SECURITY NOW!

Transcript of Episode #535

Listener Feedback #223

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

High quality (64 kbps) mp3 audio file URL: http://media.GRC.com/sn/SN-535.mp3
Quarter size (16 kbps) mp3 audio file URL: http://media.GRC.com/sn/sn-535-lq.mp3

SHOW TEASE: It's time for Security Now!. Steve Gibson is here. We've got questions, we've got answers, and he's going to talk about the latest news, including, yes, the Dell certificate fiasco. It's all coming up next on Security Now!.

Leo Laporte: This is Security Now! with Steve Gibson, Episode 535, recorded Tuesday, November 24th, 2015: Your questions, Steve's answers, #223.

It's time for Security Now!, the show where we talk about your security and safety online. And thank goodness, I thank goodness every day that Steve Gibson's out there, helping us understand this and do the best we can. Steve Gibson, our host for the last 10 years of Security Now!. Hi, Steve.

Steve Gibson: Well, the first and last 10 years of Security Now!.

Leo: And we'll continue as long as, what is it, the good lord willing and the creeks don't rise.

Steve: And there seems to be no slowdown in things for us to talk about.

Leo: Oh, man. Yeah.

Steve: So we have a Q&A this week. This is Episode 4 - sorry, 534, I was looking at the last digit, 534, Q&A #223. Lots to talk about, in addition to great questions from our listeners. We of course will cover the story about Dell stepping in it big-time.

Leo: Oh, boy.

Steve: And then answer the question, has LastPass stepped in something, too?

Leo: Uh-oh.

Steve: We'll cover Windows 10's various recent struggles briefly. There is an interesting document, 43 pages, that I'm just going to summarize briefly, from the Manhattan DA about their position on smartphone encryption...

Leo: Oh, I can imagine.

Steve: ...that I want to cover. We got a bunch of updates and miscellaneous fun stuff. An important piece of errata from something I got wrong a couple weeks ago. And then a great Q&A. So lots of good stuff.

Leo: A jam-packed show for you. Can't wait. I've got the questions, Steve's got the answers, and a lovely cup of joe, it looks like. That looks good, whatever you just sipped from. Let's get into the security news.

Steve: So our Picture of the Week is our top story, or at least our first story. And this is a picture that someone was nice enough to tweet from their own Dell machine, which shows a certificate that was issued by eDellRoot, issued to eDellRoot. In other words, it's self-signed. Interestingly enough, it's valid from 4/7/2015, so it's been around for a while. They weren't using any of the benefit of expiration, so their own certificate which they created expires on New Year's Eve of 2039. So they intended that thing to sit around in the Dell machines forever.

And the other interesting thing to note there is that certificates always have restrictions on what they're permitted to be used for. And this certificate has no restrictions. For example, GRC's certificates say it can be used for authenticating the identity of a web server or authenticating the identity of a web client. This is all issuance policies and all application policies. So it could also be used for code signing.

So anyway, so here's what happened. Essentially, this has caused a huge concern in the industry because this is another instance, and we've covered this from so many different directions in the past, another instance of some third party putting their own certificate authority certificate in the root store of the machine. And in this case, the private key is also available, and that's the piece that is normally protected with all the power a certificate authority has because what they're doing is they're, well, I guess we've covered the whole certificate PKI infrastructure over and over ad infinitum because it's something that we depend upon to such a degree.

But so what this is being compared to is Superfish, that of course we covered in the past. Then we've also seen various antimalware packages installing their own certificates. All of these are bad. However, there is a degree of badness to them. For example, if each antimalware package generates a certificate and then plants it in the root store for the machine, then at least all the certificates are different. In this case, all of the Dell machines that have this, have the identical certificate, which means they have the identical public key and private key. And that makes it trivial to break into their communications and to essentially intercept what they're doing, raising no alarms or warnings of any kind.

So, for example, Ars Technica's coverage, Dan Goodin says: "Dell does a Superfish, ships PCs with easily cloneable root certificates. In a move eerily similar to the Superfish debacle that visited Lenovo in February, Dell is shipping computers" - oh, and by the way, notice this is two months later. Two months after the Superfish debacle, Dell said, oh, let's do that.

Leo: Well, I do wonder how this happened. But probably whoever decided to do this said, oh, but we're not going to use it for advertising, as Lenovo did. We're just using it for support so we can get your Dell ID tag.

Steve: Well, yes, technically.

Leo: That's harmless; right?

Steve: Right. So Ars says: "Dell is shipping computers that come preinstalled with a digital certificate that makes it easy for attackers to cryptographically impersonate Google, Bank of America, or any other HTTPS-protected website." Blah blah blah.

I'm going to skip down to Robert Graham of Errata Security, who is a well-known expert in the industry. He says in his blog post, "It was discovered this weekend that new Dell computers, as well as old ones with updates, come with a CA, eDellRoot, that includes the private key. This means hackers can trivially eavesdrop on the SSL communications of Dell computers. If I were a black-hat hacker, I'd immediately go to the nearest big city airport and sit outside the international first class lounges and eavesdrop on everyone's encrypted communications. I suggest international first class because, if they can afford $10,000 for a ticket, they probably have something juicy on their computer worth hacking.

"I point this out in order to describe the severity of Dell's mistake. It's not a simple bug that needs to be fixed, it's a drop-everything and panic sort of bug. Dell needs to panic. Dell's corporate customers need to panic. Note that Dell's spinning of this issue has started, saying that they aren't like Lenovo because they didn't install bloatware like Superfish. This doesn't matter. The problem with Superfish wasn't the software, but the private key. In this respect, Dell's error is exactly as bad as the Superfish error."

So several things have happened. Among them, there are two exploitability test sites, the links for which I have in the show notes. One is https: - you have to remember that you're trying to test a certificate, so you want to make a secure connection - https://bogus.lessonslearned.org.

Leo: A great URL, by the way.

Steve: Yes. And so this is a...

Leo: Now, I'm doing it on my Dell, which I already know, because this came out in January, and these Dells were only infected starting in August. Although you said there was an update. I never got a update that added that root certificate. But this is what you want to see; right?

Steve: Correct.

Leo: There's a problem with this website certificate.

Steve: Right. And then the other one is edell.tlsfun.de. And that's a little more user friendly. That's the link that our listeners are going to want to send to their friends who they know have Dell laptops because it gives you a yea or nay and some explanation and background and other links and things: edell.tlsfun.de. So those are two sites which have popped up to - essentially, they have synthesized their own bogus certificates, which no one's normal correct root store would honor.

And, by the way, Firefox won't honor it. In fact, when I tried to go there, that site said, well, you're using Firefox, so you were never in any trouble, even if you're using a Dell laptop. In fact, that's one of the short-term mitigations, although we already have Dell's permanent one I'll get to in a second, is simply use Firefox. As we know, Firefox brings along its own root store because it has its own security library system as part of it.

Leo: Is this not the case for Chrome?

Steve: No. It's really interesting. It's the certificate security portion of Windows that for some reason Chrome shares. And you can see that because, if you inspect the certificate in Chrome, you get Windows dialogues that are identical to what IE or Edge will show you because Chrome is actually using a Windows API. So it didn't do its own. It's sharing Windows's. So the risk is IE, Chrome, and Edge on a Dell laptop or actually any laptop with that certificate, but presumably only Dell.

Leo: Yeah. I don't think anybody else is going to put in eDellRoot on theirs.

Steve: Well, I mean, the problem now is bad guys could, or people wanting to play around could stick that on your machine.

Leo: So make that clear, because they all use the same private key, and that private key is revealed, stupidly, in the certificate. Anybody could do a man in the middle and say, with this fake certificate, say "I'm Bank of America, baby."

Steve: Well, correct. And so, for example, an arguably legitimate, although worrisome, use is a corporate firewall that insists on scanning into secure connections. A growing number of corporations are doing this because HTTPS is becoming more prevalent. So there will be, on the perimeter of the corporate network, there will be an intercepting proxy which will disallow any secure connection that it can't see into. And so part of what the corporation has to do, every computer inside that network will have a certificate from that appliance. And so it's the certificate from that appliance which has been added to the normal collection of certificates that gives that appliance the unique ability to intercept and inspect all the traffic leaving and coming into the corporate perimeter.

And so the point is that, because all of these Dell laptops were including this certificate with the private key, it would be possible for anyone to act just like that kind of deep packet inspecting filter anywhere on the Internet and intercept any Dell communications, decrypt it, look at it, and then reencrypt it, and it would provide no warning to the user.

Now, the coolest utility is one that we've talked about before. And that's - I've got the link also in the show notes. It's Windows Apps by FS1, and the app is a tiny little 40K executable for Windows. It's at trax.x10.mx/apps.html. We've referred to it before. It went away for a while and came back. What this does, this RCC tool inspects your certificate store for anything that shouldn't be there and provides you with a little audit. So very handy. Anyone running it on a Dell machine, this thing would immediately, even before this was known publicly, would have raised a flag and said, wait a minute, what's this eDell certificate sitting around doing?

But anyway, Dell has, and again, a link in the show notes here, their official remover has now been issued. And I've got the link here. It's too long for me to stick into the podcast audio, but it's eDellRootCertFix.exe. What Dell said yesterday in a CYA-style response is they wrote: "Today we became aware that a certificate (eDellRoot), installed by our Dell Foundation Services application on our PCs, unintentionally introduced a security vulnerability. The certificate was implemented as part of a support tool and intended to make it faster and easier for our customers to service their system." Makes it easier for anyone to service your system.

"Customer security and privacy is our top concern and priority for Dell. We deeply regret" - I'm sure - "that this has happened and are taking steps to address it. The certificate is not malware or adware." Okay, big comfort. "Rather, it was intended to provide the system service tag to Dell online support, allowing us to quickly identify the computer model, making it easier and faster to service our customers. This certificate is not being used to collect personal customer information. It's also important to note that the certificate will not reinstall itself once it is properly removed using the recommended Dell process." However, note: If you simply remove it from the certificate store, the Dell service will see that it's gone and reinstall it.

Leo: Oh.

Steve: So you have to - yeah.

Leo: Thanks, Dell.

Steve: It comes back.

Leo: They could have done this without doing - I don't even understand why you do a self-signed certificate if all you want to do is identify the machine's tag.

Steve: Se, that's just it. I haven't had a chance, because I've just been...

Leo: [Crosstalk], don't you think?

Steve: ...pulling all this together. But this really, this seems odd to me. The only, well, the only thing I could imagine is that they've got a server somewhere, or some automated appliance.

Leo: So it's almost like you're logging into a VPN kind of situation.

Steve: Well, or there's something autonomous that they set up where they want to have a secure connection. And rather than simply buying a certificate for it from any of the existing certificate authorities, in which case all of the Dell computers could connect to that with no trouble, they got fancy. And they said, well, we don't want to, we don't need to buy one of those nasty certificates from someone else because then we'll have to renew it every couple years, we'll have to pay the money, blah blah blah. So we're just going to install our own root CA in all of our machines so that those machines are able to talk to our own private certificate that we don't have to buy and won't expire ever, essentially, till 2039.

Leo: As always, the worst thing in technology, the worst thing in IT is somebody who's half smart. Right? Smart enough to install a self-signed certificate, but not smart enough to understand the implications of what they just did.

Steve: Right. And 60 days after the Superfish debacle that did the same thing. Because that thing was signed in April of this year, and Superfish was in February. So, like, okay. And I think probably the good news is this has been a huge kerfuffle, at least, I mean, Dell is a substantial company with a huge corporate customer base. And corporations must be going crazy.

Leo: All of our finance computers are Dells. All of our editors' machines are Dells. My laptop is a Dell.

Steve: They make nice hardware.

Leo: I've been recommending Dells.

Steve: And it's feature complete, Leo.

Leo: Yeah, no kidding. Right down to the self-signed certificate.

Steve: So the good news is Lenovo, that happened. Now it's happened with Dell. Hopefully other people are listening. Somehow Dell didn't get the message. I don't know how many more times we're going to have to go through this. But of course this is a lesson that we're going to have to learn, obviously more than once.

Anyway, in the show notes, this RCC.exe for Windows machines, which of course are Dell machines, is a neat little quick auditing tool that spots anything. In fact, many people have tweeted that they found that their antimalware had installed its own cert in their machine. And again, it can technically be done in a safer way. If the antimalware is in your machine, and the cert's in your machine, then it's only locally that there's any decryption. I still don't like it. But if you want your secure traffic to be deeply inspected, then this is the way that happens. Even Microsoft's Fiddler tool sticks a cert in the store because it wants to be able to audit your secure communications. So if it's local, and if the certificate is - if they're all different so that it's degenerated fresh, then it's more acceptable. What isn't is that anybody who gets a copy of this cert can essentially get into the communications of Dell machines.

Leo: Don't take this the wrong way, but you trust this Sven guy; right?

Steve: Yeah.

Leo: Okay. Because he's Cuban. The site's hosted in Mexico. It's closed source software. I mean, really? You trust this guy. He's okay. FS1, like he's cool.

Steve: Think so.

Leo: Okay. Just saying. All right. What else?

Steve: Boy, Leo, we've really made you paranoid. Ten years of this podcast...

Leo: I've become paranoid. I don't, you know, when I see ".mx," and then I see he's Cuban, I just, you know, I'm sure he's a nice guy. His name is Sven from Cuba.

Steve: And I do think I'm probably too trusting. I'm too focused on the technology, and I think, oh, isn't that cool. Look, it's a 40K exe, and it audits my root store. Well, you're right, I don't...

Leo: What could possibly be wrong with that? You can, by the way, and I should probably say, you'd be better off to either use the Microsoft console management tool to remove this by hand. Of course, then Dell's going to reinstall it. And then they can run the Dell app after that.

Steve: Yeah, the thing to do is, if you look, if you open the Windows Services app, you will see Dell Foundation Services. Stop it, and then switch it from automatic to disabled.

Leo: Ah.

Steve: That'll shut it down and stop it from autostarting. Then you can delete the certificate, and it'll stay gone. But if you want to, Dell also now has an "oops, we're sorry" tool, and you can simply run that, and it does the same thing.

Leo: Cool.

Steve: So next up, LastPass.

Leo: Oh, yeah. Now, I'm worried about this one.

Steve: So two security guys, Marvin Vigo and Alberto Garcia presented at the recently concluded - I was so tempted to call it Hamsterdam because of course I did love "The Wire."

Leo: That would be the anything's legal zone. Right?

Steve: Yes, yes, exactly, Hamsterdam. Anyway, there was a Black Hat conference where they presented the talk, "Even the LastPass Will Be Stolen, Deal With It." And earlier in the week - the Dell news was recent, so this was most of my Twitter feed for the last 24 hours. Prior to that, and for days, it's been oh, my god, Steve, can't wait to hear what you think about this and so forth on Security Now!.

So these guys did a beautiful job of deeply reverse-engineering the LastPass system, not only the client, but even the server-side API, with an eye toward, I mean, really deeply digging into it. And they did find some things that LastPass could have done better, and immediately fixed, that we'll talk about. But mostly what they found, and there's some really good lessons here, are the things that any local attack would have available.

For example, and we've touched on this a number of times, but the headlines in the popular press got everybody worked up. And the press guys don't really understand this, they're just doing a quick cursory overview of the presentation outline that's been posted so far. So far we don't have the video up of their actual presentation. But, you know, and saying, oh, my god, LastPass has been hacked. These guys go out of their way, they take pains to say it has not been hacked.

But here's the problem. If I'm using LastPass, and I want the convenience of staying logged in while I'm using my computer, so that when I go to a site the login fields get populated for me, and if I've told LastPass to even hit the login button, it does that, but at least the login fields get populated, what does that mean? That means that, without me doing anything, no fingerprint, nothing that I know, nothing that I am, I mean, and not even me, somebody else could walk up to my computer, open a page in my browser, go to a site, and bang, they've just, you know, LastPass logged them in.

So the point is software can do that, too. In order to get the convenience that we want, unfortunately, we've had to empower our computer to be able to log itself into all of the websites we visit. So if malware gets into our machine, or if forensic software - I wouldn't call what these guys did "malware." There was nothing malicious about it. They just wanted to pry it apart to see how it works and see what vulnerabilities they could find. If software is able to inject itself into the browser, and the browser has the ability to access your vault for the purpose of getting passwords and usernames, then that software has access, too.

So that's essentially what these guys did, like sort of phase one. There were three different aspects to what they did, I mean, they deeply went after this. So the first is the so-called "vault decryption attack," where they were able to locally, with full access to the local system, if the user was logged in - and I'll note, if you're not logged in, they couldn't do anything, and they acknowledge that. Or if you had multiple accounts, then by default none of the other ones would be logged in, and they couldn't get into those. So it required that the user be logged in. And then the system was able to log you in, so they were able to get access to the vault. It's not surprising, but it should all give us pause. I mean, that's a consequence of the power that we have given to this password manager. And, by the way, not just LastPass. This applies to them all, every single one of them that operates this way.

Then they found something else, going further, and they call this the "disabled one-time password attack." And this is something, again, we talked about years ago, and it's another tradeoff which LastPass, a.k.a. Joe, carefully made. And that is, what do we do if a user forgets their password? I mean, this is a problem for a practical solution. What do we do? Do we tell them, sorry, you're out of luck? You no longer have any access to any of the Internet. No. And so they decided that was not practical.

So they create a disabled one-time password. You'll remember that LastPass has a one-time password facility, and you can create some one-time passwords and, for example, print them out or carry them in your wallet, if you want to, as sort of "get out of jail free" cards. But there is, by default, and you can turn this off - and by the way, you can turn off "keep me logged in." So these various features, which were there for convenience or password recovery, you can disable them. And our listeners, now understanding this better, may choose to. It's their choice. Less convenience, but more security. So it's the classic tradeoff.

There is, by default, a disabled one-time password. And by going to "recover my password" at LastPass, it jumps you through some hoops, communicates with your browser, ends up reaching into the vault and reenabling the disabled one-time password in order to give you account recovery, so that there is an answer to, "Oh, my god, I've lost my password." And so once again these guys tore that protocol apart and figured out a way, if they're local - and so this gives them - this is a get-around if the user's not logged in. If the user's logged in, it's not that difficult for anything to get access to the vault because the browser can. If the user's not logged in, as I said, it's just you fall back to a brute-force attack, which is impractical. They found some way they felt of maybe reducing it from a 256-bit to a 128-bit attack. But even that, you know, 128 bits in today's world is still an extremely strong level of encryption. We can always add bits and get stronger, but a lot of Internet connections are using 128-bit symmetric key, once the public key is used, in order to negotiate that.

So the disabled one-time password attack says, if you have that feature enabled, which it is by default, to solve the problem of, oh, my god, I forgot my password, and you're local, these guys very cleverly, with a lot of work, figured out how to get in there and reenable that in the same way that LastPass reenabled it remotely with the whole web interface and conversation, and then be able to open up the user's vault in order to get access to it. So those are the local side.

They also did LastPass side. That is, what about a malicious employee? What about the NSA compelling them to do something? What possible evil LastPass power is there? Well, now, that's a tricky thing to answer because we are trusting LastPass. We're running a plugin from LastPass. We're assuming that the plugin does what they promised us and that the plugin encrypts stuff so that they get a blob that they cannot decrypt. We understand that's always been the key concept. But nothing says it always must be.

It turns out, though, they did find something that worried them, which was something called custom_js, custom underscore js, which was some JavaScript which LastPass can inject onto web pages when the LastPass plugin is unable to parse the DOB, the Document Object Model, which is the structure of the page, which it needs to do in order to find the username and password fields. So it's sort of a helper JavaScript for specific sites, for whatever reason, like specific popular sites where for whatever reason the generic parser doesn't work.

And so these guys realized that that was a way that someone evil at the LastPass side, or maybe under order from the NSA, I don't know the law enough to know whether the NSA could compel LastPass to make a change under certain criteria. LastPass can say we can't decrypt it. The NSA maybe, I don't know, can say, well, change things for this user so you can. Anyway, so these guys did note that that, they felt, represented a vulnerability.

And then finally there is, in Firefox, all of the plugin settings are gathered together in a single file called prefs.js, which is just a blob of plaintext, readable, I think it's XML format, file which Firefox maintains, where all of its preferences and all of its plugins preferences are stored. That's the local storage for that kind of stuff. And it includes on Firefox the user's encrypted credentials. So the concern there has been that they're technically sensitive. You'd like them to stay, they are encrypted, and so we're back to the brute-force attack on them; but it's better not to have them wandering around.

The problem is that some users unwittingly post their prefs.js file from Firefox to various Internet forums for other people to look at and say, hey, you know, such and such is broken, can you look at my prefs.js file and tell me what's wrong? And Google will happily, if you put in a phrase from that file or from, it's like browserplugins dot LastPass dot, something like that, you put that into Google, Google turns them up because lots of people over time have posted them. And so, again, they're not decrypted. You'd need a brute-force attack in order to get anything back from it. But it's better not to have those things lost.

So they wrapped up their presentation with recommendations for LastPass users, saying that there are two versions of the plugin. There's the compressed obfuscated JavaScript version, and there's a binary version. They felt that using the binary version of the plugin was better because it was just - it's a binary blob. It's not subject to more ready editing and manipulation. They said do not store the master password, which means don't tell LastPass to remember it, and you have to be providing it all the time. And Leo, I know how you love providing your password all the time.

Leo: They've heard me complain a few times about this.

Steve: Yeah.

Leo: That's when I love that LastPass shows up and says, I'll fill that for you. I feel like, oh, love that.

Steve: Yeah, it's so convenient.

Leo: Yeah.

Steve: And again, as long as your local environment is secure, you're fine. And notice that two years ago, right around this time, when I gave the first unveiling of SQRL, what I talked about was, in this post-Snowden era, who's going to trust a third party? And so SQRL's power is that there is no such thing as the cloud with SQRL. There's no LastPass. There's no third party. There's nobody but you and the site you want to log into. The user does have the problem of, what if I forget my password?

Which is why there are a couple of mitigations. One is that the client prints out a page for you to put somewhere in cold storage. But more easily than that, when you create an identity, you get something called a "rescue code." It's 24 digits, which is - think of it as one and a half credit card numbers because a credit card number is 16 digits. And I make you write it down just once. Write it down just once and put it away. And that is your "get out of jail free" card. It's offline. It's never stored. But the beauty is nobody else has it, and no third party has it, and it doesn't matter what the NSA has to say about it.

You can argue, okay, well, that's kind of a pain. But, yeah, you only have to do it once maybe for your entire life. So that was the design choice I made that sort of prevents all of this other, I mean, all of the problems of, oh, my god, I forgot my password, well, that means that there needs to be some sort of recovery. And LastPass doesn't want the responsibility, so they store that in your machine. And so that creates the vulnerability.

So anyway, in terms of recommendations for LastPass users, activate the new account recovery over SMS. They say, and this I think is kind of impractical, audit your vault for malicious JavaScript payloads. I'm not sure how the common user would do that. Don't use the password reminder. Activate two-factor authentication. And then they also say add country restrictions. And their point there is that would prevent somebody who had exported your stuff from being able to log in as you somewhere else. And that's definitely - that should be top of your list, as long as you're not a frequent international traveler, in which case that would befuddle you. So add country restrictions and disallow Tor logins because that's of course another way for a bad guy to get into your system is just through using Tor. And those are both features that you can disable for heightened security.

And then they also gave LastPass corporate, Joe, some suggestions. They said get rid of the custom_js file. Change the way they're encrypting the vault to do it in one chunk. And they also used ECB encryption. I forgot to mention that earlier. ECB, that's the acronym for Electronic Code Book. And that's the non-cipher mode encryption. Famously, we've all seen that Wikipedia page that talks about encryption where you see the Linux penguin in all of its glory, looking black, white, and yellow, and then it's encrypted under Electronic Code Book, and it turns it into gray fur, but you can still see the penguin there.

The point is it's not statistically sound. It's probably all they needed. But the point was that one of the things that Electronic Code Book is, is it doesn't cause one encryption to rely upon the output of the previous one. The chaining, you know, cipher block chaining, we've talked about CBC and all of the other chaining modes, they create an interblock dependency so that the blocks don't stand alone. And even the first block typically has an initialization vector that influences it. And the IV, initialization vector, is a nonce that changes all the time, and it can be known. It's just good if it's unique because then you can't tell anything about what the first chunk of the encryption is, the first block. But with Electronic Code Book, there's no chaining. You're encrypting each block by itself.

So what it does is it does leak information because under the same key, anytime you encrypt the same thing, you get the same result. So you could see easily, for example, if the user were encrypting the same password on different sites because an identical encrypted version would appear on multiple sites. So they're saying, don't do that. And they also wanted them to strengthen their password-based key derivation function, suggested using certificate pinning rather than just relying on the public key system, actually look at the serial number or the hash of the certificate.

They also suggested that LastPass embrace open source. I'm not sure how they do that, though, as a commercial company, but it's a nice ask. And then, they said with a little smiley face, adopt a retroactive cash reward bug bounty program. Although...

Leo: Pay us. Right? Retroactive means, "And we want some of that."

Steve: Yeah, but they didn't find any bugs.

Leo: Good point.

Steve: There were no bugs found.

Leo: Good point.

Steve: This was a rigorous analysis. And I would argue any other password manager probably suffers this or more problems. One of the beauties of LastPass is that it has been pounded on for quite a while. In their conclusions, I quoted them in the show notes: "Password managers are a great tool that everyone should use. Even though we exposed weaknesses" - they say, I wouldn't say that, but I'm reading what they wrote - "in LastPass, it is still a solid tool," they wrote, "and a better option than using the same password, or only changing the last characters of your password, everywhere. There are ways to harden your LastPass configuration" - which we talked about - "that can avoid some of the explained attacks. Watch the talk and slides for more details on that.

"To finish, we want to point out that the security team at LastPass responded very quickly to all our reports, and a lot of the issues were fixed in just a couple of days. It was very easy to communicate and work with them." So really, you can't ask for more than that.

Leo: Yeah. In fact, everything's going to have vulnerabilities. I say this all the time. Every computer's going to have a problem. It's how they respond.

Steve: Yes.

Leo: How quickly they fix it. Whether they even listen to you. And so many times you hear security researchers say, yeah, we told them about this again and again and again.

Steve: We gave them six months. We never heard. So we're going public because, you know, sorry about that, you had as much time as you need.

Leo: So thank you, Joe, for - his stewardship of LastPass continues to be excellent.

Steve: Well, and the only real issues were caused by convenience.

Leo: Right.

Steve: So unfortunately, power users could say, oh, I'm going to turn off that disabled one-time password feature. Good, turn it off. Make sure you never forget your password because, if you do, they won't be able to help you. But that's the tradeoff. And if the idea of something in your machine getting into your LastPass vault unnerves you, then because your browser has to be able to do that, you have to take that ability away from it, in which case you're having to put in your LastPass password all the time. And then the problem is you're tempted to weaken it to make it easier to put it in. And then you're back to it being subject to a brute-force attack. So there are some things that don't have an answer, but a set of convenience versus security tradeoffs. That's the world, I mean, that's the real world.

Leo: And something we don't do often enough, but I'm sure you do mentally, and I think our audience should do, is kind of rank the threats.

Steve: Yes, yes.

Leo: Nothing's perfect. But if you rank the threats - you say all the time, for instance, if somebody's got physical access to your machine, that's a very different issue.

Steve: Yes.

Leo: And so it's less of a threat if it requires physical access because once somebody has physical access, kind of all bets are off. So rank the threats.

Steve: So, for example, yes, all of the worms that we suffered early, in the early days of this podcast, those were from ports that were open.

Leo: Right.

Steve: Windows had left ports open.

Leo: That's a big threat.

Steve: And, yeah, that's a remote threat. And bang, you know, takeover. It's like, how many millions, tens of millions of machines? Very different to have, finally, with XP Service Pack 2, and they enabled the firewall, I mean, overnight that just ended. And it's like, okay, Microsoft, it sure did take a long time.

Leo: I guess my point is it's better to use a password manager that's less of a threat than using the same password on every site, or an easily guessable password on every site. That's more of a threat than this kind of somewhat less of a threat with the password recovery.

Steve: Yeah. And it's just about understanding. And that's what we're here to do with the podcast. So I got some interesting tweets about Windows 10 and the question of it becoming a bit heavy-handed. What's happened is there have been many reports of Windows 10 silently removing software, which people had previously installed, after this early November update, which apparently has been causing Microsoft lots of headaches of sort of still undisclosed nature. And I heard you talking about it earlier, Leo, so I know you know something about what's going on with this, what is it, is it 1511, I think, is the number of the update. But in this case, for example, some coverage said:

"Microsoft's first big mistake" - I'm sorry, "Microsoft's first big update" - boy, Freudian. "Microsoft's first big update for its operating-system-as-a-service is deleting some user-installed apps without asking Windows owners for permission, according to dozens of complaints on message boards and forums. The affected programs include hardware monitoring tools CPU-Z and Speccy, as well as the AMD Catalyst Control Center for tweaking your Radeon graphics cards. In these instances, the programs apparently no longer functioned properly with the newest version of Windows 10, and Microsoft claims the apps were causing crashes and blue screens of death. While this may help most people who don't want to deal with troubleshooting their system to figure out what is wrong with it, some PC power users are taking issue with Windows 10 removing software without asking."

One poster on Reddit sort of put it nicely. He said: "Microsoft should ask for permission, and not for forgiveness. I would be fine with it if Windows 10 said, 'Hey, this application can cause problems, and we recommend that you uninstall it. Do you want us to do that for you?' and then," as this guy wrote, "shuts its mouth about it if you say no. But they shouldn't just uninstall it without prior warning."

And so my observation from 10,000 feet is that Windows is evolving into a different sort of beast. It's evolving into a connected service, rather than what we have traditionally known as an [crosstalk].

Leo: That's exactly right. Even Microsoft has kind of said that; right?

Steve: Yeah.

Leo: One Windows. They've said this is - that's the kind of implication of things like "This is the last version of Windows."

Steve: Right.

Leo: From now on it's just forever to be updated. It's SAS.

Steve: Yeah. And I think what we're probably going to see is this creating more pressure for diehard Windows users to give alternatives a look. Maybe Linux will inherit a percentage of previous Windows users who just sort of finally say, you know, this isn't what I want. And which is fine. Microsoft doesn't want them either. That's not who Windows 10 is for. Windows 10 is for the majority of people who do, sort of more like the profile of the Chromebook user, who just wants it to work. Doesn't want to have lots of problems. Just please make it work. But they need to use Windows apps, so Microsoft is sort of heading in that direction.

Leo: Yeah. Yeah, remember Microsoft - Windows is still primarily a business operating system.

Steve: Right.

Leo: And I think they're trying to do - it is a communication issue. He's right. They should have asked. Be a simple thing to pop up a dialogue saying, hey, we know that whatever it is, [ZCPOUID] doesn't - causes crashes. We'd like to remove it. I think it's a little easier to explain the AMD Catalyst Control Center because that's essentially the video driver, and so it's an out-of-date video driver. I'm sure for years they've been taking those off.

Steve: So I saw something in digging around that said that Microsoft had deliberately muted its install. And it's possible that this is a mistake. I mean, we're seeing Microsoft making more mistakes with Windows 10. You probably know from talking to Paul that they hugely cut back their testing side, so much so that developers are now having to test their own code, rather than there being a - I mean, and I know as a developer I can't test my own code. It's why the GRC newsgroups are so fabulous for me is it's just not possible.

Leo: Yeah, it's like an author proofreading his own copy, yeah.

Steve: Exactly, yeah.

Leo: And I also think that this is also driven somewhat by mobile, that the expectations people have with their mobile phone is that all of this - this would be normal behavior probably on a mobile phone. And so Microsoft saying, hey, well, this is how we do mobile operating systems. Why shouldn't we do desktops this way?

Steve: Yeah. And so, yeah, I think the nature is changing. I'm going to surprise people with this next one. This is a report of the Manhattan District Attorney's Office on smartphone encryption and public safety. It's a 42-page report, although only half of it; the second half is all kind of addenda and references and things. It's important enough that I created a bit.ly link for it. And what's the first bit.ly link? Because I remembered that I put an "a" on the end because I'd already used the first bit.ly link.

Leo: I don't see any...

Steve: What did I do?

Leo: Maybe you made a mistake the first time.

Steve: What happens if you do...

Leo: Without the "a."

Steve: ...a bit.ly, bit.ly/sn-535?

Leo: Now, kids, don't try this at home because...

Steve: Yeah, something's going to come up.

Leo: It says "Wyoming Oil & Gas Conservation Commission."

Steve: Oh, my god, I forgot. Oh, my lord. Yes, that's down for the Miscellany. So [crosstalk].

Leo: We will do this later today.

Steve: Yeah, don't anybody put that in. Okay. So, but if you add an "a," bit.ly/sn-535a, then you will get a PDF that I'm going to refer to. And I would commend some listeners who are interested to take some time. I'm just going to read through the foreword and one paragraph from the executive summary, to give you a sense for it. But also, I mean, it's interesting.

"Foreword: Most people today live their lives on smartphones; and, in this regard at least, criminals are no different. While in the past criminals have kept evidence of their crimes in file cabinets, closets, and safes, today that evidence is more often found on smartphones. Photos and videos of child sexual assault, text messages between sex traffickers and their customers, even a video of a murder victim being shot to death, these" - I know this is overly dramatic, but hold on - "these are just a few of the pieces of evidence found on smartphones and used to prosecute people committing horrific crimes."

And this is a little pointed, but still: "Last fall, a decision by a single company" - we know where they're aiming - "changed the way those of us in law enforcement work to keep the public safe and bring justice to victims and their families. In September 2014, Apple Inc. announced that its new operating system for smartphones and tablets would employ, by default, what is commonly referred to as 'full-disk encryption,' making data on its devices completely inaccessible without a passcode. Shortly thereafter, Google Inc. announced it would do the same. Apple's and Google's decisions to enable full-disk encryption by default on smartphones means that law enforcement officials can no longer access evidence of crimes stored on smartphones, even though the officials have a search warrant issued by a neutral judge.

"Apple and Google are not responsible for keeping the public safe. That's the job of law enforcement. But the consequences of these companies' actions on the public safety are severe. That is why my office has been working with our law enforcement partners around the world to craft the solution recommended in this report. We believe there is a responsible way to balance safety and security."

And then I'm skipping down to Part V in the Executive Summary, which follows the Foreword, where they write: "Part V?" - and this is the summary of Part V - "sets forth a proposed solution: Congress should enact a statute that requires" - and listen to the wording carefully - "any designer of an operating system for a smartphone or tablet manufactured, leased, or sold in the U.S. to ensure that data on its devices is accessible pursuant to a search warrant. Such a law would be well within Congress's Commerce Clause powers and does not require costly or difficult technological innovations." And I agree.

Leo: What?

Steve: Yes. I think that's...

Leo: Really.

Steve: ...the right solution, yes. I think that's where we're probably going to go, and I think that's where we're going to end. This does not mean a backdoor. This means essentially that, under court order, Apple can be compelled to provide something or to unlock a phone that they are given, much as they have been able to prior to iOS8.

Leo: It's a backdoor, but Apple has the keys. Only Apple has the keys.

Steve: Well, and, yes, and that's the point. That's why this proposal makes sense. And they're not asking for data in flight. They're not asking for on-the-fly decryption. They're just saying, it is the case that phones are massively evidentiary. And we live in a country, in the U.S. I'm talking about, of course, where if law enforcement gets a judge to agree, your home can be broken into. You may not like it, but your home can be searched. Your car can be searched. I mean, that's the balance that has traditionally been struck and that we're all living with rather happily now.

The question has been how does technology confound this? And the point here is this is not a key that the government has. This is not any sort of a master key. This would not be an algorithm where, if it got loose, suddenly all Apple iPhones and iOS devices would then be subject to break-in. Apple could do it right. And what "doing it right" means is Apple has an ongoing relationship with all the devices that are tethered to them. It would mean a high-quality random key stored in that secure element which would be unique and unchangeable in every phone, which Apple would secure at their end, and which under subpoena from law enforcement would enable the phone to be unlocked. And to me, the remaining issue is that does this create a slippery slope. Is there, like, oh, well, if we can have that, what more can we get?

Leo: Well, you know, I was thinking that because really a lot of criminals meet in private in homes and apartments, exchange child pornography, plan terrorist attacks. I think apartment buildings should be required to put a microphone in every apartment, and then that way with federal law enforcement subpoena or warrant, could turn those microphones on. I think that's another good idea. The problem is, I think law enforcement - look. I understand. We would like to catch criminals and bad guys. But I think law enforcement overreaches a little bit. And this sounds like an overreach. I can't agree with you, Steve. This is an overreach.

Steve: Okay.

Leo: I mean, really?

Steve: Yeah.

Leo: Okay. I mean, you're right, this is not saying - it's not a key escrow. It's not, you know, it is with warrant. It's no different than it was a few years back.

Steve: Correct.

Leo: I didn't encrypt my Google phones usually, but now of course all my Google phones since Android 5.0 and all my iPhones are encrypted. I didn't do it. They just do it automatically for me. It's an interesting question.

Steve: Yeah. I think it's going to happen. And if it does, I could live with it. I think that's my point is I don't think there's an argument, a sound argument on - and I should say, Leo, I also get your point. I mean, completely. And so there will no doubt be an interesting debate. But there are huge technological arguments that we have discussed that make this probably impossible for data in transit. And that's not what's being asked for here. This is...

Leo: Yeah. You know, one of the reasons I think Google started doing this and Apple started doing this is not because of the U.S. government, but because of countries like China, which by the way could say exactly this paragraph, except replace the words "manufactured, leased, or sold in the U.S." with "manufactured, leased, or sold in China."

Steve: Yeah.

Leo: And what Google doesn't want to do, remember Yahoo! was forced to turn over emails from a Chinese dissident who was then prosecuted and executed.

Steve: Yeah.

Leo: I think Google and Apple both saw the writing on the wall, and it wasn't the U.S. government they were worried about.

Steve: They don't want the responsibility.

Leo: They don't want to be put in the position where a government of any kind, U.S. or Chinese, could come to them and say, "You've got to reveal this information." So that's why they really don't want this. And as soon as this happens, of course, it's not just going to be the U.S. government. It's going to be every other government in the world.

Steve: Yeah.

Leo: And so I don't think Apple did this particularly to protect our privacy. I think they wanted to get out of the fray.

Steve: Well, and you can imagine, you know, we covered the stories of the three-month backlog that they had in the iOS7 era where essentially they were having to brute-force their own customers' phones because brute-forcing was possible. And so law enforcement would get a subpoena...

Leo: They hated it, yeah.

Steve: ...would hand them the phone and say, you know, we need this as soon as possible. And Apple was like, ah, well, we've got about a 90-day backlog at the moment, which upset law enforcement. And so you can imagine Apple just saying, you know, let's just fix this. And let's sell privacy. And we know that Tim's been selling privacy. It's been a benefit. But...

Leo: Is it unreasonable to say to law enforcement, well, you also had other means than looking at somebody's smartphone of finding and prosecuting criminals. You didn't always, you know, and it wasn't necessarily because you could see into their data. Is it not unreasonable to say there are plenty of other police methods, I mean, torture works, too, but - or maybe it doesn't. There's some debate over that. But assuming that torture works doesn't mean you should use it.

Steve: Yeah. I mean, I'm with you completely. But I also am watching what's going on. I'm sure you've heard McCain announcing that there will be hearings. And it's like, oh, okay, here we go.

Leo: No, I think you might be right. I think it's going to happen.

Steve: I don't know how this issue gets resolved.

Leo: Yeah. Well, I know how I would like it to get resolved, which is stay out of my stuff. And if that, you know, I think there's very clear evidence that law enforcement would love to implement mass surveillance. Oh, wait a minute, they already did. So let's not make this any easier for them than it is.

Steve: Yup.

Leo: But I understand, on the other hand, being a good liberal, I'm listening to your point of view, and I understand what you're saying. I think you're absolutely right. It's a very tough question.

Steve: I guess to me, and I'm not a legal scholar nor a constitutional scholar. But this feels proportional. It feels like this is like what we have with existing subpoenas and court orders, where a neutral judge weighs the evidence and says, okay, yeah, there's reasonable reason to believe and that law, I mean, we want law enforcement to be effective. We'd like bad guys to get caught. And frankly, I mean, and the argument that I don't have anything on my phone, well, we know that's not the argument. It's should the phone have content which is protected, absolutely inaccessible.

Leo: Well, to make a point in your favor, with a warrant, law enforcement could come into your home, search your file cabinets, search all your possessions. As long as there's a warrant, they have a right to do that. And I think you probably - I would submit that there's much more private stuff in your home than there is on your smartphone.

Steve: Yeah.

Leo: So why should the smartphone have some sort of special protection that your home does not? So that's the argument on the other side.

Steve: So this document also had a couple of little tasty tidbits that I knew that our listeners would find interesting, I mean, just as fact. Under "The Difficulty of Getting Passcodes from Defendants," there was some interesting new information, to me. They wrote: "Case law holds almost universally that a defendant cannot be compelled by, e.g., a grand jury subpoena or order of the court to provide the government with her or his passcode" - of course we've talked about this often - "because such compulsion would violate the defendant's Fifth Amendment right against self-incrimination. There are two potential exceptions to this rule," which is what I found interesting and wanted to share.

"First, it is an open question whether, instead of being compelled to provide the government with a passcode, the defendant might be compelled to unlock her or his phone using the passcode. There have been no cases considering this precise question; and, although a court might conclude that it is no different from the situation in which a defendant is compelled to provide the government with the passcode, it might also determine that the situations are somewhat different," meaning don't tell us your secret, but here's your phone. You must use that secret, which you're allowed to keep secret, in order to give us access, in order to unlock the phone. So the point was that's an open and different question.

And then, secondly, "If the existence of evidence on the phone is a foregone conclusion, then the defendant may have no Fifth Amendment privilege with respect to the contents of the phone, and thus may be compelled to provide the government with the passcode. It would be difficult in most circumstances, however, for the government to establish with the requisite degree of certainty the existence of evidence in a phone that would clear the 'foregone conclusion' hurdle." But anyway, from a legal standpoint, this is the kind of stuff that Denise knows cold. But what this says is Fifth Amendment privilege only extends where there's suspicion but essentially not proof, but the idea being they would be able to demonstrate what is in the phone, and then that would not allow the person to claim Fifth Amendment privilege.

Leo: I don't think that's anything new. I think that's consistent.

Steve: Right, right.

Leo: Yeah, that's consistent with prior law.

Steve: And then they just finish: "In any event, even if the government could lawfully compel a defendant to disclose her or his passcode" - meaning even if those other things applied - "or to open her or his phone using the passcode, there is substantial likelihood that any defendant who faces potentially serious criminal charges would simply refuse to comply with the subpoena or order, and go into contempt." So anyway...

Leo: So if let's say Google and Apple decide, you know, lawmakers pass a law, and Google and Apple comply and turn off the full-disk encryption, or make it optional, is there...

Steve: Oh, no, no, not that. Neither of those. It would be...

Leo: A backdoor.

Steve: Well...

Leo: Well, their backdoor.

Steve: That's a loaded term. Have the ability...

Leo: To decrypt; right.

Steve: ...under court order, on a phone-by-phone basis, that they are able to unlock it. That's what this asks for.

Leo: But, okay. So TrueCrypt does full-disk encryption.

Steve: Yup.

Leo: Maybe because of the nature of the mobile platform, you can't go out and get a third-party full-disk encryption utility on your smartphone.

Steve: Right, right. Now, for example, somebody...

Leo: But what about - wouldn't government want to ban TrueCrypt at the same time? Otherwise the same problems.

Steve: Somebody using Threema, for example. Threema has in your phone a log of your dialogue. Yet it is stored encrypted.

Leo: Right.

Steve: So if you were using Threema, then Apple unlocking your phone would give nobody visibility into Threema because its security is absolute. So there.

Leo: Well, that kind of is my point, which is...

Steve: Right.

Leo: Who's this going to catch?

Steve: Right, yeah. This is going to - this turns back the clock a couple years to the way we were before, where...

Leo: But the good news is terrorists can still use encrypted communications, so we're okay.

Steve: Wait, now, say that again? The good news is...

Leo: Terrorists and pederasts...

Steve: ...terrorists can still use...

Leo: ...can still use encryption. So we're okay.

Steve: No. The good news is Mom and I can discuss...

Leo: Yeah, you don't get any privacy.

Steve: ...cooking for Thanksgiving, and nobody can see it.

Leo: No, no. You get no privacy. It's the bad guys who will have the incentive...

Steve: I've got Mom up on Threema.

Leo: You've got Mom using Threema? Wow.

Steve: So the NSA does not know what Mom's cranberry sauce recipe is because that is secret.

Leo: Definitely there's now a flag on you and Mom in the NSA headquarters, I might add. And by the way, what's next? You've got to ban TrueCrypt; right? You've got to ban...

Steve: Well, you know, this is why I backed away from CryptoLink, my own nascent VPN, was that I was worried that what's coming is no one can offer encryption without providing a backdoor. So, but my point is...

Leo: Yeah. See, that's the slippery slope is that once you accept...

Steve: That's the slippery slope part.

Leo: Once you accept the premise that government needs to be able to search anything, and encryption confounds that, then, oh, well, the nice thing is it's easy to do on a smartphone. You just tell Apple and Google, keep a key, would you? But then you've accepted the premise.

Steve: Yes.

Leo: So now, well, if nobody has the right to encryption that's invulnerable to a search warrant, then we've got to ban all these other things. Or maybe get them to put a backdoor on it.

Steve: Like I said, Leo, I just love...

Leo: Bye-bye, BitLocker.

Steve: I love this era that we're in. It is so full of interesting balancing questions. And, I mean, really good ones because, again, we would like law enforcement to have the information it needs, but not at the cost of sacrificing global privacy.

Leo: Right.

Steve: And so my point is this particular niche doesn't sacrifice global privacy. This comports with the way the U.S. Constitution provides search in the instance of reasonable suspicion. And so to me, I don't know how you resist this. But I certainly acknowledge the slippery slope problem [crosstalk].

Leo: Yeah, because you really haven't solved anything just by the cell phone thing. You've got to really go farther if you want to stop these child molesters.

Steve: Yeah.

Leo: Yeah, it's just so - it's so difficult.

Steve: Okay. It is. Interesting. So I just wanted to mention that a bunch of my followers have been tweeting that they're in receipt of Let's Encrypt beta invitations. So as 2015 nears the end, Let's Encrypt is kind of incrementally and carefully rolling its service out, making it available to sites. I think this is exactly the way you do something like this is you get some experience, you fold that back, and you deal with any problems before you turn the whole world loose on it because I think it's going to be quite popular.

So miscellaneous goodies. I did want to mention that, while it seemed like last week our discussion of the whole crypto controversy, where we've been spending a lot of time, might have been a rehash. Many people got a lot from it. I think in some ways we took it further than we had, used some different analogies than we had. And it ended up being really useful. And one favorite tweet that several people sent me, I don't know that Ben Hughes was its originator, but it's such a perfect gotcha that I had to share it with everybody. So he tweeted: "If banning encryption would stop terrorists from using it, why don't they just make terrorism illegal and be done with it?"

Leo: There you go.

Steve: Which I thought, ooh, boy, that's perfect.

Leo: That's the variation of, if they outlaw guns, only outlaws will have guns.

Steve: Yeah, precisely, yeah. Exactly. Also, look at that screenshot on the next page, Leo.

Leo: Okay.

Steve: I wanted to let people know, I just wanted to close...

Leo: Showoff.

Steve: ...close the loop here. I've been, and our listeners and viewers know, we've had zero problems. We're back to, like, fabulous connections. The initial switchover from my pair of T1s was causing some trouble due to my LAN issues, as I was recabling and changing switches and things. Then a friend of mine in Atlanta with Cox sent me a beautiful Netgear CM600, which is the now available, recently certified by Cox - he was just waiting for Cox to certify it - 24 downstream channels and 8 upstream channels. So I am now on multiple different bandwidth meters getting 341Mb down and 33Mb up. For, let's see, that's - when I had the two T1s, they were bonded, so I was getting 3.54Mb. In other words...

Leo: Symmetric.

Steve: ...one one-hundredth of what I'm getting now.

Leo: Probably for a lot more money, even; right?

Steve: Yes. And I was paying something like 460 a month. This is something like $70. And, I mean, I'm not even sure of that. It might be less.

Leo: Now, which speed test did you use for this?

Steve: This one is Speedtest.net.

Leo: Yeah. I'd be careful because...

Steve: I know. I used DSL and a couple others. And they were all comparable.

Leo: There's some evidence that ISPs note which one you're using and tune their performance suitably.

Steve: Well, yes. It's absolutely the case that they know what IP I'm going to, and they could unthrottle my connection [crosstalk].

Leo: Turn him up real quick.

Steve: Yup. However, I have to say, last night when I downloaded the first episode of the new Syfy "Expanse" series that we'll be talking about in a minute, it came down very nicely. Oh, and I finally found a good use for my iPad Pro. But anyway, we'll get to that in a second.

Leo: Oh, you're not giving it back, huh?

Steve: Unh-unh. Actually, the darn Pencil hasn't shipped yet, and so I can't...

Leo: You've got to wait for that. You've got to wait for that.

Steve: Yeah. I have to have that experience. But I also wanted a quick Auralux follow-up. It's very clear from many of the tweets - agonizing, in some cases, tweets - that overall productivity among a significant subset of our listeners has...

Leo: Damn you, Gibson.

Steve: ...significantly collapsed this past week as a consequence of Auralux. And I'm just - everybody is thankful, not only for Thanksgiving, but they're thankful that there's a four-day weekend coming up, so they may be able to make some progress on Auralux. So it's been a huge...

Leo: I made it my pick for the app cap for iOS Today. I just - it's so much fun, and it's so challenging. This is hard.

Steve: Now, remember when you were saying, oh, it's so slow? And I was thinking, oh, Leo, just wait. I'm like, I'm going bing bing bing bing bing bing bing. I'm like, you know, marshaling forces and sending them and, yeah, I mean, it really is. And there are some, I mean, there's one I'm still scratching my head on. So, yeah.

Leo: Yeah, yeah. This is normal. They call this one a "normal" one. There's some of them are really hard when you get into the hard ones. They're like, hey, they get to, wait a minute, they get to start with four planets, and you got one? What the what?

Steve: And the other thing, you notice where like they set up sort of a streaming effect where the last guy sends it to the next guy, who sends it to the next guy, and it builds up this current. I mean, anyway, it's just wonderfully...

Leo: It's fun, yeah. It's hard, though.

Steve: Yeah. And so for anyone who missed last week's podcast, available on iOS platform and on Android, Auralux, A-U-R-A-L-U-X. It's a goodie.

Leo: I'm trying to get - if I get this center star to build up, I might have a shot. I want to get the orange and the green.

Steve: Yeah, but unfortunately, you've got to get your little blue guys in, in order to...

Leo: I know. Look it, I just lost this planet, I know. And now they're going to invade me.

Steve: Oh, I hate that.

Leo: I want to play against real humans, though. They need to add an online version of this because the AI, like the AI should, when one of the - when, like, Orange gets the center planet, Green should try to get him out of there. Instead, he's still attacking me. It's like, come on, pay attention. Don't bother with me.

Steve: Yeah. Going for each other.

Leo: Oh, I just lost the whole game. There it goes. It's over.

Steve: I know. It really is frustrating.

Leo: It's over, man, it's over.

Steve: Okay. So yesterday Syfy channel, and I spelled it wrong here, S-Y-F-Y, unfortunately, released a tease for the December, I think it's December 10th the series officially starts. And this is the "Expanse" series. We knew it was coming, oh, about nine months ago, I guess, because I read the whole series because the books are always better. And so I read them in advance of seeing the movie, just so that I would know. It's available on Apple TV, iTunes, the Google Play Store, and Comcast on Demand, and probably other places. I looked for it on Fire TV, but I didn't see it there.

So this is the first episode, 44 minutes, blessedly commercial free. And you can see where the commercial breaks are. And I don't know. The very first opening vignette, and you know I'm not going to give any spoilers, was extremely awkward and unclear. Whereas, of course, I know exactly what happened because I read the book. But nobody watching the first beginning of the first episode would have any idea what that was. So I'm worried that the book, which was rich with narrative, may not translate very well. But we do have a new sci-fi series starting officially in early December. And you can see the first episode now. So I wanted to make sure that our sci-fi avid listeners knew. And I know that we've got a bunch of them. Okay. Back to that bit.ly/sn-535.

Leo: I kept it around just in case you were going back to that one.

Steve: This week's difficult-to-beat "You're Doing It Wrong." Everybody, bit.ly/sn-535, all lowercase. This is from the Wyoming Oil & Gas Conservation Commission. And in little tiny fine print in the bottom it says "This page requires a pass word" - and those are two separate words - "which allows the user to locate wells for filing of Sundries Form 4. Please call if you have any questions or problems."

Now, unfortunately, the page which has this fill-in for your password explains that you should be very careful not to use any special characters, and it gives them all to you, because the SQL server which they have interpreting what you send will pick that up and treat them as SQL commands. So it's like, okay, you know, wow.

Leo: My password is Little Johnny Drop Tables.

Steve: Little old Johnny Drop Tables, yup. Now...

Leo: They even tell you ahead of time, hey, just so you know, try not to use any SQL commands in your password. Just, you know...

Steve: They're using a GET query rather than a POST. I saw somebody who actually did put something in there, say that what they put in was in the URL. So once again, you know - and someone commented, it's like IT set it up for the office secretary to use.

Leo: Yeah. Just make the monkey123 and be done with it, come on.

Steve: Wow. Yeah.

Leo: It's written in ColdFusion, if that tells you anything.

Steve: Wow.

Leo: Wow.

Steve: Now, I tweet the show note link every week, and I get people asking me for the show note link. I also tweet things that I think will be interesting, like last night I tweeted that the first episode of "Expanse," of the "Expanse" series was available. And of course I don't tweet a lot, I'm not a high-volume tweeter. Many people are still not on the Twitter. I get it. I understand. But I wanted to make sure that people knew that it would be possible to, for example, follow me or any other people, low-volume tweeters, with SMS. That still works and exists. And the other problem is people who are on Twitter sometimes follow 3,000 people, so my very occasional tweets are just going to be lost in the noise. You'll never see them. They'll scroll right off into oblivion. So in the U.S., all the carriers use 40404.

And so, for example, if you were to tweet the string "follow sggrc" to the recipient 40404, that's telling Twitter that, when I tweet something, which I rarely do, but that you don't want to miss, you'll receive it as a text message. So I just wanted to make sure people knew there was, for whatever reason, people not using Twitter who are sending me email, actually, I don't see tweets from them because they're not on Twitter, but email saying, hey, where's the link for last week's show notes? Well, you know, I always tweet it. So people can follow me, if they care, just using SMS, which is something some people forget. You don't have to even have a Twitter subscription.

Now, errata. It turns out that I completely - and when the moment - I'm interrupting myself. But the moment I saw this it's like, oh, my lord. It's like seeing a typo where you say your and you meant Y-O-U-'-R-E, but you write your, Y-O-U-R.

Leo: Oh, I hate doing that, yeah. Because people might think you didn't know better.

Steve: You know better. You know the difference. But that's the way it came out. So anyway, I appreciated finding this in the mailbag from someone who I guess his name is Haykan, although he's got some unicode craziness in his name. He is in Sweden. He says: "Hi. Regarding wildcard certificates, I think you mixed things up a bit." Oh, boy, did I. "Wildcard certificates do not require SNI. SNI only matters if the server wants to use multiple certificates on the same IP. There's no problem with just multiple names supported by the same certificate on a single IP. You can use a single wildcard certificate, supporting all the subdomains you like, with no need for SNI."

Now, to wind the clock back two weeks, that was me mis-answering a question from someone whose hosting provider was saying, oh, you've been a loyal customer, thank you for renewing. We're going to give you secure access to your website. And they were doing it, I was sure, by having a cert that was *.alshostingservice.com. And then I got all mixed up, saying that that required Server Name Identification, which is SNI. It doesn't.

So thank you very much for the correction, and I wanted to make sure I corrected the record because, for example, GRC has three different domains which are covered, but I use an EV certificate that does not allow wildcards. And if I had a "*." certificate, then that doesn't need SNI. So I explained the SNI thing correctly last week about the need to select the certificate on the fly, except I mashed it together with the wildcard cert, which does not need server name identification because it's the same cert is being matched with the pattern of all, thus the asterisk. And finally...

Leo: [Crosstalk] the short code thing, I don't know if this would happen to everybody. But turning on your short codes on Twitter now means that everybody else, I'm getting everybody else's texts, as well. So...

Steve: Turning on...

Leo: So as soon as I said I should follow SGgrc, I'm now getting Don Lemon from CNN and Jason - in other words, my text messages are filling up. So I think what you're - you might be turning on in general.

Steve: Oh. Is the setting in your Twitter account for "enable mobile."

Leo: Yeah. And if you have a mobile number registered with them, it'll say, oh, I know this number. Let me send you everything.

Steve: Ah. Got it.

Leo: And that's not what you want. So you might have to go into your settings and modify [crosstalk].

Steve: Sorry about that.

Leo: You know, that's - I just wanted to clarify that because some people are going to have that happen.

Steve: Yeah, good, good. Keegan R. Griffiths in Australia, I found this in the mailbag, on the 20th of November said: "Hello, Steve. I've been watching/listening to SN for about six months now and absolutely love it. I have a question about SpinRite. In recent podcasts, some SpinRite stories have mentioned running it on SSDs, and I'd like to ask if and how SpinRite can repair issues on SSDs as it was developed for mechanical drives that are quite different to their solid-state counterparts. Keep up the great work. I hope Leo and yourself continue this wonderful service for many years to come."

So I have covered this in the past. But Keegan, and anyone who's joined more recently, I'll just explain that it is true that SpinRite, obviously, was developed in the pre-SSD era. But what has come to light in recent increased SSD use and density and popularity and reduction in prices that's made them affordable, is the solid-state drive manufacturers have unfortunately played the same game that the hard drive manufacturers have played. And that is, they've allowed the pressure to increase densities and lower costs to push their technology, different as it is, into sort of a gray zone where they're relying on math in order to, where necessary, resolve ambiguity in the data that's recorded.

And the point is that density is so high that they can't always get back the ones and zero bits exactly the way they wrote them. They can get them mostly back the way they wrote them, and then they rely on error correction to correct those that were a little too fuzzy and that read back as the wrong bit, or the bit set the wrong way. So much as we would like to believe that SSDs are like RAM, like they're solid-state, and they're not prone, unfortunately that's not the case.

And in fact we just saw some coverage of them adding more bits to the cell. For example, the SSDs I managed to purchase were single-level cell SSDs a few years ago. I don't even think you can get those anymore because the idea being the single-bit cell would only store a one or a zero, that is, a voltage, a charge of either fully charged or fully discharged. But that meant that cell only had two states, one or zero.

Then someone said, hey. If we store either no charge, a third of a charge, two thirds of a charge, or a full charge, that's four states, or two bits. And so we can double the density of the SSD, like for free. Unfortunately, of course, it wasn't quite for free because now your charge discriminator has to be much more accurate, and you're going to tend to be more in the gray zone. And but what we again recently heard was they'd gone to three bits, meaning eight different levels of charge. And again, more reliance on correction.

So it turns out that, by happy coincidence, all of the technology that I developed for doing really deep hard drive recovery exactly maps onto the same technology - actually, almost. Because I do some seeks back and forth in order to get the heads repositioned in a different location every time I do a read. That's obviously not going to help us with an SSD, and that'll be one of the things that 6.1 eliminates when it sees that it's on an SSD because it'll be explicitly SSD aware. So SpinRite will get better with SSDs, but already is able to recover because it turns out there's enough similarity in the way they both store data that SpinRite is able to recover from either.

Leo: Nice. It's a happy accident. I have your PDF with all the questions. Let me open it, and we will get underway. Questions. Starting with Justin in, now, I never know how to pronounce it [Oh-lay-tha]. Is it Olathe, Kansas? Is that correct?

Steve: You know, I was glad you were going to have to because I thought, you know, I messed up Guinness or Guineas or whatever it was I did last week.

Leo: Guineas [crosstalk], Guineas Book of World Records, yeah.

Steve: Yeah.

Leo: I think it's Olathe. Chatroom? It's Olathe; right? Anyway, he says: I received an automated email - this is interesting - from - let me close out the slideshow mode. Thank you, I don't want to see a slide show. I received automated email from Amazon saying, as part of routine monitoring which they perform on email addresses and passwords included in lists of non-Amazon related breaches and hacks - wow - they found the email associated with my Amazon account and preemptively reset my password. Whoa.

This is the first time I've ever encountered a site that proactively scanned the lists. Maybe they were using Troy Hunt's HaveIBeenPwned.com, he posits, or a similar list Amazon maintains. Are you aware, is this a common practice among companies with large web presences like the Amazons of the world? I can't wait for SQRL to be widely available and hopefully provide a robust solution to these continual password woes. I might add to that, by the way, Amazon turned on two-factor authentication just recently.

Steve: Yes, or made available two-factor authentication.

Leo: Right. They didn't turn it on. You have to turn it on. But it's now available, which is good news.

Steve: So Justin's note was nice enough to enclose the email from Amazon, which says: "This is an important message from Amazon.com. At Amazon, we take your security and privacy very seriously. As part of routine monitoring, we discovered a list of email address and password sets posted online. While the list was not Amazon-related, we know that many customers reuse their passwords on several websites. We believe your email address and password set was on that list. So we have taken the precaution of resetting your Amazon.com password. We apologize for any inconvenience this has caused, but felt that it was necessary to help protect you and your Amazon account. To regain access to your Amazon customer account: 1. Go to Amazon.com." And I love that they didn't say "click this link."

Leo: Good.

Steve: They said "Go to Amazon.com and click the 'Your Account' link at the top of our website. Click the link that says 'Forgot your password?' Follow the instructions to set a new password for your account. Please choose a new password, and do not use the same password you used with us previously."

Leo: Reasonable, yeah.

Steve: Otherwise you're going to get another email from us tomorrow.

Leo: I hope so, yeah.

Steve: "We also highly recommend that you choose a password that you are not using on any other sites. We look forward to seeing you again soon." Hopefully the person doesn't disappear forever now. "Sincerely, Amazon.com." So, first of all, very impressive. I mean, this being proactive like this is...

Leo: Yeah.

Steve: You know, I echo your reaction upon learning of this, Leo.

Leo: Yeah, good job, Amazon.

Steve: It's like, wow. And very nice that they didn't say "click this link." The danger is that, if this practice became widespread, then it's a perfect way for phishers, P-H-I-S-H-E-R, phishers, to send phishing mail where they would make a slight change and say "Click this link to go to Amazon.com," and then of course send you to Amazone.com or some slight variation and hope that you don't notice the difference and then tell you to enter, log on with your old credentials so you can change it to your new ones, and of course that would allow them access. So anyway, very, very impressive. And I just loved how proactive they were.

Leo: This comes to us from Marco Silva in Funchal, Madeira Islands, Portugal. I'd like to go there soon. He needs some browser configuration help. He says: Hi, Steve and Leo. In Episode 531 you re-mentioned the issue on the weak Diffie-Hellman export-grade crypto because the guys who found this issue also noticed that most servers are only using the same common 1024-bit prime.

In your show notes, you have a link to an EFF article called "How to Protect Yourself from NSA Attacks on 1024-bit DH." In that article, they present some practical tips to protect yourselves. However, in the web browser section, they also show us how to remove the Diffie-Hellman crypto from various browsers. But they say, "It's important to note there is a trade-off here: Removing your client's support for DHE ciphers will eliminate the risk of this attack, but it may also remove Forward Secrecy support altogether for some sites." Ooh.

So what I would like to ask you is what should we do? Remove the DHE cipher, or leave it there in order to maintain Forward Secrecy support? I've been listening almost since the beginning and love the show. I hope Leo is able to come - oh, thank you, I will, I'll be right there - come to the Madeira Islands some time in the future. It's a very beautiful place, and I'm sure he'd love it here. He will also like our Madeira wine. I do love Madeira. It was used to celebrate the independence of the United States. I think Steve will like it here, too. Madeira is...

Steve: And I'm never going to go.

Leo: What? Why not?

Steve: Well, because I like it right here.

Leo: Are you not a traveler? You don't like to travel?

Steve: I've got great wine. No, I'm not a traveler. In fact, we've been trying to see whether there's any way to get me up for The New Screen Savers, and...

Leo: You don't even want to come to Petaluma, let alone Madeira.

Steve: Well, the problem is really, as they say, there's no way to get there from here. The problem is on the weekend I keep hearing every one of your guests talking about the traffic.

Leo: Oh, yeah, traffic's terrible, yeah.

Steve: But the good news is, in talking to, oh, shoot, Tanya, she believed that you guys could get me into the local airport.

Leo: Yeah, easily.

Steve: It turns out, well, that's only from LAX. But at the end of March the Orange County airport will start a direct flight to the airport that you have up there. And then you're just not going to be able to get rid of me, Leo. I'm going to just be underfoot.

Leo: That would be a good way to come, actually, yeah. We take those planes. There are not very many a day, so you have to arrange your schedule a little bit.

Steve: I only need one.

Leo: Yeah, one's enough.

Steve: Yeah.

Leo: And we'll send a helicopter down for you some time.

Steve: So, okay. So Marco, what he's saying is that, as has been in the press, as we've talked a couple weeks ago, there were - remember we discussed this originally months ago. Then it kind of came back around because the original problem was the weakening of Diffie-Hellman, what was revealed was that, due to the past export limits, which shortened the maximum allowable number of bits on public key crypto to 512, there were still servers that supported 512 Diffie-Hellman.

And so what was done was that, quickly, servers were told, just stop doing that. Turn that off. And you can go to SSL Labs, GRC, and there are test pages that will show you GRC. I turned those off a long time ago. But then what came back was that, while the 1024-bit version doesn't have the brute-force ability problem, it has a different problem, which is it's a pre-computation problem where, because everyone used - because technically you can reuse the same, one of the same primes in Diffie-Hellman. It's still not a good idea. But it turns out most of the Internet uses the same one. So the idea then is disable this Diffie-Hellman cipher.

Well, now, this is confusing, and this is what kind of caught Marco because there are two flavors of Diffie-Hellman. There's DHE, which is Diffie-Hellman Ephemeral, and it's the ephemeralness that we want. That's the Forward Secrecy part, where if in the future the server's private key became known, it would not be possible to go back and decrypt everything that that server had exchanged during the lifetime of that key. Which is the way it is now, if you do not use an ephemeral cipher. So the trick is that the elliptic curve ciphers are secure, but they're not as widely supported. They've come along more recently.

So this note that was made about if you disable DHE ciphers in your browser, remember that the browser, the client and the server negotiate the cipher suites that they have in common. And these are cipher suites that we're discussing. If you disabled the DHE that is almost certainly available, then your browser and the remote server might end up not negotiating the elliptic curve Diffie-Hellman ephemeral cipher, which is increasingly available, but you have less guarantee. And so then they fall back to a non-Diffie-Hellman key agreement protocol, which wasn't offering the perfect forward secrecy that the ephemeral Diffie-Hellman does.

So, he says, what should I do? I would say do nothing. This is going to be solved for us by the browsers. There's already conversation in the industry about retiring the DH, the non-elliptic curve Diffie-Hellman, just removing it from the browsers. What will happen is the metrics will be put in place. Notice will be given, very much like the SHA-1 issue. Servers will be told, you really want to start supporting elliptic curve. We'll make sure that it's available, so there's no reason why servers won't be able to support it. And then we'll just say, you know, Diffie-Hellman non-elliptic curve, your day is past, thanks very much. So I think that's what's going to happen. I don't think users really ought to mess with this. Browsers will take care of it. And we know how preemptive browsers are being about the security of their users.

Leo: Thank goodness, yeah.

Steve: So I think you need to do nothing. And that wraps our podcast. We will pick this up next week.

Leo: I see the clock on the wall says 4:00 p.m. Pacific time. Which means that the witching hour has arrived.

Steve: And we've given our listeners an hour and 57 minutes, which I'm told is several commutes' worth for most of them. So that ought to tide them over till next week.

Leo: I don't know, this family's got to go back to Sonora. Can we do three or four more hours? It'd just really be helpful for them.

Steve: I'll just go make some more coffee.

Leo: Steve's at GRC.com, the Gibson Research Corporation. That's where you'll find all the stuff he does. Of course SpinRite, the world's best hard drive maintenance and recovery utility, but also the Perfect Paper Passwords, SQRL, Password Haystacks, all the free stuff he does, ShieldsUP!. People still use that, right? It's still really a great way...

Steve: No, it's cranking away. We're approaching 100 million users.

Leo: That is amazing. That is amazing. He also puts the podcast there, so you can get audio, 16Kb, if you want, and 64Kb, as well, and full transcriptions. Elaine Farris writes those nice transcriptions for us. That's all at GRC.com. We also have the show, audio and video, as well, at TWiT.tv/sn. It's also on YouTube, YouTube.com/securitynow. It's also on every podcast application in the world. And even now, on Apple TV, there are four Apple TV apps for TWiT, and all of them have Security Now!. So there's lots of ways to watch.

If you want to watch live, we're here about 1:30 in the afternoon every Tuesday, that's 4:30 Eastern, 21:30 UTC, so you can watch live, or you can do like this good-looking family has done and just pop in, and we'll entertain you. I think we put the six year old to sleep, but the rest of them have been wide-eyed for the whole thing.

VISITOR: They slept in the car on the way.

Leo: Yeah. They're going to sleep from now on. Just email tickets@twit.tv. We'll make sure there's room for you. Let's see. I guess that wraps up. Have a great Thanksgiving. You going to stay home?

Steve: I'm going to stay here. I go out, we have a great restaurant we go to. A group of my friends and some local neighbors get together, and we all go down. And then I'll be up for Christmas. But staying down here.

Leo: Good. Well, have a great...

Steve: And I should mention, I referred briefly about some work on SQRL. In the last week we hammered out a local application authentication solution, which we are not going to implement, just for the sake of time. It's time to ship this puppy. But I wanted to go through the logic of it and nail it down so that it was known that it could do that. So it can be used to authenticate, not only to remote servers, but local applications will be able to use it with absolute full security.

So, for example, you could use SQRL to log into LastPass, as an example, instead of using a username and password, even though it's a local app. So we figured out how to do that with a tiny little extension to the protocol, which it turned out very elegantly. But we're going to save that for Version 1.1 or whatever and get SQRL finished.

Leo: Yay. Looking forward to seeing it. Thanks, Steve. Thanks, everybody. We'll see you next time.

Steve: Thanks, Leo.