governments, I'm sure, don't have to bother cracking anything. They just get certificates if they want them.

Okay. So if the certificate would expire during 2016, then you get a yellow warning symbol over the "http" in the URL, or maybe it's over the lock. So Chrome is saying there's a problem with this site's certificate, even though that's not true. It's absolutely not true. If you're unlucky enough to have more recently purchased your certificate - and in January of this year 92% of the certificates purchased in January were SHA-1, and certificates are either two or three years long. But certificates purchased in January would be valid for two years. January 2014 would be valid in 2016. And most people purchased three-year certificates because you get a discount. You get a better price per year if you do that. And so they're valid in 2017.

So, and of course certificates have continued to be purchased all through 2014 with SHA-1 predominantly because that's the standard currently. If the certificate has its expiration in 2017, then not only do you get a stronger warning in the URL, it's like red with crosshatches, but you get an additional "mixed content" pop-up.

Leo: Hate that.

Steve: Yes, that requires you to click through. And this happens in two months. This happens in November. So, and again, not because there's any problem, but because Google decided we want to create what they're calling a slow-motion emergency. And here's the double standard. Google is using SHA-1. They have their own intermediate certificate which allows them to mint certificates on the fly. They create three-month-long certificates, and they're issuing them all the time, moving them forward three months at a time.

So notice that they're forcing all of the rest of the industry, I mean, 92% of the industry has to remake their certificates or have Chrome, which is now the dominant browser, complaining about their website security when they're nothing wrong with their website security, while Google continues using short-term SHA-1 certificates that won't be expiring in 2016 and 2017 because they only mint them for three months because they've got that all automated, and they're continually churning them out. It's why I had to stop using one of those neat Firefox plugins, because it kept seeing new Google certificates all the time and was warning me that Google certificates were changing. I had to stop using it because it was creating so much noise. So this is really a double standard. And again, Google is using SHA-1 because there's nothing wrong with it. But they just decided we're going to force everyone to reissue certificates.

Jeremy Rowley, who is an executive at DigiCert, of course my certificate company of choice, he also responded many times in this thread to Ryan. His first response was - and later ones were much longer and full of bullet points and raising all kinds of good points because, again, this is unilateral, you could argue, abuse of power. Google has earned the position they have with Chrome, and now they've just decided, okay, we're going to wield it as we choose to. And it's not a problem for us, even though we're using SHA-1 certs, too, because we can mint them on the fly. So we won't be tripping our advance warning systems.

Jeremy wrote: "With rekeys, duplicates, early renewal programs and so forth, I think most of the certificates valid in 2016 will be replaced well before then." So he's saying there is already natural churn and a movement in this direction. He said: "Considering many CAs are working to move existing SHA-1 customers to SHA-2 before 2016, showing a UI degradation now will probably only serve to upset website owners rather than spur faster adoption. Most of these customers are on a set transition plan that will move them to SHA-2 well before the deadline, despite having a certificate currently that extends past the end of 2015. At least consider a window in the validity period range where the transition will likely happen before 2016, despite a post-2016 validity period, such as March 2016," he just suggests.

"Putting the degradation on certs with a validity period past March 2016 will likely have the same effect in spurring a transition to SHA-2, but let people keep their current transition plan intact." So essentially Jeremy is saying, is wording differently what the CloudFlare guy said, which was, wait a minute, you know, this is just too soon. Microsoft said 2017. Google, if you want to say sometime in 2016, if you want to push it forward, okay. But don't push it to 60 days from now. That's wrong. I mean, it's really a problem. So that's the story. It's I think telling that Google is - the first line of Ryan Sleevi's post is "The use of SHA-1 within TLS certificates is no longer sufficiently secure." That's how he starts this. Yet that's all that Google uses.

Leo: So if there were - somebody in the chatroom said, hey, if it were Heartbleed, you would update, even though you had just bought a certificate, you would update it. Isn't there a security issue with SHA-1? Or no? You're saying there is no issue.

Steve: No, there is no known problem, or we wouldn't all be using it. Google wouldn't be using it. There's no known problem. What they're trying to do, they're looking at the past. And, I mean, they're not wrong about...

Leo: Because of improvements of computational ability.

Steve: Yes, exactly.

Leo: But we aren't there yet.

Steve: Correct. And we're not going to actually be feasibly there until 2020 something, like 2021.

Leo: This does seem like a kind of premature rush to fix this problem.

Steve: Yeah, I mean, to me it almost feels like they just want to be first. They want to be the ones to yell "Fire" in the theater, just so that they can. And they want to do that.

Leo: That's kind of a shame.

Steve: It is. I mean, and so what is going to happen is there are a lot of us who - oh, and I should say, Leo, when I minted my first - when I went to DigiCert, the default is SHA-256, and I had compatibility problems. That is to say...

Leo: Oh, that's interesting.

Steve: Yes. GRC is using an SHA-1 cert right now today. And in fact I want to see whether I can reissue an SHA-1 cert that expires on midnight of New Year's 2015 to thumb my nose at Google, essentially continuing to use SHA-1 while not triggering this ridiculous notification, this bogus "Your site is insecure; you're not protected; there's a security problem." I mean, that's what users are going to believe.

This is going to be - watch what happens in November. This is going to be a disaster because a lot of us are on the inside. We're following this. The CA industry is - they're taking steps. But there are an awful lot of sites now that are going to have users saying, hey, what's wrong with your server? It's not secure anymore. Then it's like, okay, yes, it is. I mean, ignore that. So that's bad. We don't want people being told and taught to ignore the security, the lack of security notification. It's taken us until now to get people to start paying attention to that and notice if the thing is green or if the padlock is closed and so forth. Now we're going to be saying, oh...

Leo: Ignore that.

Steve: Oh, that's just Google. That's just them being picky, and everything is really fine. So you can ignore what that says. But that's not the message we want to start teaching. But it's going to start happening in November. Yeah, now, so what this means is it means that all servers, 92% of the certificates need to be either reissued under SHA-1 so that they're not - so that Chrome will see, oh, my god, that they're not valid in 2016, that they expire on New Year's Eve of 2015, at the end of 2015, or go to SHA-256 certs.

And so certificate authorities will need to create a system that makes this easier. And, I mean, there were some good - I'm not arguing with many of the points that Google made, that Ryan made in his posting. There was a lot that was valid. He raised the point that, as an industry, we're too rigid. The MD5 catastrophe, where no one should have still been using MD5, but we all still were, and suddenly there was a great panic to get away from MD5. That's, I mean, he's not wrong about that. And he's not wrong that just dealing with certificates is not really frightening, but it's something you only do every three years typically. And then it's like, oh, you've got to, you know, and we see them, like people being caught off guard. They expire. And then suddenly, oh, my god, our certificate's expired. You run around trying to get them updated.

I mean, it's just - they're arguing that, in general, this shouldn't be a problem. It should not be a problem for us to do this. It should not be a problem for the entire Internet to leave SHA-1 in 90 days. I don't know what they're smoking up there, but this is going to be interesting in November because the entire Internet will not be off of SHA-1 in

November. There's just - there's no chance. And people are going to get yellow triangles and say, "Okay, what does that mean?" "Oh, ignore that." Oh, geez, okay.

Leo: We should point out that Google Chrome Canary is - just because something's in Canary, it's for testing purposes, doesn't mean it's going to migrate to the Chrome that everybody uses. Not even to Beta.

Steve: Oh, no, it's...

Leo: I know it's their intent.

Steve: Okay, yes, that's absolutely right. I've got people complaining about mixed content warnings on my site now.

Leo: But you shouldn't, because anybody who's using Canary is a very advanced user. And I don't even use Canary. I mean, I do occasionally, but it's not really recommended. And they could very easily at this point back down.

Steve: Yes.

Leo: It's not in the - so I bet they will. I mean, it seems like they have to. Seems odd.

Steve: I know.

Leo: Has anybody responded to these criticisms?

Steve: Oh, Leo, oh, my goodness, yes. I mean, if you click on the - it's the link, let's see, where is it? Oh, you see the pie chart. That's from Netcraft. That's current. It's "Out of the blue," so it's a page above the blue pie chart, the groups.google.com message. That's Ryan's posting. And this thing, I mean, the industry is really not happy. So, yeah, I mean, it's an education looking through that, yeah. So there's Ryan's posting. And if you go down you'll find familiar names. I recognized Jeremy's name immediately, and Nick Sullivan's name, because these guys are active in the industry. But this is Google's stated intent with Chrome, basically using the power of Chrome to force the industry to change. When there's no problem.

Leo: Right. Oh, wow. Huh. Well, I'm glad you raised the awareness on this. I can only think that they'll back down on this.

Steve: I can't know. I don't know. But it's like, hey, we like to have exciting podcasts, and I have a feeling in November we may be having a few.

Leo: Yeah. Well, if they go ahead, it's going to be a nightmare. I mean...

Steve: It's going to be an absolute meltdown.

Leo: They could also change how Google Chrome reacts to these particular certs. That's the default reaction right now, but they could have a special reaction.

Steve: I don't - so, okay, so here's the problem. This is clearly a pressure move. One of the reasons the industry is upset is that this is Google "Do No Evil," basically commanding the industry to do something it actually doesn't need to do. There is no need for this to happen. Microsoft's 2017 sunset was just fine with everybody, and way in advance of computational feasibility for SHA-1 being a problem. But they're just saying, no, we want you to do it now, just because we can, because we're Google, because we have the power to force the industry to change. Nobody wants to be strong-armed, I mean, just on principle.

Leo: Well, I imagine what the upshot of this will be, people like me will say "Stop using Chrome." So that's something they may want to consider.

Steve: And so my point was that, in response to yours, is that, when you connect to a site, you're either secure or not. I mean, it's like, how do you say - how does the user understand that there's a battle going on between Google and the web server, and you're the innocent bystander being used to put pressure on Google's behalf against the site, that Google is trying to leverage their users of Chrome to put political pressure to do something that the site actually doesn't need to do. And the political pressure they're putting on the site is claiming that the site is insecure. I mean, there's nothing worse.

Leo: Now, there wouldn't be a warning if the cert expires before the end of the year in 2015.

Steve: Correct.

Leo: So can they - but they'd have to reissue certs that go beyond.

Steve: Yes. Everybody on the Internet. I mean, this is the problem. As Jeremy said, companies have systems, have cycles, have planning. I mean, and the CloudFlare guy, how many sites do they host? Tens of thousands of sites that all have to change. And here's the other issue, Leo, the reason I wanted you to bring up that other page? That's the page of SHA-256 compatibility. Look through it. I mean, the early Windows phones are not compatible. XP SP2 is not compatible, doesn't know about SHA-256. There are all kinds of other things that are going to break just because they're in environments that can't move. And the problem is they don't have to move except Google's decided, eh, we're going to make them. We're going to make everybody do this.

Leo: That's kind of Google's point is we have to do this; right? Or everybody would just sit on what they've got right now. So...

Steve: Except we already had the Microsoft deadline of 2017, a hard deadline nobody is upset about. And the CloudFlare guy, Nick Sullivan, and Jeremy at DigiCert, all they're saying is, 90 days? Come on. Give us a, you know, come on, I mean, that's too soon.

Leo: Some people in the chatroom are saying, well, maybe Google knows of an attack, and they just don't want to tell anybody.

Steve: That's absolutely possible. You're right. Although...

Leo: And so they think there is some urgency that maybe we don't think exists.

Steve: Maybe then Google shouldn't be using them, either.

Leo: But those are short-term certs, which they'll stop doing soon, I presume.

Steve: Why? Their browser doesn't care. They can march right along through the rest of the year and through all of 2015, right up to their own deadline. And then that gives them plenty of time. They've got a year from now before they have to worry about it.

Leo: Is it conceivable that Google knows the government has cracked these, and can't say, but wants people to create government-proof certs?

Steve: Well, in that case, Google doesn't care about the government spying on their users despite all their talk...

Leo: Because they're not using them.

Steve: Because they're using SHA-1 right now. And presumably they will all through 2015. I'm going to. I'm going to see if I can get DigiCert to cut me off on midnight. And then, okay, fine, Google.

Leo: It's very interesting. I have to think, I mean, I'm reading this Google Groups posting.

Steve: For the first time, yeah.

Leo: Yeah. And I have to think there'll be something more substantive in response from Google, I hope, to this show.

Steve: Actually, Google doubled down on it with a formal security blog posting. I read it, but I don't think I have the link here in the show notes. And it basically said, yes, this is what we're doing. It had a different headline, and it was not - it wasn't just cloning what Ryan posted. It was, like, Google's formal declaration. It's like, okay.

Leo: Well, well, well.

Steve: Like I said, we are not going to have any dull podcasts.

Leo: It'll be very interesting, yeah. Well, Google, let's hear from you. Because this is weird, frankly. And I don't really understand the scope of it. They're claiming all this is going to affect fewer than 1% of sites. But it's more like 92% of sites.

Steve: Yeah, uh-huh. Everyone else thinks that.

Leo: Why is their number so low? Where are they getting that number from?

Steve: I don't know what they're thinking of.

Leo: All right. Well, get ready because I imagine people will go to Firefox from this, frankly. Go ahead.

Steve: And, well, yeah. Okay. The other thing I wanted to mention was remember that the way the system is set up is, in order for this to be effective, that is, if SHA-1 is a problem, then we can't have any SHA-1 accepted. By that I mean, if a web server is switching to SHA-256, and it's got great security on into the year 4000, so, great. It's using SHA-256. But if SHA-1 is a problem, or believed to be a problem, then no browsers can accept SHA-1. That is, that's what we actually have to prevent is the acceptance of SHA-1 by any browser because a bad guy, I mean, because the browser doesn't know that a given site has an SHA-256 certificate, if they receive an SHA-1 fraudulent certificate from a fraudulent site. If the browser is still willing to accept SHA-1, then we still have a problem.

So what would have happened is that, by 2017, there would, with four years of notice, nobody would generate any sympathy for still trying to use SHA-1. And all the browsers would - Mozilla would issue a new Firefox, removing support for SHA-1. Lord knows Chrome would, although they'd have to stop using it themselves first. And we know that Microsoft will. And at that point we are then safe from any future use of SHA-1 in the year 2021, when it might theoretically be computationally feasible. Remember, governments don't need to crack certificates. They just issue them. There's no question that the NSA can have a certificate for any site that it wants. If we don't believe that today, we haven't been paying attention for the last year.

Leo: Right. Wow. All right, well, be very interesting over the next few months.

Steve: Fun times. Fun times, yeah.

Leo: You'll find Steve Gibson at GRC.com. That's his website. Questions, thoughts about this or any of our topics can be fed to him at GRC.com/feedback. We'll probably be doing a Q&A episode next week, so this would be a good time to do that: GRC.com/feedback. You'll also find 16Kb versions of this show there, along with full human written transcriptions and lots of other great stuff, including SpinRite.

Steve: I got email from Elaine last week. Toward the very end of the podcast, when you and I were - we were, like, talking over each other. And I said something that made absolutely no sense. And I can't remember now what the phrase was. But so she sent it to me, she said, "I'm going to hold the transcript until you tell me what you meant." And I read it, and I said - I wrote back, "I have no idea." Just moved out of whole - it's like I said something like "The Romans are in the trees" or something. It's like, what? It's like, oh, I have no idea what that was.

Leo: That's why we have humans.

Steve: Yes. Thank you, Elaine.

Leo: So these transcripts make sense. You'll also get lots of great stuff like SpinRite, the world's best hard drive maintenance and recovery utility at GRC.com. So, yes, it's Steve's bread and butter, so check it out.

Steve: It fuels all of this.

Leo: Yeah. Work proceeds on SQRL. You can read about it there and a whole lot more.

Steve: Yup.

Leo: We have full quality audio and video at our site, TWiT.tv/sn. And of course the best thing to do is subscribe to every single episode on your favorite podcatcher. There are lots of them, including dedicated TWiT apps on all the platforms. And that way you can make sure you don't miss an episode. We do Security Now!, by the way, Tuesdays, right after MacBreak Weekly, 1:00 p.m. Pacific, 4:00 p.m. Eastern time. That's 2000 UTC. And in two weeks...

Steve: Yes.

Leo: Is that right? Or is it three? Three weeks.

Steve: On the 30th. Three weeks.

Leo: No, that's two weeks. On the 30th or the...

Steve: Yeah, two weeks, yeah.

Leo: Anyway, that's right, because Microsoft is, believe it or not, announcing the next version of Windows on September 30th.

Steve: Oh, thank god.

Leo: So Mary Jo Foley and Paul Thurrott will be at that event in San Francisco on the 30th and will immediately after the event come up here and do a special edition of Windows Weekly in this time slot. And then you will be going into the Windows Weekly time slot the following day.

Steve: Yes, exactly, so I'll be on Wednesday at 11:00.

Leo: October 1st, Wednesday.

Steve: Ah, right.

Leo: Right, just for - but that's two weeks hence, and that's just for that week, and then we'll back to normal.

Steve: Yup. And we'll remind everybody next week.

Leo: Yeah, yeah, of course.

Steve: We mostly have to - we have to remind ourselves, as well.

Leo: And, frankly, that's what that was for. Don't worry, folks.

Steve: And everybody, don't forget about yesterday's Triangulation. You will not regret that podcast.

Leo: Oh, thank you, yeah.

Steve: Thanks, my friend.

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