SECURITY NOW!

Transcript of Episode #656

TLS v1.3

Description: We discuss the mess with U.S. voting machines, technology's inherent security versus convenience tradeoff, the evolving 2018 global threat landscape, and welcome news on the bug bounty front from Netflix and Dropbox. We have the interesting results of Stack Overflow's eighth annual survey of 101,592 developers, worrisome news on the U.S. government data overreach front, some useful and important new web browser features, messenger app troubles, a critical Drupal update coming tomorrow, some welcome news for DNS security and privacy, a bit of miscellany, and a look at the just-ratified TLS v1.3.

High quality (64 kbps) mp3 audio file URL: http://media.GRC.com/sn/SN-656.mp3
Quarter size (16 kbps) mp3 audio file URL: http://media.GRC.com/sn/sn-656-lq.mp3

SHOW TEASE: It's time for Security Now!. Steve Gibson's here. It's time to talk all the latest security news and, when we get to the end, a review of, finally, the arrival of TLS 1.3. Oh, and if you're a Drupal user, you're going to want to stay tuned. There's some very important news about Drupal security coming up. Lots more, too. Security Now! is next.

Leo Laporte: This is Security Now! with Steve Gibson, Episode 656, recorded Tuesday, March 27th, 2018: TLS Happens.

It's time for Security Now!, the show where we cover your privacy and security online with the guy in charge at GRC.com, the Gibson Research Corporation's head honcho, Mr. Steve Gibson. Hi, Steve.

Steve Gibson: One might say the only honcho. There's not a lot of honcho competition over here.

Leo: I don't even know what a "honcho" is.

Steve: I don't, either. Sounds like someone that rides on a horse, maybe. I don't know.

Leo: Yeah, Honcho meet Pancho. Hello. Happy Birthday, Steve. I understand you had a celebration yesterday.

Steve: It was, yeah, March 26th is the annual anniversary of my existence on the planet. And so far I'm still here.

Leo: Yay. Yay. We're all relieved.

Steve: And speaking of "still here," so is TLS. Today's title is TLS v1.3 Happens. And when I mentioned that to you, you got all excited.

Leo: Woohoo!

Steve: And I said, yeah, I guess that's exciting for a certain demographic.

Leo: You know, the only reason I said that is it seems like it's been about to happen for ages.

Steve: Has been about to happen. And in fact at the end of the show we'll talk about what it means. But one of the things that I did was I plotted the length of time between TLS, or SSL as it was formerly named, sort of like Prince, the number of years between. And the fact that it's been increasingly slow suggests that we're beginning to get it right. And in fact one of the adoption protocols that we'll talk about for TLS reminded me of how right the way this was done is, and how wrong the way the Wi-Fi Alliance does theirs. I mean, you couldn't find a more polar opposite approach where - well, anyway, we'll talk about that.

We had a lot of other news. There's the mess with U.S. voting machines and the continuous concerns and politics surrounding this upcoming midterm election and then the next presidential election and what that means. We've got an item that put me in mind of technology's inherent security versus convenience tradeoff, how we want both, but it's a tradeoff no matter what we do.

Symantec's annual global Threat Landscape Report is out and looking back at 2017 with some interesting numbers that we need to talk about. There's some welcome news on the bug bounty front with Netflix and Dropbox in different ways. The interesting results from Stack Overflow, which is a major developer site, you know, many times when I'm just googling something that I want to quickly understand or get some details of, Stack Overflow will pop up in Google search results. So it's a big deal.

They have their Eighth Annual Survey, the biggest one they've ever had. 101,592 developers were asked stuff, which is interesting, and we'll talk about that. We've got, and I heard you talking about this at least on MacBreak Weekly because I just had the show on as I was wrapping things up and getting things ready. Worrisome news on the U.S. government data overreach front. And I wanted to mention that I'm really liking the new Leo. I don't know what happened to the old...

Leo: What new Leo?

Steve: Well, the new "I'm concerned about privacy" Leo.

Leo: Oh, yeah, I have changed a little.

Steve: Rather than, "Ah, I don't care about this. It doesn't matter, blah blah blah." It's like, now there's a new Leo in town, and I think it's wonderful.

Leo: Yeah, well...

Steve: A whole new Leo. It's like, where did this come from?

Leo: I don't think I'm the only one. I think the Facebook debacle has really kind of widened people's eyes.

Steve: Yeah. And so we'll also talk about that because we've got some messenger app troubles, Facebook Messenger among them. Some important new browser features. And you guys are Drupal-based; right?

Leo: Yeah, we have a Drupal backend.

Steve: So there is a critical, CRITICAL, all caps, preannounced Drupal update coming tomorrow which was announced last Wednesday deliberately with a week's advance. Of course we didn't know about it last Tuesday for the last podcast. It happened the day after. So I want to make sure that you guys and our listeners all know, I mean, they're saying - they've declared, I think it's between 18:00 and 19:30 UTC. I don't know what that is for normal people. But, I mean, like tomorrow is the announcement with it is urgent that everybody update. Apparently they expect there to be exploits happening within an hour of their announcement.

Leo: That's at noon Pacific, I think, 9:00 a.m. Eastern, yeah.

Steve: Okay, good. So we'll talk about that. We also have some welcome news on the DNS security and privacy front, a tiny bit of miscellany, and then we'll wrap up with a look at the just-ratified, finally arrived, much-discussed, and somewhat controversial due to perfect forward secrecy, TLS v1.3, which has actually now landed.

Leo: Wow. Amazing.

Steve: Yes. And a pretty fun Picture of the Week, too, while we're at it, which sort of ties into one of the things I want to talk about in our miscellany. So a good podcast.

Leo: All coming up.

Steve: So our Picture of the Week some clever person produced shortly after Elon Musk's stunning launch of the Tesla and the dummy in the spacesuit, the Starman. I just got a big kick out of it. I'm going to mention the Star Trek franchise in our miscellany section here in an hour and a half or so. But anyway, the Picture of the Week is just - it's a kick. It's the Enterprise Voyager crew, who apparently opened their shuttle doors and pulled in the red Tesla roadster. One of them is aiming, I can't remember his name now, aiming a tricorder at it, and the Vulcan guy has got his hand on it, and they're sort of standing around thinking, what the heck is this thing doing?

Leo: It's V'ger. I love it. I love it.

Steve: Well, anyway, just a fun little bit of just, yeah, love the creativity of the Internet. So for the sake of just not leaving something out that probably doesn't really matter, or I guess it matters because it just demonstrates sort of how nothing is happening that should, what made the news is that the U.S. Department of Homeland Security has designated election systems as a subsector of the country, the U.S.'s critical infrastructure after the intelligence community broadly concluded last year that Russia had tried to interfere with the 2016 U.S. presidential election. As a consequence, it turns out there's something that I was really never very aware of. They're called ISACs, I-S-A-C, which is the abbreviation for Information Sharing and Analysis Center, which is our tax dollars here in the U.S. apparently spinning around in circles and getting nothing done.

Leo: Oh, man.

Steve: Look at this website. I have a link here in the show notes: NationalISACs.org/member-isacs. And so there's already a bunch of them. Who knew? And I'll just give you - I'll read down to the I's in alphabetical order: automotive, aviation, communications, defense industrial base, downstream natural gas - as opposed to upstream natural gas, I guess - electricity, emergency management and response, financial services, healthcare, information technology. Okay, that gives you an idea.

So who knows what this is. I mean, it just - it's bureaucracy at its finest. And the problem of course is that - so the idea is that states and localities are supposed to become members of the election system's ISAC, and I don't know what happens then. I mean, the problem is, and we've been discussing voting machines for years, which arguably are, based on last year's Black Hat and Defcon, horrifically insecure. I mean, we've talked about unauthenticated remote WiFi access, Evil Maid access where there's like an active USB port on the side, and you can just kind of sally up next to the machine and slip in a thumb drive and take it over remotely and make it reboot and dump its contents out.

And we were talking earlier this year about the problem that the Black Hat and Defcon conference organizers are having for this coming summer's events because now the people who have traditionally been reselling these machines are being illegally threatened by their machines' manufacturers against reselling, with attorneys writing them fraudulent letters claiming that it's illegal to resell them. And certainly after, what, we're approaching the end of year 12 now, and nothing could be more clear to us and our audience than that this whole process is completely bass-ackwards. I mean, completely broken.

What there should be is the kind of process that we'll be discussing, that adopted and ratified TLS v1.3, which is a standard that industry and academia together worked on, polished, refined, argued over, fought over, wrestled to the ground, tested, implemented, and just absolutely nailed. And now it is being adopted. That's the process that we should have for something in many ways sort of similar to, that shows many similar characteristics to voting automation. The idea that Diebold - pronouncing the name correctly this time - could offer for sale a black box that people adopt and use for something as important as voting, where it's, oh, this is our proprietary technology, and you can't see inside, but trust us. No. No.

I mean, and it's just - it's insane that, unfortunately, this aspect of capitalism has been allowed to take hold and be employed in the U.S. or any election system. It ought to be an open standard, developed in full view. And then, yes, I have no problem if then companies want to compete on implementing the standard which is created, which is what we do now, where we have other standards. The standard is open. You implement the standard. It is verified that you have done so correctly. And then you're welcome to sell your hardware that runs the system that has been universally agreed upon and pounded on. You're welcome to sell that. Fine. But this notion that, oh, we've got the greatest security experts; and, after all, we're us. Who cares?

Anyway, so nothing is happening. And I don't know how, like how long it's going to take for anything real to happen. But as long as we allow closed architecture voting, which is what this has become, I mean, and this is why now they're saying, okay, well, let's get a paper trail. Okay, well, okay, I guess that's better. But it still seems problematical. So this is a problem we know how to solve, but unfortunately commercial interests in this country are preventing its proper solution. So who knows when that'll change. However, creating a new bureaucracy to talk about it doesn't seem to me like we're moving in the right direction.

The Intercept had an interesting article last week where they probably wouldn't surprise anyone to note that the NSA had a system using some of their crazy project names, in this case OAKSTAR and MONEYROCKET, which tied into the XKeyScore search system. It turns out that, since 2003 at least, the NSA has been actively tracking bitcoin use. And certainly who would this surprise? Maybe what is surprising is that there was a deliberate project designed to do this.

So it is the case that, when we first started talking about these nodes on the Internet that were installed in high-throughput datacenters where the NSA was basically monitoring all traffic that passed through, we spent a lot of time talking about what that means and the consequences of it to privacy and HTTP versus HTTPS and all of that. It turns out that one of the things that was already being done in the auspices of these OAKSTAR and MONEYROCKET projects was that traffic was being analyzed for bitcoin addresses, and the NSA was already - they'd already rolled up their sleeves some time ago and were essentially way ahead of the rest of pretty much the academic community, which came later, and said, you know, this isn't really as anonymous as everyone seems to think it is. And this blockchain that is creating an audit trail can itself be audited, and we can figure out where the money is going.

Well, the NSA was doing that quite some time ago. So this is what put me in mind of noting that we really cannot have it both ways. We want convenience. And I would argue that we can have the opportunity for security. That is, with limits, security can be had on the Internet. And we'll be talking a little bit later about some of the revelations regarding Facebook and their Messenger that again demonstrate that it really is illusory beyond a certain point; that it is really, if you're relying on technology which you didn't create, if you're relying on any magic, any hocus-pocus, any black box, then you actually can hope, and maybe you have the opportunity for security, but there's no guarantee.

And we've talked about how, if you really want to have a secure communication with someone, you've got to meet them naked under a thick blanket in the middle of a football field and whisper into their ear. Otherwise, if you're using any technology, it's just not secure. And of course this has been known for a long time. Phone tapping and tracking people's cellular phones, even before we realized it was actually happening, it was the fodder for fictional stories, and at one point it was seemingly fiction. Now it turns out, no, it's something that's just going on all the time.

So anyway, I thought it was interesting that people are still seeming surprised to some degree that this is going on. And I think our takeaway is that the Internet provides, again, sort of opportunistic security. But the moment you rely on something you didn't create, the moment you rely on something you don't have absolute control over, then you're making an assumption that it's safe, which may or may not be true.

Leo: But, I mean, this is modern life. We're all interdependent in modern life. You drive down the freeway at 60 miles an hour in a two-ton hunk of metal, and you just hope that the parachute's been properly packed. I mean, there's no - you know?

Steve: Right.

Leo: And that the guy coming in the other direction didn't have one too many. I mean, it's society. Really, the truth is it's a problem of civilization as much as anything else. We're interdependent.

Steve: Well, yes. And look at the convenience that we get. I mean, look at the leverage this provides. I'm not suggesting - we all know I'm no Luddite. I'm not suggesting we roll things back and give up electrons. I love electrons. But I think that's the point is that there's this, you know, in hearing people talk, there's this notion that security can be an absolute. And clearly one of the prevailing lessons of the many years of this podcast is that, no, there really isn't an absolute. The more interest there is in penetrating security, the less secure it is.

Leo: And I think also it's important, part of the equation that we don't really get too much into is the hazards of the risks, what the risks are. For instance, there's only four days' food supply in the United States. You want a real problem...

Steve: Yeah.

Leo: You know, just interrupt the food supply and watch what happens. So that's a serious risk, but a difficult thing to accomplish. Hacking the grid, a little bit easier, apparently. In fact, apparently it's already happened.

Steve: A little too easy, yes.

Leo: With some pretty severe consequences. Somebody knowing where I get my hair cut isn't the end of the world. The hazards are lower or lighter. I mean, they're there. And as you point out, I've recognized recently that they're significant. But I don't think it's like somebody's going to haul me away in chains, either.

Steve: Well, no. And I guess I would argue that this is, I mean, not like any of us are contemplating being criminals. But it seems like a really bad idea these days because...

Leo: Yeah. You would think.

Steve: Yeah.

Leo: But don't the criminals get enabled, too? I mean, that's kind of why I am against the idea the Department of Justice has been floating of the hidden keys in the phones that - it's something you talked about. It would be one good solution to this is a unique key that is unlockable with a second lock by the manufacturer should law enforcement need it. But at the same time, I don't know. It's kind of...

Steve: Well, in fact, yes. Russia has given Telegram 15 days from last Tuesday.

Leo: Right. Right.

Steve: So it was last Tuesday that the high court in Russia said to Telegram, you turn over your keys, or we're shutting you down. And Telegram intends to fight and appeal. But they got shut down by China because...

Leo: We knew Putin passed this. This was passed a year ago.

Steve: Yes. Yes.

Leo: I'm not willing to give up. You've got to just - what it is, is knowledge. You've got to know what the tradeoffs are, what the risks are.

Steve: Yes. Yes.

Leo: And we have a benefit by using all this stuff. Is it worth...

Steve: I would argue a massive benefit.

Leo: Massive. And this is what Jeff Jarvis is always saying, too, is that is it worth the tradeoff? You just need - it's about knowledge, not about walking it into ignorantly.

Steve: Yeah. So, okay. I got a kick out of Symantec's number for the explosion in cryptojacking. Because, you know, when the denominator is very small, even a modest numerator results in a very large quotient.

Leo: Oh, Steve, math is hard. Why are you bringing up math again?

Steve: So Symantec, and we talk about this every year because it's always interesting stuff, they published for 2018 their Annual Internet Security Threat Report, looking retrospectively back on what happened in 2017. Well, now, of course we've been talking about cryptojacking. And in fact, yes, it is the new thing. It is the latest and greatest means for bad guys to try to make money. But the headline is "Cryptojacking attacks explode by 8,500%."

Leo: Wow.

Steve: And it's like, whoa, that must just be - how are we even operating? And of course I'm put in mind of the fact that there was probably one cryptojacking attempt at the end of 2016. And so, yes, by comparison...

Leo: There's 8,500...

Steve: ...that is a massive [crosstalk].

Leo: What is cryptojacking, just so we know what's increasing by 8,500%? Ransomware?

Steve: No, no, crypto mining. It's running miners on other people's stuff.

Leo: Oh, that thing.

Steve: Yes. And of course, if you count all of the browsers...

Leo: It's through the roof, yeah.

Steve: ...that visited Coinhive recently, I'm sure that's a bunch. So it's like, okay, fine. So, yes, no surprise here that cryptojacking is the big deal. What was interesting in their report is that they're hypothesizing that the next target will be IoT. And, okay, maybe. So we know that end user browsers were the first target, a la Coinhive, because you could just run a Monero miner on somebody's computer, and they'd wonder why their page was so slow until it turns out that an ad or a compromised website had injected crypto mining code in their browser page, and the computer was like, oh, my god, really? This is the worst JavaScript I've ever had to interpret.

So next we covered that higher power servers have since been hit, and that, oh, there's money in those servers. If you can get some serious metal cranking hashes, and a bunch of them all aimed at the same cryptocurrency account, it adds up. So there's money there. Now, of course the next question would be, okay, what about IoT? And you would think just based on the number of light bulbs and Nest thermostats and door locks and alarm systems and DVRs and routers, certainly lots of routers, that, wow, okay, there is a target-rich environment.

The problem is that this is a little more like browsers, where the value proposition from a browser was difficult to make, which was why there was the switch to high-powered servers because this really is all about how fast can you hash? And any consumer device is super cost-sensitive. And so the processing power is scaled down in your average light bulb to just the bare minimum to decide what color it should be and to get it onto your network and hope. So it remains to be seen whether, I mean, now, maybe routers - routers have some beef behind them. So I could see consumer routers being targeted.

But I would be surprised if this penetrates much into these devices which are just barely online. I mean, they're online enough to receive a packet. And I would argue they make - and I'm not suggesting this - that they make better attack platforms for DoS because it's easy, takes very little processing power to emit a packet. And they can do a lot of damage at the receiving end, especially if it's being leveraged off a memcached server. But there's no way to short-circuit the proof-of-work required for crypto currency. And, I mean, that says it, "proof of work." It's hard to get much work out of a light bulb. It's just, you know. So I think we'll have to see whether that's ever going to happen.

Leo: We make it up in volume.

Steve: Exactly. Now, they did say, Symantec said that they logged 1.7 million cryptocurrency-driven intrusions during the month of December 2017 alone. But again, that can't even be servers. That's got to be browsers. So they're casting a wide net for their definition. The report went on to say that lower tech targeted phishing remains the number one way the majority of attacks begin. That's still the way people get in is they send somebody a seductive email that says - and my own email system sees those constantly, just people sending you some nonsense with a zip file saying that you've won something, or DHL needs your signature release to deliver the package, I mean, just random stuff.

Leo: Or the IRS; right? I should play you this voicemail that Lisa got from the "IRS" saying the sheriffs are on their way. They're coming for you. It's all fake, yeah, but it works.

Steve: And I think it might have been on mainstream media, like on TV, that someone I saw, it was either that or it was on one of your weekend shows, someone saying this is not the way the IRS works.

Leo: No, they don't do this, no.

Steve: Exactly.

Leo: No, no. But it's this time of year it happens. It's very sad, really.

Steve: Right.

Leo: It doesn't get you and me, it gets elderly people who are kind of a little clueless about technology or about how things work, and they get scared.

Steve: Yeah. So the number is interesting: 71% of attacks are traceable back to spearphishing, 71%.

Leo: Wow, that's interesting.

Steve: Yes, nearly three out of four attacks begin that way. So it is still, I mean we know that that's the way that the APT, the Advanced Persistent Threat got into Sony, was that we tracked it back to an administrator that opened a piece of email and clicked on something that he or she shouldn't have, and that allowed somebody in. And the concern is that, as I've often commented, it's amazing to me that all these servers are being compromised just for their CPU power for cryptocurrency mining. Nobody much cares, it seems, that they just did establish a beachhead in someone's network. I mean, if you're in their server, you can, like, look around and see what else delicious might be there.

Leo: Yeah, that's much more concerning.

Steve: Yes, yes. And instead it's just like, oh, how strong is this processor? Let's squeeze it for something. Let's squeeze some coin out of it.

Leo: Coin out of it.

Steve: So the concern, though, is in a targeted attack, an enterprise's network is the target, that is, they went in for a reason, and then they tend to set up shop. And one of the things that Symantec finds when they do forensic after the fact, after the discovery that, ooh, bad guys are in and have been rummaging around, is they're often shocked to see how far back the trail of evidence goes. So it's like, yikes, they've been here for three years, and no one's spotted them until now. And so they get in, and then they set up shop, and they sit there watching what's going on.

Leo: That's why you need a canary.

Steve: That's right. That's a very good point. Another aspect that Symantec put their finger on that I was glad for, because we've looked at this, and that is that we are seeing an almost doubling in the instances of so-called software update supply chain attacks. Remember that CCleaner got hit last year where a compromised version of CCleaner managed to get onto their download server. And so even though you were actually getting it from the legitimate source, you were downloading a compromised binary. And so that's the so-called "software update supply chain," where we have all of our stuff is now updating itself all the time, and we're hoping that the servers from which we are obtaining that code are strong and have not been compromised.

But, boy, that's a powerful attack. If you can get something malicious into a popular supply chain, well, in fact, that's the way Petya managed to get into the Ukraine was through compromising a legitimate accounting software app that then updated itself and allowed Petya to get in and then spread laterally through corporate networks across the globe. So that's something we have covered and that we've been seeing.

And then, of course, mobile malware continues to be a challenge. Again, Symantec had some interesting stats, and that was that about 20%, only in the Android world because, as we know, the Android supply chain ecosystem is, by design and for the sake of some increased freedom, much less tightly controlled than iOS's chain, which Apple has a stranglehold over. In the Android world, only one in five devices are running the latest major release of Android, and only 2.3% Symantec says are up to date with the latest minor release.

So the other lesson we know is worrisome as the software update supply chain compromise is still the best advice is to keep your devices current and updated as much as possible. And that's just more challenging to do if you're an Android user. You may want to be doing it and just not have it be easy. It's a little bit like the position those of us with older laptops are in now with BIOSes not being updated for the Meltdown and Spectre vulnerabilities because they're just not going to get an update. Their manufacturers are like, eh, no, that was then. And so we can hope that Microsoft will be dealing with that.

So anyway, I thought that they had - Symantec always has some interesting numbers from their annual report, and this was no different.

So speaking of the wild future...

Leo: Yes?

Steve: There's something interesting happening, and to me it demonstrates an evolution in the industry which is good, and that is that Netflix has launched a public bug bounty program.

Leo: Oh, good, good.

Steve: Of course we were talk - yes. We were talking last week about the big and oddly time-limited, but okay, because it's big I guess they had to, Microsoft and Intel bug bounties. Remember those were for up to a quarter million dollars each if you found some previously unknown side-channel attacks. Last Wednesday Netflix announced it would be expanding its own bug bounty program, opening it to any white hat hacker and increasing their top reward to $15,000. Now, what's interesting is that there's a company, a venture-financed company, Bugcrowd, which allows any registered hackers, so I guess you register as a hacker, to scour Netflix mobile, cloud, and software platform for minor and/or critical bugs; and they'll have a value, depending upon how big they are, between $100 and $15,000.

And so Netflix is doing this through Bugcrowd, and the bounties can be used with the Netflix.com website experience, as they put it, as well as their Android and iOS mobile apps, which of course are super popular, used by 117 million Netflix users. According to Bugcrowd, the typical Netflix bug bounty payoff historically - because Netflix was using them, but it was a closed program. I remember we talked about it, where a hundred developers, like a hundred known hackers were invited into the program, and there was a low ceiling on how much they could make. And I remember kind of being, like, oh, okay.

Well, anyway, now it's wide open. Anybody who is a registered hacker with Bugcrowd can do this, and the limit is set at $15,000. And I think Bugcrowd as an entity is going to be worth us keeping an eye on. They describe themselves on their site as a "Radical Cybersecurity Advantage: Managed Bug Bounties for the Enterprise." And last month they took in a 26 million round of funding and expanded into London and Sydney.

Leo: You know what this is, this means that there are a lot of people who make this their living. This is how they make a living.

Steve: Yes. And in fact, this is exactly to my point, Leo. This seems like a good thing.

Leo: If you're good enough to do this, there are people who, you know, this is their business. They find bugs. Like Dan Kaminsky; right?

Steve: Yeah. And the whole idea of bug bounties, I think, is worth thinking about like just as a thing.

Leo: It's like hiring a hired gun. Find our flaws, yeah.

Steve: Yeah. Well, and as sort of a meme in the industry, what this suggests is that, as we have been saying, a company is inherently unable to police itself. It just - it's got management that's pushing its developers to ship stuff before the developers want to because they engaged all of their other corporate infrastructure and got committed to a release date. Or it's the problem of coders being unable to, I mean, just coders' egos being unable to see their own mistakes.

I mean, and I've talked about this often. The reason you use a debugging tool is you can stare at your own code. You know there's a bug right in front of you, and you're looking right at it, and you cannot see it. You just - you can't, until you finally single-step the debugger onto the error, and it goes [buzzer sound]. And it's like, ooh. Then finally there, when your face is just rubbed in it, and it's like, oh, I mean, I love that. I just love the experience because it just sort of brings you up short and says, okay, yeah. You did forget to zero that register. Or, yes, that just did overflow the size that you were sure would be plenty large enough or didn't even think about, or the compiler caught you out and said, oh, we're going to optimize this because who needs this extra thing. And it's like, no, no, I meant that on purpose.

So, you know, bugs come from all kinds of different places. But so what this is, is we're beginning to see the formalization of third-party external paid challenging of code. And I think this represents an important maturation of this whole cyber ecosystem that we're living in. Too often what we see are things like we've seen in the past with the Wi-Fi Alliance, where nobody challenges anything. They just proclaim something as, oh, here you go. Everybody gets a sticker on their router if they followed our specs. And then it turns out one after the other they're just a disaster.

So instead we have something like the TLS effort that has been painfully and carefully developed over time. But that's expensive. I mean, I would argue no commercial entity can afford that kind of process for something that they're trying to get to market quickly, as everybody is always trying to do. So if you're not going to do the 10-year next version, the point release of TLS, then you're just going to ship something. So it's a perfect solution to arrange for somebody paid and coordinated in a proper fashion, which is what Bugcrowd does.

Many mature enterprises would be happy, much as Netflix has said fortunately they are, to pay somewhere between $100 and $15,000, averaging at a little over 1,000, for someone finding a bug. Thank you very much. I mean, I would love it if someone found a bug and said, "Hey, Steve, here's something you missed." Yes, thank you. So anyway, I just think - so I'll bet you this is something we're going to see. I mean, we are seeing this notion of bounties more and more. I just think they make sense.

And to that end, also last Wednesday, Dropbox updated its vulnerability disclosure policy to clarify its relationship with cybersecurity researchers and also to offer, to propose, essentially, a standard for the rest of the tech industry to hopefully follow. Among other things, Dropbox has pledged that they would not bring a DMCA claim against good faith participants in their bug bounty program, which in retrospect seems obvious, but I think it's good to make it clear.

So what they're proposing, they had a series of standards: a clear statement that external security research is welcomed; a pledge not to initiate legal action for security research conducted pursuant to the policy, including good faith accidental violations. Yes. A clear statement that we, they say, Dropbox, consider actions consistent with the policy as constituting "authorized" conduct under the Computer Fraud and Abuse Act (CFAA).

Also a pledge that we, Dropbox, won't bring a DMCA (Digital Millennium Copyright Act) action against a researcher for research consistent with the policy; a pledge that, if a third party initiates legal action, Dropbox will make it clear when a researcher was acting in compliance with the policy and was therefore authorized by us. A specific note that we, Dropbox in this case, won't negotiate bounties under duress, meaning, for example, if you find something, tell us immediately with no conditions attached.

Also specific instructions on what a researcher should do if they inadvertently encounter data not belonging to themselves, so like a hold harmless policy. And a request to give us - and this is significant relative to the AMD disclosure - a question to give us reasonable time to fix an issue before making it public. They say: "We do not, and should not, reserve the right to take forever to fix a security issue." But they would like to have time to do so.

So again, I think this again moves us in the right direction. This says that, in addition to a bounty program to reasonably incentivize people who like to do this kind of reverse-engineering and want to find problems, I mean, I think this should be a career. And as we begin to see moves like this by Dropbox where they say, yes, we're going to make it clear what our policies are about reverse-engineering our stuff and attaching a reward to it; and, exactly as you said, Leo, this could become something that someone could support themselves doing if they could become good enough at it. And lord knows it's a target-rich environment out there. I wouldn't be at all surprised if you could support yourself.

Leo: Yeah. I think this is really interesting.

Steve: So Stack Overflow, as I mentioned at the top of the show, a very popular and well-known developer-centric site, basically my interaction with them is they're just a big online forum, well indexed by Google. And so it's often the case that when I'm searching for some random, like why doesn't this seem to be working, has anybody else ever encountered this before kind of thing, I'll put in a query, and up comes Stack Overflow with a voting and ranking system to sort of separate the wheat from the chaff and allow me to see what's going on. They have for eight years been publishing a developer survey where they just put out a survey to the people visiting the site saying we want to know what you guys are doing. What's important to you? What do you like? What do you wish for? What stuff are you using?

So what did they learn this year? At almost 70%, 69.8%, JavaScript has remained the most popular programming / scripting / markup language for the sixth year in a row. They've been doing this for eight years. So JavaScript, number one. Then in second, and almost in third place, at 38.8 and 34.4, Python has just bypassed C# after it passed up PHP last year. So that makes Python the number one fastest growing language. As for frameworks, at just shy of 50% (49.6%), Node.js has remained the most popular framework. And barely, almost neck and neck, 27.8% vs. 27.2, React has surpassed .NET core to enter the top three frameworks.

58.7% has MySQL as the most popular database technology, where it has been for all eight years of the survey. And SQLite dropped from number three to number five, now just shy of 20%. Rust ranked as the most loved programming language at 78.9%, with Kotlin taking second place at 75.1 and Python in third place at 68%.

I got a kick out of this: 89.9%, so just shy of 90%, has Visual Basic 6 as the most dreaded language for its third year with that dubious honor. And what was interesting was number two, okay, VB6 is number one. That puts it above Cobol in dread ranking, probably because people actually encounter it. I can't imagine anybody seeing Cobol anymore these days. But anyway, yes, Cobol would be dreaded, too. But not quite as much as Visual Basic 6.

For the second year in a row, developers chose Python as the language they most wanted to learn, but unfortunately their employers have them using JavaScript at the moment. So maybe they'll get to Python if they get a different job. Visual Studio Code and Visual Studio ranked first and second as the most popular IDEs, although looking over that category they seemed more like editors to me. For example, Notepad++, Sublime Text, and Vim were the third, fourth, and fifth places. So it's like, okay.

Android Studio was the most popular IDE among mobile developers, and Vim the most popular among sysadmins and devops. And, not surprisingly, as with years past, Windows remained the developers' primary operating system, followed by macOS and Linux, respectively. And, let's see, one, two - I have five. More than half of all developers use two monitors. Some of us use five. I just love screen real estate. It's funny, I have Lorrie set up with a large monitor now that's perched over her laptop, which she uses. And she says, "I can't imagine how I lived without this before." It's like, yeah, I know, it spoils you to have a bunch of screen. And at 16%, the number one ranked focus for developers is web.

Leo: Yeah.

Steve: So, wow.

Leo: It's not surprising, yeah.

Steve: Yeah, yeah. So there is the full survey at Stack Overflow, although Bleeping Computer, I have a link in the show notes, has a very nice breakdown from which I was able to quickly pull these numbers. So a tip of the hat to Bleeping Computer for their coverage of this.

And as I alluded to at the beginning of the show, and Leo, as I heard you talking about, no doubt over the weekend, but also just on MacBreak Weekly, the CLOUD Act, which was essentially snuck into the big $1.3 trillion omnibus spending bill which both houses of Congress passed last Thursday. I saw a picture of this thing sitting on a table next to our President, who was announcing - it was at least 18 inches high of printed paper. And I don't know why, I mean, it's horrific to imagine that that's the legislation. I think it was 2,200 pages.

Anyway, this was first introduced, this notion of the CLOUD Act was first introduced in the middle of February. And it sort of languished because, well, because we needed more time to think about it. It's regarded as the government's response to Microsoft's refusal five years ago to turn over U.S. citizens' data which was residing in their cloud servers in Ireland. So CLOUD, it's a cleverly named acronym. CLOUD actually stands for Clarifying Lawful Overseas Use of Data. And in this case we can imagine what "clarification" amounts to. The EFF has been blowing a gasket over this, as you can imagine, ever since it surfaced. They consider, and their page on this calls this a backdoor to the U.S. Constitution's Fourth Amendment, which was our protection against unwarranted search and seizure.

So essentially, as you've been saying, Leo, and I just wanted to bring it to our listeners' attention, this is now law. This CLOUD Act was slipped into this must-pass budget spending bill in order to make this law. And this effectively eliminates the need for search warrants or probable cause for obtaining U.S. citizens' data stored online overseas. And as a consequence - oh, and also it supports international cross-border treaties so that we're able to make foreign citizens' data available to other governments. And of course in a treaty they're able to make U.S. citizens' data available.

So anyway, this has got a lot of privacy watchers and security people very much up in arms and upset, although we already had this, as I understand it, this international treaty mechanism in place. Obviously it wasn't clear, thus the clarification, and Microsoft's ability five years ago to argue against agreeing to produce this data that was stored offshore. I think that in the future the law we have now says, sorry, we're not going to be able to provide that protection to our customers. Did you have anything more, Leo, about this?

Leo: No. Yeah, I mean, we've talked about it a lot, but don't know what to do. [Crosstalk] right through.

Steve: Yeah, it's unfortunate, yup. Chrome and Opera already support, as we have been discussing, built-in ad filtering. Firefox will be joining them. It's slated for the third quarter of 2018. And, boy, any Firefox enthusiasts, I would recommend checking out their Roadmap Wiki. Firefox's Roadmap is quite exciting. I mean, I guess any browser's roadmap is going to be exciting, just like looking at the things that they're working on and that they're planning to do. As I scanned through it, I thought, ooh, good. This is going to be a good year for Firefox.

Of course, many people are taking a look at it again as a consequence of their quantum technology, which really gave it a big boost in speed. What the Roadmap Wiki announces is a built-in, default-enabled abusive ad blocker, which will be appearing, as it has with Chrome, later this year, and enabled by default. Those of us who already have uBlock Origin or Adblock Plus or one of those add-ons, already have a lot of this sort of protection.

And I have to say, as I have mentioned before, I'm always amazed and a little bit chagrined when I use any browser that has zero content filtering. Depending upon where you go, some sites are virtually unusable just with all of the nonsense visually that you're assaulted by, not to mention how much slower they seem to operate as a consequence of them having to pull all this stuff down into your browser. So for the typical user who isn't already adding some sort of content filtering technology to their system, it'll be nice to have Firefox keeping pace. And do take a look at the Mozilla Roadmap if you're a Firefox enthusiast.

The concern over homograph attacks is growing because the use of the attacks is growing. Homograph attacks we've talked about before. They leverage the use of Unicode in domain names. Now, the intent is to support large character set Unicode to be able to allow the display of non-ASCII-based domain names. The problem is that, as soon as you allow Unicode, you open yourselves and users to spoofing and fraudulent domain names. There are all kinds of examples.

But basically Unicode, due to the fact that there are so many lookalike characters, it's possible to create, I mean, we're always giving PayPal as an example. There are, for example, characters with, like, little dots below them that you might not even notice that is a different domain name. Well, if the browser shows you the Unicode version, what you see looks like PayPal and isn't. So browsers have typically solved this problem by showing you, and the name is sort of a play off of Unicode, they have Punycode because there is an ASCII representation of Unicode using standard ASCII. It uses hyphens and numbers in a sort of a weird ASCII construction.

The point is that, if you show the Punycode rather than Unicode, there's no way anyone is going to be confused. PayPal with some Unicode rendered as Punycode looks nothing like PayPal. So Edge, for example, Microsoft's Edge browser, does not render Unicode in the address bar. It shows you the ASCII Punycode. Chrome has kind of a compromise. They show the ASCII, that is, the Punycode, in the title bar, but I guess to be a little more friendly they show the Unicode in the URL address bar, which is a little worrisome to me with Chrome being the majority browser and so many non-savvy users. I mean, if you didn't look in the title bar for the page and see something crazy, the Punycode, then you could easily click okay before realizing you are not at the proper domain.

The good news is there is an add-on for Chrome produced by Phish [P-H-I-S-H] dot AI, called IDN Protect. And IDN is International Domain Names. So if you google "Phish.AI IDN Protect," you'll get a link to their Chrome add-on in the Chrome Store, and also you'll see their GitHub link because the GitHub page for this add-on is also there. Firefox does show, unfortunately, and I must have talked about this before on the podcast because when I went to check it I had flipped the default. By default, Firefox will show Unicode, but you can change that.

So as always, go to in Firefox about:config and just search for "puny," P-U-N-Y, and you'll get one item will survive the search weed-out. And mine was already dark, meaning that it was the non-default setting. It had been set to dark for IDN_show_punycode. And that's what you want. I can't imagine, I mean, unless you're a very careful international web user who really wants to see things in Unicode, the danger is big. Edge won't show it to you at all. Chrome won't show it to you in the title bar. It will in the address bar, unfortunately. You can disable it in Firefox.

And I would argue, if you want to protect your friends and family, consider this IDN Protect. What it does is it will detect if a domain name has Unicode and just put up an intercept, just to bring your attention to the fact that the page you are at has a Unicode domain name. Inspect it carefully. Make sure you're where you want to go. We are seeing more instances of this being used for spoofing. And anybody can register a domain name. Anybody can now easily get an HTTPS certificate for it. You'd have a hard, well, you wouldn't be able, probably, to get an EV cert; but you could certainly get a TLS connection showing a secure connection, so all of the other window dressing that makes someone feel secure, yet they really shouldn't be. So I would argue that, if you're a Firefox user, turn off showing Punycode for Firefox and consider this little Chrome Store add-on just to make sure you notice if you're going somewhere you don't expect to be.

And Leo, I know you talked about this before. I just did want to make a note for our podcast that Facebook was found - of course they've been much in the news lately over the whole Cambridge Analytica mess with the election. But it was discovered by some users who downloaded their Facebook history that, to their absolute surprise, a complete log of their use of SMS texting and telephone log was part of what Facebook was collecting. And really it looks like Facebook has not been forthcoming about this. Facebook argues that they always asked for permission. But Android did not require apps to get permission initially. They then have successively tightened that up.

I mean, all the evidence suggests that Facebook was acquiring this data without their users' permission. A number of very security and privacy-conscious users are absolutely sure they never gave permission, and in some cases installed the Facebook Messenger app several times, always being careful about what permissions they gave, yet Facebook was collecting this data without permission. So anyway, just a heads-up to users of Facebook on Android, Facebook Messenger on Android, that this had been going on. And I don't know if it makes sense to, if you can, purge it? I know that, Leo, you have famously just proactively deleted your Facebook account.

Leo: I undid it, though. I had till the end of the month. And you know why I undid it? My wife guilted me into it. She said, "Because you've deleted your Facebook account, it now says Lisa Laporte is married, but not to whom." And she wants it to say to me.

Steve: Well, that's sweet, I guess.

Leo: Yeah, how can I say no?

Steve: But she would be married to someone Laporte.

Leo: No, no. The name disappears.

Steve: Well, but Lisa Laporte.

Leo: Well, yeah, I know, but I don't - it's really interesting. So this is how normal people react. Your whole life, her whole life history she feels has disappeared; right?

Steve: Wow. And were you able to, like, expunge all of this other stuff? I mean, so you did this to make a statement.

Leo: Yeah, because they already - I know they have everything they want, and they've had it for a long time.

Steve: And Leo, you're not a big mystery to the world.

Leo: No, I'm not anyway. But I feel like, yeah, I want to send a signal to Facebook that this isn't okay, and then by extension to everybody else that there is a price to be paid, that consumers aren't pushovers here. And so I really did want to do this. But you know what they say: "Happy wife, happy life." I don't care that much. I care more about Lisa. But there's no way of verifying that they deleted everything. I mean, who knows what they do? We know they lie.

Steve: Yeah. Yeah.

Leo: I don't know. You know, it didn't matter that much to me, but I did want to send that signal. On the other hand it's - obviously more people join Facebook every day than they'd ever possibly lose. So it's kind of a hopeless thing.

Steve: Well, and there's all this talk now about regulation, about social media.

Leo: Yeah, we're glad to see that.

Steve: And so we don't know what's going to come of it. But it really, I mean, it is a big deal. And even Marcus being quoted saying, "Yes, I think we should be regulated," it's like, oh, okay.

Leo: Yeah, well, what he's really saying is just a little bit. A little bit. It's prudent to ask for some regulation. Please sir, hit me.

Steve: So last Wednesday they put out a deliberate one-week notice of what they called a "highly critical release" to go public on March 28th, which is tomorrow. And so they wrote for a description: "There will be a security release of Drupal 7.x, 8.3.x, 8.4.x, and 8.5.x on March 28th, 2018 between 18:00 and 19:30 UTC, one week from the publication of this document, that will fix a highly critical security vulnerability. The Drupal Security Team urges you to reserve time for core updates at that time because exploits might be developed within hours or days."

Leo: Because Drupal culls its PHP code. As soon as there's an update, they start looking at it; right?

Steve: Yup. Security release announcements will appear on the Drupal.org security advisory page. They said: "While Drupal 8.3.x and 8.4.x are no longer supported, and we don't normally provide security updates for unsupported minor releases, given the potential severity of this issue we are providing 8.3.x and 8.4.x releases that include the fix for sites which have not yet had a chance to update to 8.5.0. The Drupal security team strongly recommends the following."

They said: "Sites on 8.3.x should immediately update to the 8.3.x release that will be provided in the advisory, and then plan to update to the latest 8.5 security release in the next month. Sites on 8.4.x should immediately update to the 8.4.x release that will be provided in the advisory, and then plan to update to the latest 8.5 security release in the next month. Sites on 7.x or 8.5.x can immediately update when the advisory is released using the normal procedure."

Oh, and they did note that this will not require a database update. They said: "The security advisory will list the appropriate version numbers for all three Drupal 8 branches. Your site's update report page will recommend the 8.5.X release, even if you are on 8.3.x or 8.4.x. But temporarily updating to the provided back port for your site's current version will ensure you can update quickly without the possible side effects of a minor version update."

So they said: "The Security Team or any other party is not able to release any additional information about this vulnerability until the announcement is made. The announcement will be made public at," and then they give the URL www.drupal.org/security, "over Twitter and in email to those who have subscribed to our list." So no further details, but they couldn't be making anything more clear. I mean, there's nothing they could do to drive home the point more clearly that this has to, you know, that anyone using Drupal tomorrow - what, you said between 18:00 and 19:30 is around noon in Pacific time?

Leo: Yeah, I think so. Noon or 11:00. Noon or 1:00, I mean. But, yeah, I talked to Patrick. He's all over it. He knew it all the time.

Steve: Ah, he's cool.

Leo: He's going to be sitting there, his sandwich in hand, ready to push the button.

Steve: Finger on the button, finger on the button; right.

Leo: It doesn't give you a lot of time, does it. I mean, it's like...

Steve: No, no.

Leo: ...wow, it's a race between you and the bad guys.

Steve: And the fact that there has been a week means that the bad guys are also - they're sitting here clicking refresh on their browser, waiting to get the information.

Leo: They're staring at the code. They're going, what is in there? What's in there?

Steve: Yes. And what we absolutely know, in fact they probably have their diff engines all set and primed to, like, to immediately process. Now, what we absolutely know is many sites will not do this. Our listeners are. You guys are ready. Responsible sites. But, yes, it is the number two CMS on the 'Net, and there are many people who had somebody else set their Drupal up some time ago, and then they wandered off. They were subcontractors. It's not being maintained. So I'll just bet you next week or two weeks or three weeks, if not all of those, we'll be talking about exploits because this sounds like they're going to be - they have no choice but to change their system in a way that bad guys could leverage. It's clear that that's what they're worried about.

Leo: We'll do live coverage, live streaming coverage of Patrick patching the server.

Steve: So some good news on the DNS front. We've been talking about it a lot because, as we know, DNS is sort of the laggard within the security domain of the Internet. It is by default over UDP, so it is spoofable. It is interceptable. There's no encryption available, unlike with TCP connections where we can use TLS tunneling. There is DNSSEC, but DNSSEC is signing. It's not privacy. So even when we had DNSSEC to secure the DNS, as it's called, the Domain Name System, we still don't have privacy. That's not part of it.

So it turns out that there is an effort underway which has sort of been done, I mean, it's not private because it's an IETF draft process which has been moving slowly towards ratification as IETF draft processes do. But I'm bullish about this. To me, this makes sense. The bad news is the acronym. Technically it's called Trusted Recursive Resolver via DNS over HTTPS. Unfortunately, they only kept the last three, DNS over HTTPS, so it's DoH. Yes, it's D-O-H is the acronym. The TRR, the Trusted Recursive Resolver, they decided that was to heavy for them to drag along. That would be TRRDOH. Instead we just get DoH. So this is the DoH protocol.

The good news is this is an experiment for, well, which is moving through draft, for essentially allowing a client, like a web browser, to establish a semi-static TLS connection to a DNS over HTTPS. In other words, you bring up a secure tunnel. You maintain the tunnel for the duration of your use of the client. So when you launch Chrome or you launch Firefox, the browser would establish a tunnel, so it would perform the authentication and the handshaking and establish security with a provider once and leave that connection up.

Remember that TLS and TCP do not require any continuous packet traffic in order to maintain a static connection. There will probably be plenty. If you're using your browser, you're always doing DNS lookups. But the idea is that this would solve, very much like a VPN, but without any of the overhead. When you think about it, we've pretty much got HTTP now moved to HTTPS so that we've got authentication so we have security, and we've got encryption so we have privacy for our individual web browsing actions. But unless we're using a VPN, as we mentioned last week, even some VPNs it turns out are not tunneling DNS for the system. If they're only a VPN for your browser, then DNS lookups still go out the old-fashioned way, so you don't have browser-based security. And essentially just understand this represents a huge privacy problem. Anybody observing, passively observing TCP traffic, I'm sorry, UDP traffic, Internet traffic, is able to see by default everywhere you go because your browser is asking for the IP for a domain name, and that lookup process is occurring.

So where we stand is we know that Google has announced and is running an API that's supporting DNS over HTTPS, with Google's public DNS service. Mozilla has teamed up with Cloudflare, and they're conducting a pilot test which will begin, I believe, with the next version of Firefox, which would be v60. Matthew Prince, who's the cofounder and CEO of Cloudflare, was interviewed by Threatpost over all of this, and he was quoted saying: "DNS is a 45-year-old protocol. It was never built to have encryption in it or to be secure. Yet, DNS acts as the Internet's vital directory and is vulnerable to many different types of abuses. This is a privacy nightmare, this is a security nightmare, and this is a performance nightmare; and yet it's the foundation of the Internet."

So what Firefox will be doing is with v60 bringing up optionally this Trusted Recursive Resolver, TRR. And in fact that's the way to find it in the about:config for Firefox as soon as you have 60. It'll be in the nightly builds, and at this point not enabled by default. So you'll need to go to about:config and then search for "trr." That'll bring up all of the options that allow you to turn that on. And Ghacks.net site has a very nice write-up of all of the Firefox settings. There's a whole bunch of different options.

My feeling is this is where we're going to end up. There are privacy worrywarts who are concerned that all this does is move the privacy problem, like it relocates it to the endpoint, to the supplier. I would disagree with that. I think that this is enough of a problem not to have DNS protected to allow it to be so spoofable and so visible by default that it is a win to incorporate a DNS tunnel into DNS lookup for our browsers. And, for example, the connection with Mozilla and Cloudflare is not meant to be the way Mozilla's always going to have it. I would imagine it will be an option.

But the presumption is there will be many different providers of DNS over HTTPS, and that individual users will be able to decide, certainly in the case of Mozilla and, who knows, I would imagine even in the case of Chrome. I would imagine, since Google offers

DNS over HTTPS by default, they will be the default provider for their own browser. But certainly users will probably be able to have that as a configurable setting.

So anyway, I just wanted to put this on everyone's radar. This feels like the solution that's going to win. Certainly we know about OpenDNS and their own solution for using DNSCurve in order to encrypt and provide both privacy and security. That's always had the feeling of a closed standard. And they may very well, I wouldn't be at all surprised, as an example, OpenDNS was another provider of the IETF coming standard DNS over HTTPS, or DoH, as I guess we're going to have to get accustomed to calling it. So it looks like there will be a way before long for us to have secure DNS, as well.

A bit of miscellany: Last Saturday I decided to take a look at "Star Trek: Discovery." I had deliberately eschewed it until now because it was only available for pay on CBS All Access, which really means paid access. Lorrie and I watched five episodes in a single sitting.

Leo: Because you only have a week; right? So you're going to try to...

Steve: Because you have a week, exactly.

Leo: ...watch the whole thing in one week.

Steve: Oh, we will do another chunk tonight and another. Because she's been out of town for the last two nights. Otherwise it would already be done. I just wanted to say it really seems good. This is not Jean Luc's prime directive-driven universe. It's set 10 years ahead of Kirk and Spock when the Klingons have not yet been brought into the Federation, and they're not happy. The captain of the Discovery is sort of a get-the-job-done renegade. And so far I can only vouch for the first five episodes, but we loved it.

Leo: Hmm. I haven't watched it for the same reasons. I didn't want to - and I saw mixed reviews on the first few episodes. Does it get better, do you think?

Steve: I think it does. I really - I'm excited. And of course everyone knows I'm really excited about the April 13th release of "Lost in Space," which looks great. But I think this had, I think it was 13, maybe 15, I don't remember now how many episodes total. But it seemed to have a nice first season. And so, yes, we'll be watching it during our seven-day free trial and then saying, uh, thank you very much.

Leo: I love it that they got the original theme in there at the end. That's awesome.

Steve: Yeah. Yeah. And again, Leo, I'm giving it a full double Vulcan salute, you know, thumbs up because...

Leo: All right. Well, you've got me, then. You think you'll keep paying for it? How many episodes do you have to watch in seven days?

Steve: I think 13. I think it's 13, might be 15. There were a bunch. But believe me, that's not a problem. I mean, it's really good. We will be watching it tonight and tomorrow night.

Leo: How are you watching it? Is there a Roku app?

Steve: I'm a Fire TV person because I can't stand the Amazon controller. So I'm sure there is a - oh, and if you have Amazon Prime, it is available through Amazon Prime through a connection with CBS All Access.

Leo: Oh, but you have to sign up for CBS All Access, and then you can do it.

Steve: Well, I just pressed a button on Amazon, and it sort of did it through Amazon. So it was all - it was very transparent.

Leo: Oh, yeah, I could do that, yeah. Yeah, that makes it too easy.

Steve: It is very easy. And then you've just go to go into Amazon Prime and...

Leo: Just forget you're paying for it, yeah.

Steve: Yeah, exactly, and then say no thank you because it's $10 a month after that. And there's nothing else that I want to watch. Maybe there'll be other things that people will want to watch. I mean, remember we were all fans of "The Good Wife," which was then extended by CBS as "The Good Fight," and that's also under paid access. But, eh, it's not that good. So anyway.

And lastly, before we talk about TLS v1.3, I wanted to mention the Disk Cleanup app in all of our Windows versions. Disk Cleanup has been there forever. And I have recently been running Disk Cleanup on my 7 and 8 and 9 - well, not 9, there's no 9 - 7 and 8 and 10. And it's finding 15GB of stuff to clean up.

Leo: Wow.

Steve: Yeah.

Leo: Now, do you have to do the cleanup system files to get that stuff? Or is that just in the...

Steve: Yes, yes, you do.

Leo: Oh, okay. That's the key is press that button.

Steve: Exactly. So everybody, put Disk Cleanup into your search bar or your Cortana or whatever, find it. And then, when it comes up, down in the lower left is a Cleanup System Files. Press that and wait a while, and then go and turn all the checkboxes on. Now, again, of course, this is going to remove your ability to back out of where you currently are with Windows Update and Windows versions.

Leo: Because you're getting rid of restore points and stuff like that.

Steve: Yes, yes. But oh, my god, I mean, it's huge amounts, well, not only of space, but of time. So don't do this if you intend to use your system again that week. Well, depending upon what day it is. No, seriously, it's an overnight thing. So at the end of the day, start that process. You can see how much space you're going to get back. My biggest was 15GB of space. So that's not nothing.

I just wanted to kind of put it on people's radar that you don't have to use a third-party tool. You don't have to do anything sketchy. It's Microsoft's own thing that's always been there. You do need to click up Cleanup System Files to get all the biggies. And don't do it unless you're sure that your system is stable and you won't be rolling back to anything previous. But there's a lot of space being taken up unless you deliberately capture it, recover it yourself.

And finally, TLS v1.3. We've talked about it a number of times. We were talking about it just a couple weeks ago because we were getting close to ratification. That did happen a couple days ago. It is final. And this of course is the next point release of Transport Layer Security, which is the successor to Secure Socket Layer, SSL, which we have had until TLS came along, which provides authentication with the whole much controversial CA-issued secure server certificates, which are signed by certificate authorities that allow our browsers to verify that we are connecting to the domain, the owner of the domain we believe we are, and through that process negotiate encryption between the endpoints so that we get privacy.

Just the tip of the specification says, under Document Quality, and everyone will understand why I'm reading this in a second, they said: "There are over 10 interoperable implementations of the protocol from different sources, written in different languages. The major web browser vendors and TLS library vendors have draft implementations or have indicated they will support the protocol in the future. In addition to having extensive review in the TLS working group, the protocol has received unprecedented security review by the academic community.

"Several TRON" - which is the abbreviation for TLS Ready or Not - "conferences were held with academic community to give them a chance to present their findings for TLS. This has resulted in improvements to the protocol. There was also much consideration and discussion around any contentious points" - and we know what those were - "resolved through polls and working group last calls. Please note that ID-nits complains about the obsoleted/updated RFCs not being listed in the abstract. This is intentional because the abstract is now a concise and comprehensive overview and is free from citations."

So again, as I said earlier, this is the way this should be done. And thank goodness for something as critical as this it's the way it was done. We didn't start this way. There really never was an SSL 1.0 because Netscape created the spec, and it was so troubled that it was never released. 2.0, despite their best efforts, it was basically a single-shop operation, and it had lots of problems. So for a core protocol like this, this is the way it needs to be done. And as unfortunately the Wi-Fi Alliance's recent announcement demonstrates, they still don't have a clue. They're still not getting it right. So I imagine we'll be talking about future WiFi security problems in the future. Probably not any more TLS problems for some time.

So TLS 1.3 brings us a bunch of things. We get improved cryptography by completely dropping support for earlier and now formally obsoleted hashing algorithms like MD5 that still had a presence in cipher suites that might have been acceptable. No more. It also adds support for much stronger modern alternatives such as the ChaCha20, Poly1305, the Edwards Ed25519 Curve, the x25519 Curve - those are the elliptic curves, actually those are the ones which I adopted for SQRL's use - and x448. So much stronger crypto.

1.3 also supports quicker initial handshake connection negotiation between the client and the server, cutting down on the required round trips, as I have referred to before. So HTTPS over TLS 1.3 will no longer be slower than HTTP for that reason, and that removes one of the arguments against going to HTTPS, saying, oh, there's more overhead and it's slower. No, not any longer.

1.3 also supports new features to reduce the time needed to establish encryption handshakes with hosts to which the client has recently connected. So that technology has been nailed down, meaning that, if you tend to be going back to Google a lot, you're able to just essentially pick up where you left off, even if you had dropped a connection for a while. 1.3 also brings stronger protection against downgrade attacks, which is one of the things we've most spoken about in previous versions of SSL. It was often possible, because the client offered the server a list of protocols, suites, cipher suites that it understood, and the server was able to choose among those.

There was no provision in the earlier versions of these protocols to prevent intercepting the client's list and only using the weakest of the things it was offering, which would cause the server to sort of shrug and go, uh, okay. Anyway, can't do that anymore with TLS 1.3. We've solved this problem of downgrade attacks finally. So additional security as a consequence.

And the one big remaining question was what about perfect forward secrecy? Despite the efforts of the financial business sector that seemed to be the most well-organized group fighting against the adoption in 1.3 of mandatory support for perfect forward secrecy which, as we know, would protect us against the future disclosure of a server certificate which we're not protected from at the moment. The idea would be, if you captured TLS traffic that you could not decrypt today, but you then in the future managed to obtain a certificate, even an expired retired certificate, from a server that had been involved in that initial negotiation, you were able to then come back and decrypt traffic from the past that you had captured. That's lack of perfect forward secrecy. That is not possible to do under TLS v1.3.

So this is going to mean that there will still be laggards who are forcing TLS 1.2 because they don't want to update their technology in their IT infrastructure to support 1.3. But time is going to move forward; and at some point we're going to see, as we have seen in the deprecation of 1.0, we are now using 1.1 and 1.2 and 1.3. At some point 1.1 is going to go away, and then at some point 1.2 is going to go away. And the world will then all be using 1.3 with perfect forward secrecy as a requirement. So browsers like Chrome, Edge, Firefox, and Pale Moon have already rolled out support for earlier versions of the

TLS 1.3 draft; and they will be updating if necessary, if any update is necessary, updating their support to the final draft standard now that we have that.

And as I mentioned also at the top of the show, I did a little bit of a timeline. We got SSL 2.0 in 1995 because 1.0 never really happened. A year later we went to SSL 3.0 Three years later we went to TLS 1.0, that is, 1999. That's how long TLS 1.0 was ratified, 1999. Then TLS 1.1 came in 2006 after seven years. So we lived a long time with TLS 1.0. TLS 1.2 relatively quickly fixed some problems because we were getting better with security with 1.1. So it was only two years later, in 2008. Yet that was a decade ago. 2008, 10 years ago, is when we got 1.2. And so now 10 years later we are moving to 1.3.

So my feeling is this is probably going to, I mean, it'll be very surprising if there are any showstopping problems in 1.3. This thing, first of all, it is a mature evolution of 1.2 that we've lived with for 10 years. We've learned a lot from 1.2. We have new crypto. We've got new ciphers. We've got improvements in the protocol. So basically 1.3 wraps up a decade of intense understanding and learning and evolution about how to solve this problem, and makes it a standard. We're not all using it today. There will be places that we cannot connect to over 1.3. But it is the future, and at least we now have a standard against which to implement all future technology. So, yay, that's how the process works.

Leo: Nice, very nice. And if you don't mind, I'd like to share, as we close out the show, a tweet from Edward Snowden that he just recently put up that talks about a 2009 interview, actually it's a little clip from a 2009, what would that be, almost 10 years ago, interview with Mark Zuckerberg in the BBC. And it's only a 30-second clip, but I think this will - I think you'll enjoy this.

[Clip]

BBC: So who is going to own the Facebook content, the person who puts it there, or you?

ZUCKERBERG: The person who's putting the content on Facebook always owns the information. And that's why this is such an important thing and why Facebook is such a special service that people feel a lot of ownership over; right? This is their information. They own it. They often want to...

BBC: And you won't sell it?

ZUCKERBERG: No, of course not. I mean, they want to share it with only a few people.

BBC: So just to be clear, you're not going to sell or share any of the information on Facebook.

ZUCKERBERG: What the terms say is just we're not going to share people's information except for with the people that they've asked for it to be shared.

[End clip]

Leo: Okay. Just, you know, so you know what the truth is about Facebook. Ahem.

Steve: Ah, interesting. Interesting.

Leo: That was nine years ago.

Steve: Interesting.

Leo: No, no, we won't sell it. We won't sell it. Never sell it.

Steve: Interesting.

Leo: No. Why would we sell it? What possible reason would we have to sell it? We just give it to you.

Steve: Yeah, that's a little blast from the past. How things change.

Leo: How things change. Steve Gibson, nothing seems to change with Steve. He's still here at 13 years, and we hope he goes on another 13 years. A lot of people worry about this 999 thing. Just keep it in mind, we can give you four digits. If needed, we can give you a digitectomy. Or what's the opposite of a digitectomy? We'll give you an added digit, and you can go on as long as you want because we sure want you to keep going on.

Steve Gibson's website is GRC.com. That's where you go to find his SpinRite, the world's best hard drive maintenance and recovery utility. If you have a hard drive of any kind, you need SpinRite. While you're there, check out all sorts of good stuff he's got up there, SQRL and Perfect Passwords and Password Haystacks and ShieldsUP!, which is probably in the billions by now, billions served. I don't know, the number goes up all the time. And of course this show. You'll find it there, as well, including handwritten transcripts by the fine Elaine Farris - she does a great job of translating geek into English - and MP3 audio of the show. If you want video of the show - and people do want to see you, Steve. They want to see the 62-year-old moustache.

Steve: Yeah, until they have one.

Leo: You can come to our site, TWiT.tv/sn. You can watch live. We do the show every Tuesday, 1:30 Pacific, 4:30 Eastern, 20:30 UTC. That would be at live.twit.tv or TWiT.tv/live. We have two pages depending on which player you want to use. You can also watch on other services. We're all over the place. If you do watch live, though, join us in the chatroom, irc.twit.tv. You can be with the kids in the back of the class throwing spitballs, stuff like that.

We have on-demand versions of audio and video. Steve's got his; we've got ours. And the best thing to do, because it all comes from the same place, ultimately, would be subscribe in your favorite podcatcher. That way you get every episode. You can start building your collection of Security Now! shows. Steve, have a great

Tuesday afternoon. I know where you'll be, seated on the couch this evening.

Steve: Yes, sir.

Leo: Enjoy your trip through space, and I'll see you next week.

Steve: I will do that, and I'll have a full report next week on what finally happened because there's no doubt that we will be powering through.

Leo: I've got to watch it. I'm going home tonight.

Steve: I recommend it. It is really - it is gritty. It's wonderful.

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

Steve: Okay, my friend. Bye.