SECURITY NOW!

Transcript of Episode #823

TLS Confusion Attacks

Description: This week we're going to start by looking at a moment-by-moment reconstruction of a recent Chrome browser attack-and-patch battle. Then we're going to recap last week's industry-wide June patch fest, followed by looking at TikTok's controversial but unsurprising privacy policy update. We need to also cover the wonderful spy novel-ish ANOM sting operation which lowered the boom on as many as 800 criminals. For our happily infrequent Errata section we'll challenge an apparently erroneous statement I made last week. I want to share an interesting laptop data recovery experience which BitLocker made much more complex a few weeks ago which I think our listeners will find interesting. Then we're going to tackle this week's topic of some very troubling research which again demonstrates just how difficult it is to design robustly secure networked systems.

High quality (64 kbps) mp3 audio file URL: http://media.GRC.com/sn/SN-823.mp3
Quarter size (16 kbps) mp3 audio file URL: http://media.GRC.com/sn/sn-823-lq.mp3

SHOW TEASE: It's time for Security Now!. Lots to talk about. Industry-wide Patch Tuesday. Is that such a good idea? Lots of bug fixes. Steve's got a list. Also coming up in just a little bit, TikTok. They got caught collecting your biometric information? Really? And then a great story from Steve about how he fixed a friend's laptop. This was a life-or-death situation, and Steve rode to the rescue. It's all coming up next on Security Now!.

Leo Laporte: This is Security Now! with Steve Gibson, Episode 823, recorded Tuesday, June 15th, 2021: TLS Confusion Attacks.

It's time for Security Now!, the show where we cover your security and privacy online with the man and the myth and the legend, Mr. Steve Gibson.

Steve Gibson: Big on myth.

Leo: You're man and legend. I don't know about the myth part. Good to see you.

Steve: Oh, Leo.

Leo: Uh-oh. What's the matter? Bad guys winning again?

Steve: We're all going to be very depressed by the end of this podcast.

Leo: Uh-oh. Uh-oh.

Steve: But that's the way it goes, unfortunately. And that's the way it goes.

Leo: That's the way it is.

Steve: That's the way it is. We're going to start, however, by looking at I think a fun moment-by-moment reconstruction of a recent Chrome browser attack-and-patch battle. Or maybe it was a pitched patch battle. Anyway, then we're going to recap last week's industry-wide June patch fest, followed by looking at TikTok's controversial but unsurprising privacy policy update. We need to cover also that wonderful spy novel-ish ANOM sting operation which was just breaking as we were recording last week's podcast. And of course it lowered the boom on as many as 800 criminals.

For our happily infrequent Errata section, we'll challenge an apparently erroneous statement I made last week. And actually based on something I just saw The Verge post, it now seems almost assured that it is incorrect. Then I want to share an interesting laptop data recovery experience which BitLocker made much more complex a few weeks ago, which I think our listeners will find interesting. I had it in the show notes last week, just like a note to talk about it, but we didn't have time. I think we will this week.

Then we're going to tackle this week's topic of some very troubling research which again demonstrates just how difficult it is to design robustly secure networked systems, even if nobody makes any mistakes. If there are no bugs, if there are no patches, if nothing is wrong, it still doesn't work. And, yeah. So we will discuss what I'm calling TLS Confusion Attacks, although they are unfortunately named the ALPACA Attacks.

Leo: Oh, yeah. I remember those.

Steve: ALPACA.

Leo: ALPACA.

Steve: Yeah, well, ALPACA, yeah. And we do have a fun, well, not very surprising, but it's a useful Picture of the Week. So overall...

Leo: Is it a picture of an alpaca?

Steve: No.

Leo: Oh, okay. I had to learn recently the difference between an alpaca and a llama. The alpacas...

Steve: They're not the same.

Leo: No. Alpacas are the cute ones. Literally. Look at a picture. Just while I'm doing this ad.

Steve: Okay.

Leo: Look at a picture. Just google "alpaca" and "llama," and tell me if I'm not right. That's how I remember the difference. I am ready to fire up the Picture of the Week, Mr. G.

Steve: Well, this is pretty simple, but I got a little bit of a kick out of it. It's just a single-frame cartoon, shows two bad guys in black hoodie masks. One's got a mask on, the other's got a larger ski mask and glasses. And the first guy says, "Hey, Harvey." Oh, and they're both sitting in front of laptops, doing their dastardly deeds. "Hey, Harvey. Someone hacked into my bank account and stole all the money I made from ransomware. This is so unfair." Anyway.

Leo: It's funny because it's true.

Steve: It's sad, yes.

Leo: That somebody might be the FBI. We don't know. But yeah.

Steve: Yeah. So Google is finding that with offering the world's number one web browser comes being the world's number one web target. Using some reports published by Kaspersky Labs, I was able to reconstruct a timeline of some recent Chrome vulnerabilities and fixes. And what it reveals is instructive. So exactly two months ago, during April 14th and 15th of this year, Kaspersky Labs detected a wave of highly targeted attacks aimed at multiple companies. Those were companies that are paying them to oversee their security, monitor their web traffic, keep an eye on things, thus the way they were able to detect this.

Upon closer analysis, Kaspersky discovered that these attacks were exploiting chained Chrome and Windows zero-day vulnerabilities. Though they were unable to retrieve the exploit code itself, which was used to accomplish remote code execution in Chrome, they were able to obtain and analyze an elevation of privilege exploit that was used to escape the sandbox to obtain system privileges. And remember that, while we tend to focus upon remote execution attacks because those are like so obviously bad, exploitable elevation of privilege vulnerabilities are also extremely powerful since they inherently breach the security boundaries that we rely upon to allow externally provided browser code, JavaScript and WebAssem, to run in our browsers, while curtailing what that code can do. If code can arrange to make itself privileged, then "can do" is what results, and that's not what you want from code that you've obtained from someone you don't really trust.

So in this case the elevation of privilege exploit that they discovered had been fine-tuned to work against the latest and most prominent builds of Windows 10. Well, not only the latest, but dating back to 17763, which was RS5; and included 18362, which was 19H1;

18363, 19H2; 19041, 20H1; and 19042, 20H2. So, I mean, there was some serious attacking going on. And it does all this by exploiting two separate vulnerabilities in Windows OS kernel once it gets free, once it escapes from the browser.

So five days after this discovery, on the 20th, Kaspersky had figured out what was going on, pulled a report together, and reported these vulnerabilities to Microsoft. And two successive CVEs were assigned: 2021-31955, which was an information disclosure vulnerability, that's actually a kernel disclosure, which we'll be getting to when we talk about this last past week's patches; and also 31956, which is the elevation of privilege vulnerability. Both of these in-the-wild vulnerabilities were patched, as I said, last Tuesday as part of June's patch cycle.

The observed attacks were all conducted through Chrome. But as I noted before, Kaspersky was unable to retrieve the JavaScript which fully implemented the entire exploit. But they did have a clue, the Pwn2Own competition that had taken place the preceding week, April 6th through the 8th, which we talked about at the time. And of course Chrome was a prominent target of that. During the competition one of the participating teams was able to successfully demonstrate an exploitation of the Chrome renderer process using a type mismatch bug. And a few days later, on April 12th, the Chromium developers committed two new issues, 1196683 and 1195777, to the open source repository of V8, which as we know is the Chrome and Chromium JavaScript engine. Both were type-related bugs.

One of these bugs, that 6683, patched the vulnerability that was used during Pwn2Own. And both bug fixes were committed together, along with regression tests. Regression tests are JavaScript tests which are used to trigger the vulnerabilities. And of course they're important because you want to keep those vulnerabilities from ever coming back by mistake in the past. Later that same day, a user with the Twitter handle @r4j0x00 published a working remote code execution exploit on GitHub targeting an up-to-date version of Google Chrome. That exploit used the vulnerability from issue 6683 to execute shell code in the context of the browser renderer process. The exploit published to GitHub did not contain a sandbox escape and was therefore intended to only work when the browser was launched, and you wouldn't normally do this, with the command line option "-no-sandbox." So not a huge concern, but still interesting.

Then on the 13th, the day before Kaspersky first became aware of the attacks in the wild, Google released Chrome update 89.0.4389.128 for Windows, Mac, and Linux, with fixes for two vulnerabilities. The one, that CVE-2021-21220, used during Pwn2Own was one of them. However, since several of Kaspersky's customers, who were attacked on those first two dates, April 14th and 15th, already had their Chrome browsers updated, Kaspersky believes that the Pwn2Own-originated vulnerability was not used in those attacks because it would have failed against those just-updated browsers.

Then, the next day, on the 14th, Google moved from their version 89 to version 90. And of course we covered that move also. There was a whole bunch of features fixed. That was a major planned feature release. And that brought out 90.0.4430.72 for Windows, Mac, and Linux. This release closed the door on 37 vulnerabilities. And on the same day, on that 14th, a new Chrome exploit was posted on GitHub, which of course released it to the public. That newly published exploit used that second vulnerability, the 5777 which, even though it had been committed on the 12th, still worked on the just-released Chrome 90 from the 14th.

So, I mean, there was just like a - 89 apparently may have fixed it. But technically, 90 regressed because it wasn't part of the planned release in 90, and that change had just been committed prior to, well, probably after 90 essentially was RTM'd. So that problem, this 5777 problem, was fixed six days later on the 20th. Kaspersky suspects that the attackers were also able to use the JavaScript file containing the regression test for that second problem to quickly develop the exploit and were probably using the second vulnerability in their attacks. So talk about cat and mouse.

What do we learn from this? We learn that attackers are extremely active and deft at scrutinizing everything that happens in public view. They're looking everywhere at once. In this instance, the results of the Pwn2Own competition likely primed them to be on the lookout for Chromium commits that would soon follow. And they did. The Chromium project is at a disadvantage of inherently being open. It's a good thing to create JavaScript regression tests to make sure that bugs which have been fixed never return. But as likely happened here, the appearance of the regression tests probably predated by eight days the deployment of an updated Chrome browser which fixed those flaws, due to just like a one-day timing uncertainty. That created that eight-day exploitation window during which a problem was publicly known and documented, but not yet patched, and in the hands of Chrome's users.

We can also see from this almost minute-by-minute back-and-forth, that those who are responsible for security can never take a vacation day from their jobs. Newly discovered vulnerabilities must be immediately stomped out, and those fixes must also be rapidly deployed. And we can see that today's web browser attackers themselves are hyper-vigilant. They know that they won't have much time to leverage any transient advantage that they might be able to briefly obtain.

And, finally, we learn that we should be so thankful that Microsoft had the wisdom to scrap their own independent web browser development in favor of their adoption of the Chromium project. Web browsers can no longer be the domain of massive, slow-moving, bureaucratic behemoths. Web browsers is no longer where Microsoft should be. Putting any modern web browser up for the world to attack requires far too much agility. Google has clearly optimized their Chromium group for short-cycle, nearly instantaneous response. And in an example like this, nothing less would be sufficient to protect Chrome users from today's attackers.

So, whew. We normally just sort of step back and look at the big major version changes. But I thought that this was really interesting to see just how quickly any instance of a vulnerability is jumped on. And it's not clear what percentage of vulnerabilities are being discovered themselves by the attackers, and how many, like in this instance, they're able to allow other people to discover and, just as a consequence of timing uncertainty, are still able to create an attack window that is unfortunately useful to them. They clearly had other pieces of ammunition ready in order to create the chain. They already had those Windows zero-days which got closed last week, yet they still needed a way to deploy them. And the second that Chrome surfaced a means by which they could, that got chained into an attack, and Kaspersky found their customers being victimized through that brief little window of opportunity. It's the world we're in today. Wow.

Leo: Think most attacks are chained like that these days?

Steve: Yeah. It's really rare - an example of one that isn't, for example, is a credential breach in RDP. We know that bad guys, in fact, this is how the attack that DarkSide used against Colonial Pipeline, we learned that it was almost certainly a credential for their VPN that was purchased on the dark web. And so there is was like, wait. This is the username and password? Great. Thank you very much. And they just logged right into the network.

Leo: Oh, just [crosstalk].

Steve: Exactly. Exactly. But what we're seeing is surprisingly sophisticated chained attacks.

Leo: Yeah, because that wasn't an exploit. That wouldn't take advantage of exploits. That was just a breach, you know.

Steve: That was actually a front door for which the keys had gotten loose.

Leo: But exploits are, that makes sense, increasingly chained because one doesn't do it by itself. But yeah, that makes sense.

Steve: Yeah. And the reason one doesn't is that typically you do have layers of security. First you're in the Chrome code, so you've got to somehow get out of the Chrome code. Then once you're out of the Chrome code, then you're still in the sandbox. Now you've got to get out of the Chrome browser's sandbox in order to have any contact with the operating system. Once you have contact with the operating system, you've got to be able to do something nefarious with that. So now you need an elevation of privilege, not only to get out of the sandbox, but then to get into the kernel. And so it's, yeah, it's a series of things you need. And what's often the case is that they have a toolkit of exploitable, potentially exploitable pieces where they're just waiting for one more piece to fall into their lap, and then they can put them in a chain and go.

Leo: Yeah, yeah. Layered security in response to layered attacks and vice versa. That makes sense, yeah. And that's why it's so hard to defend, frankly.

Steve: Yes, yes. And speaking of which, last Tuesday was the industry-wide Patch Tuesday. And as I noted last week...

Leo: You mean wider than Microsoft? Everybody's doing it now?

Steve: Yeah, exactly. As I noted last week, many other companies have decided to synchronize their patch cycles with Microsoft. And you kind of wonder. Is it to hide? It's like, oh, no, everybody's doing it.

Leo: Yeah, because I don't think it's ideal that you get a bunch of patches in a bunch of different things all at once. That's just a recipe for disaster.

Steve: Yeah. Intel in this case fixed 73...

Leo: Oh. Geez.

Steve: Yeah, security vulnerabilities which included some that were severe, impacting the UEFI/BIOS firmware for their processors, as well as their Bluetooth products, their Active Management Technology tools, which as we know runs down on the motherboard underneath everything else. That's the AMT stuff. And its NUC mini PC offerings, including some problems in its own security library. Among other things in the security library there was a problem with a random number generator not being as random as hoped. So I guess we - how many years have we been talking about these problems, Leo, with insufficiently random, random number generators.

Leo: At least since the show began.

Steve: Yeah.

Leo: That's why it's called a "pseudorandom number generator."

Steve: That's right. So among those that were fixed, there were some rated critical; though, interestingly, Intel boasted that most were found internally. Intel's Jerry Bryant, he's the guy who is their spokesman, and we've quoted him in the past. In this case he said: "Today we released 29 security advisories addressing 73 vulnerabilities. 40 of those 73, or 55%, were found internally through our own proactive security research. Of the remaining 33 CVEs being addressed, 29, or 40%, were reported through our bug bounty program. Overall, 95% of the issues being addressed today were the result of our ongoing investments in security assurance, which is consistent with our 2020 Product Security Report." To which I thought, that's right, Jerry. And if only you had found those problems before you deployed the buggy code into the wild, the world would not now need to scramble around to update our now-known defective devices before the forces of darkness are able to reverse engineer those changes to use them against all of the unwitting. But I suppose better late than never.

Anyone using an Intel NUC and I happen to be a big NUC fan, and I expect that's going to be my chosen platform moving forward ought to check to see whether there's an update to their device's firmware waiting for them over at Intel. Intel doesn't appear to offer a firmware appraisal tool. I went looking for it yesterday, and there wasn't one that I could find. So you need to figure out which model of device you have - the good news is you can just turn it over and look at its belly - and which firmware version it's running. You can actually do that through the Windows Management Instrumentation, the WMI stuff, which I did, in order to get the BIOS version or just, you know, reboot and see what it says as it's coming up, and then see whether you've got the latest available. Probably worth doing.

By far the winner of this month's critical patch derby was Adobe, who addressed 41 CVEs in their own Patch Tuesday last week. 21 of them were rated critical in severity. They impacted Acrobat and Reader, Photoshop, Creative Cloud Desktop, their RoboHelp Server, Adobe After Effects, and Adobe Animate.

Leo: Ai yai yai.

Steve: I know.

Leo: I'm liking this idea less and less. I mean, gosh, I've got Microsoft, Intel, and Adobe patches all coming in at the same time?

Steve: Just give up on Tuesday.

Leo: That's a nightmare. It's not good.

Steve: And you know, I know that things seem sort of gloomy in the software industry right now. But really, we should count our blessings that Adobe never decided to do an operating system.

Leo: Yeah, no kidding. That's a silver lining, yes.

Steve: Can you imagine? They were never able to make Flash safe. So thank god we're not, like, some chunk of the industry doesn't have Adobe OS. That would, you know, no, no. Just no. And speaking of operating systems, however, the original second Tuesday patcher, Microsoft, managed to break last month's 16-month low mark, which last month was just 55 flaws, by coming in this month with only 50 or 51, depending upon how one counts. However, last month had only three zero-days, and this month we have six exploits appearing in the wild. The vulnerabilities fixed span Windows desktop, SharePoint Server, Windows kernel, and Outlook.

Two of the vulnerabilities are related to a separate vulnerability in Acrobat Reader for which Microsoft released fixes in its Enhanced Cryptographic Provider. By leveraging those flaws, an attacker could elevate their privileges on the targeted system if they trick a user into opening a specially crafted PDF file using a vulnerable version of Acrobat or Reader from within a not-yet-patched version of Windows. So that got patched last week. Microsoft stated in its advisories that it's seen these vulnerabilities being exploited in the wild.

In other words, they still work. Adobe had previously fixed this in its previous month's security update. So if you're using Acrobat or Reader, having both patched would be safest. That is to say, both Acrobat or Reader and Windows, because it's a combination. You need problems in both in order to get exploited. Fix them both. Again, layers are good.

One of the critical vulnerabilities fixed this month was in the Windows Defender codebase which would have allowed an attacker to execute remote code on the host machine. The good news is that Defender is continually keeping itself up to date independently, even under Windows 7. Microsoft's Office MSGraph component also had a remote code execution vulnerability that could be used to deliver a malicious payload to a victim's machine through Microsoft Office, since the built-in MSGraph component can be embedded in most Office documents, which will then exploit the flaw when any one of those documents which has the MSGraph component embedded in it is opened. So again, these are all ways that email, right, phishing email can attack a user because they just do something that should be fine, but isn't.

Leo: Ugh.

Steve: Yeah. It's just so wrong. Another vulnerability being actively exploited - these are all things like, you know, these were all loose already; right? I mean, Windows users were being hurt by these things, these six different things that got fixed last Tuesday. In this case, there's a privilege escalation flaw, or elevation, in the DWM Core Library, which is what Microsoft calls it. DWM is the Desktop Windows Manager. It can be exploited by running an executable or script on the local machine. It has a CVSS of only 8.4, so it's bad. Microsoft only labeled it as "important," even though, again, actively exploited in the wild? Oh, yeah. I think maybe you need to get that fixed.

And finally, CVE-2021-31955, that's the flaw that I talked about before relative to Chrome. That's the flaw that allows an attacker to read the contents of the Windows OS kernel memory. They called it an "information disclosure." I call it "game over." The kernel is the land of exploitable secrets, and Microsoft stated in their advisory that it has also been actively exploited in the wild. So, yeah, another Patch Tuesday and a few hundred overall, if you consider Intel, Adobe, and Microsoft, a few hundred fewer outstanding bugs in the software the world is using. That would be a hopeful sign; right? A few hundred fewer? If we didn't appear to be creating them anew at an even faster clip than we're eliminating them.

I did know last week, but I didn't have a chance to get to it, but it was in one of those tabs that I had left open in Firefox that I mentioned. And Leo, you know, unlike you, being "hip" is not a designation I have ever at any prior point in my life laid claim to. I just never had that.

Leo: Nor I, Steve. Nor I.

Steve: Well, I don't know. You're way hipper than I am. You know what a lot of this stuff is. All indications are that I'm headed in the other direction.

Leo: You and me both, bud.

Steve: Those who are hip will know that TikTok is a popular short-form video sharing service.

Leo: Oh, well, I'm that hip, yes.

Steve: See? Okay, yeah.

Leo: I knew that.

Steve: Not I. You know?

Leo: Okay.

Steve: Anyway, I'm sure that they're not happy that they recently made the tech news press due to a somewhat chilling explicit change in their service's privacy policy. Okay, now, stepping back a bit, there's some interesting back story here. It's likely that the recent changes to their privacy policy followed from a $92 million settlement of a class-action lawsuit. We've talked about the super-tight Illinois Biometric Information Privacy Act, abbreviated "BIPA," B-I-P-A. Illinois is where privacy lawsuits go to be filed since BIPA, as we've previously covered extensively, pretty much takes no prisoners.

In this case, the lawsuit was originally filed just over a year ago in May of 2020. And the group of plaintiffs in this suit, which was consolidated from more than 20 separate cases originally filed against TikTok, reached a preliminary settlement just before the end of this February. The suit alleges that TikTok violates BIPA - and boy, does it - when it collected and shared the personal and biometric information of its users without first obtaining their consent.

Attorney Ekwan Rhow, who is with the firm - and I had to read this a couple times, I kid you not - of Bird, Marella, Boxer, Wolpert, Nessim, Drooks, Lincenberg & Rhow...

Leo: That's my password.

Steve: That's right. They say use a bunch of words, string them together...

Leo: Yeah.

Steve: Yes. That probably is one of their passwords.

Leo: I don't think they're in the dictionary, to be honest with you.

Steve: Boy. Anyway, Ekwan was quoted saying: "This is one of the largest settlements ever achieved in a consumer BIPA case, and one of the largest privacy class action settlements. It presents an excellent recovery for the class, and it serves as a reminder to corporations that privacy matters, and they will be held accountable for violating consumers' rights." Another attorney in the case, Beth Fegan, added: "Illinois is on the cutting edge of privacy law, and this settlement enforces those crucial protections. Biometric information is among the most sensitive of private information. It's crucial that privacy and identity is protected by stalwart governance to guard against underhanded attempts at theft." And yeah, I'd love to know, and you and I, Leo, agree on this also, what portion of the settlement went to the members of the aggrieved class.

Leo: More than half, guaranteed, yeah, yeah.

Steve: And what portion those oh-so-concerned attorneys retained for themselves.

Leo: Yeah.

Steve: But that's beside the point. And although 92 million is an arresting sum, it's actually small potatoes when measured against Facebook's $650 million BIPA settlement.

Leo: Wow.

Steve: That $650 million settlement was arrived at only after the judge rejected the previous $550 million proposal, arguing that the smaller figure would not be enough to compensate the sheer number of people in the class - probably all Facebook users - based on the penalties laid out in the BIPA law.

Okay. So of course this is not the first time TikTok has made the tech press news. They were also in legal hot water when the U.S. federal government was considering a ban on TikTok after privacy experts warned that the government of China might have access to personal information gathered through the app. Okay. But in any event, the recent settlement alleged that TikTok was using clandestinely captured biometric and personal data from users in the U.S. to target ads without meeting the informed consent requirements of Illinois's very rigid biometric state law, you know, privacy enforcing state law regarding biometrics. And we've talked about just how facial recognition, BIPA has been used to slam the door on that. So as part of the settlement, TikTok agreed to avoid collecting or storing biometric information, biometric identifiers, geolocation, or GPS data unless expressly disclosed in its privacy policy, which of course, fine print, who reads that? But still.

We have TikTok's new privacy policy which is what has garnered so much attention and concern. It says: "We collect information about your approximate location, including location information based on your SIM card and/or IP address. With your permission, we may also collect precise location data such as GPS."

Then under Image and Audio Information they said: "We may collect information about the images and audio that are part of your user content, such as identifying the objects and scenery that appear, the existence and location within a image of face and body features and attributes, the nature of the audio, and the text of the words spoken in your user content. We may collect this information to enable special video effects, for content moderation, for demographic classification, for content and ad recommendations, and for other non-personally identifying operations." And, finally: "We may collect biometric identifiers and biometric information as defined under U.S. laws, such as faceprints and voiceprints, from your user content. Where required by law, we will seek any required permissions from you prior to any such collection."

So TikTok is now, finally, formally disclosing what they weren't before, thus the BIPA class action, which cost them $92 million, that they were presumably doing, collecting biometric information, and then they say "such as," but apparently also not limited to, faceprints and voiceprints. And it should escape no one that the whole point of biometrics, that is, the metrics of the bio, is identification. Again, not new information. But now this is out of the shadows. And given that only five states in the U.S. currently that's California, Illinois, New York, Texas, and Washington have laws which restrict the collection of biometric data, the updated privacy disclosure likely means that TikTok will not be required to obtain explicit permission from its users in all the other states of the U.S. In other words, users are consenting to have their biometric data collected simply by agreeing to TikTok's terms of service, which most users are just going to say "Yeah, yeah, whatever" to. So maybe they don't care. The onus is on the user, of course, to decide, you know, to know what it is they're agreeing to.

On the other hand, that brings us to something else that happened recently that we have not yet had a chance to talk about. And that's iOS 14.5, speaking of users explicitly agreeing to things. One of the changes which has been made in iOS 14.5 was it required apps to obtain explicit tracking permission from their users. It added the provision and the requirement for any apps wishing to track their users' activity across other companies' apps and websites to expressly request and obtain such permission. Overall, the feedback has been a resounding "No, thanks."

The application analytics company, Flurry, has been sampling 2.5 million users since the release of iOS 14.5. Their research has shown that the worldwide global "opt into tracking" rate has settled at about 25%. In other words, 75% of users globally are saying, oh, I have a choice? Then no thank you. And users in the United States are even more tracking shy. For the U.S., the "It's okay, go ahead and track me" rate has settled out at around 14%. In other words, 86% of U.S. users want to see tracking 86'd from their online experiences.

So you know, Leo, with the anti-tracking stovepiping which Firefox is doing, with Google's moves to sort of, you know, to move from third-party tracking to FLoC, and with FLoC having such problems, I wonder if ultimately this thing, the whole tracking-ness is going to go by the wayside. It's just going to have been sort of an upheaval that we went through because it happened when we weren't looking. It happened because people couldn't really see it. But then some companies started making privacy and non-tracking a feature. Users decided, you know, we don't want it. And then when they were asked, they said no. So if you ask, then it's going to fail.

Leo: And of course Apple's new iCloud Plus relay system is very much privacy - I think these companies are hearing consumers say we don't want tracking. The thing that I think is important to note is what Google and others are doing is eliminating third-party tracking. But at no point has Google said, well, we're not going to track you. Google first-party tracking, in my opinion all it does is say that the only kind of tracking will be first-party tracking, which means companies like Facebook, google, Apple, Microsoft, all will have a huge dominant position going forward because no one can compete with them.

Steve: Right.

Leo: But they will continue. That's why Spotify is buying podcast companies, because you can't track in a podcast unless you own the app. And then you can do first-party tracking. If you think you watch stuff on YouTube and Google's not paying attention to what you watch, or you look at videos on TikTok, and TikTok's not paying attention to what you watch, you're missing how this stuff works. That's how it works.

Steve: Yup.

Leo: So I don't think tracking's going away. I think third-party tracking is going away. And I don't know if that's exactly the result people intend because all it does is entrenches the incumbents. And that's not what you want, either, I don't think.

Steve: So the news of ANOM was breaking just as we were recording last week's podcast.

Leo: Right. And this is something, by the way, I just want to give you credit because a lot - we were talking about this on MacBreak Weekly. I don't know if you heard. But everybody jumps on these things so that they can get the hits; right? Even if we don't know what happened, even if they're going to make up something that's not true. But you, I give you a lot of credit, resist that temptation and do the research before we start talking about something like this. And I really appreciate you doing it. And of course details emerge over time.

Steve: Exactly. So today I'm able to provide a comprehensive description of this classic, though quite high-tech sting operation. Europol's press release of last Tuesday, the 8th, was headlined "800 Criminals Arrested in Biggest Ever Law Enforcement Operation Against Encrypted Communication." So the ANOM sting operation, which is known as Operation Trojan Shield by the U.S. Federal Bureau of Investigation, our FBI, was called Operation Ironside by the AFP, that's the Australian Federal Police. It is/was a collaboration by law enforcement agencies from as many as 20 countries. It ran from late 2018 when it got off the ground, through just now, 2021.

And during that time the operation intercepted and exploited the intelligence obtained through 27 million messages. Imagine just processing, I mean, like reading 27 million messages. You'd want to be like handing them out. Here's your 10,000 for today. Have your staff take a look at them. And here, a different country law enforcement, you take this 10,000 and let us know if you find anything. So the messages were sent through and intercepted by the supposedly secure smartphone-based messaging app ANOM. The ANOM service, which was widely used by criminals, was actually a trojan horse, covertly distributed by the U.S. FBI and the Australian Federal Police, the AFP. It enabled them to monitor all communications taking place across the network. And finally, through collaboration with other law enforcement agencies worldwide, the operation resulted in the arrest, as I said, and as that headline said, of over 800 subjects allegedly involved in criminal activity across 16 different countries.

Among the arrested people were alleged members of Australian-based Italian mafia, Albanian organized crime, outlaw motorcycle gangs, drug syndicates, and other organized crime groups. At its height, the ANOM service grew to a service of more than 12,000 encrypted devices being held by over 300 criminal syndicates operating across more than 100 countries. The platform's goal was to target deliberately, I mean, this whole thing was deliberately created. And I'll get to that in a second. Its goal was to target global organized crime, drug trafficking, and money laundering organizations, regardless of where they operated.

Very cleverly, the service was designed to appeal to what the underworld wanted by offering an encrypted device with features sought by the organized crime networks, including things like remote wipe and duress passwords. We've talked about those before. Duress passwords are those which could be given when a bad guy is under duress, which would have the opposite of the intended effect. It would wipe rather than unlock a device. So you could just see them rubbing their hands together. It's like, oh, goodie, this has the features that we need. These features gradually persuaded criminal networks to adopt the device, thus the way it became so widespread.

And the way this happened is the stuff of spy novels. First, the shutdown of the Canadian secure messaging platform Phantom Secure in early 2018 left international criminals who were using it in need of an alternative system for secure communication. And around the same time the FBI branch office in San Diego, California was working with a person who had been developing his own next-generation encrypted device for use by crime, by criminal networks. That individual had been nabbed and was facing charges. So he cooperated with the FBI in exchange for a reduced sentence.

He offered to develop ANOM and to then distribute it to the underworld through their existing networks with which he was familiar. The first communication devices with ANOM were offered by this informant to three former distributors of that now shut down Phantom Secure system starting around October of 2018. So that's when this all began to get off the ground. The FBI also negotiated with an unnamed third country to set up a communication interception, but based on a court order that allowed passing the information back to the FBI.

Leo: That's because the FBI's restricted on how it can collect information on U.S. citizens.

Steve: Exactly.

Leo: So this is a real backdoor to that.

Steve: Exactly, uh-huh.

Leo: But okay.

Steve: Yes. Since October 2019, ANOM communications have been passed on to the FBI by this third country, exactly as you said, Leo, to get around our own laws. During the culmination of ANOM's operation, a series of large-scale law enforcement actions were executed - that's what just happened a week ago - across 16 countries, resulting in more than 700 residential searches, more than 800 arrests, and the seizure of over eight tons of cocaine, 22 tons of cannabis and cannabis resin, two tons of synthetic drugs (amphetamine and methamphetamine), six tons of synthetic drug precursors, 250 firearms, 55 luxury vehicles, and over $48 million in various worldwide currencies and cryptocurrencies. And innumerable spin-off operations are still planned for the weeks to come.

Leo: The reason this all happened at once is I think that the jig was up; right? Somebody tipped them that these ANOM phones weren't secure. And so they had to roll it up, roll up the network, yeah.

Steve: Right. And you'd have to do the whole thing at once.

Leo: All at once, yeah.

Steve: Yes. So Wikipedia had some interesting background information about the ANOM devices. Wikipedia said:

"The ANOM devices consisted of a messaging app running on smartphones that had been specially modified to disable normal functions such as voice telephony, email, or location services. After checking that normal functionality was disabled, the messaging apps then communicated with one another via supposedly secure proxy servers, which then copied all sent messages to servers controlled by the FBI. The FBI could then decrypt the messages with a private key associated with the message, without ever needing remote access to the devices. The devices also had a fixed identification number assigned to each user, allowing messages from the same user to be connected to that user. According to a since-deleted Reddit post discovered by Motherboard, the ANOM app was for Android. A WordPress blog post described the app as using a custom Android OS."

Leo: That makes sense. What else are you going to use? Yeah.

Steve: It does, yeah. "About 50 devices were distributed in Australia for beta testing from October 2018. The intercepted communications showed that every device was used for criminal activities, primarily being used by organized criminal gangs. Use of the app spread through word of mouth, and was also encouraged by undercover agents. Drug trafficker Hakan Ayik was identified 'as someone who was trusted and was going to be able to successfully distribute this platform,' and without his knowledge was encouraged by undercover agents to use and sell the devices on the black market, further expanding its use. After users of the devices requested smaller and newer phones" - hey, Rev. 2.

Leo: Hey, yo, this phone is really big. You got an iPhone, man? I don't know why I'm using this big old thing. You know what the best part is? They charged them $2,000 every six months to use the phone. Which is brilliant; right?

Steve: Beautiful. Brilliant, yeah.

Leo: Don't give them away because then people might not trust them. Oh, I paid $2,000 for this. It must be something.

Steve: Must be some deep crypto in this thing.

Leo: Some good stuff on here, man.

Steve: That's right. This is really uncrackable.

Leo: Oh, uncrackable. It's expensive.

Steve: It's interesting that the most commonly used languages on the app were Dutch, German, and Swedish.

Leo: Oh, interesting, huh.

Steve: "So after a slow start," writes Wikipedia, "the rate of distribution of ANOM increased from mid-2019. By October 2019, there were several hundred users. By May 2021, a month ago, there had been 11,800 devices with ANOM installed, of which about 9,000 were in active use. New Zealand had 57 users of the ANOM communication system. The Swedish Police had access to conversations from 1,600 users, of which they focused their surveillance on 600. Europol stated 27 million messages were collected from ANOM devices across those 100 countries." And then the article concludes: "Some skepticism of the app did exist. One March 2021 WordPress blog post called the app a 'scam.'" And Leo, that may be something like what you're referring to where people were beginning to say, uh, really?

Leo: Yeah.

Steve: And anyway, I would call it a significant success.

Leo: Notably, the guy you mentioned, the Turkish drug lord who really spurred the growth of this, has not been arrested. He's living apparently a lavish lifestyle, I think is what the Australian police said, in Istanbul. And I guess because he's in Turkey they can't arrest him. But very wise, the AFP says, you know, Hakan, everybody thinks you set them up. It might be wise to turn yourself in now because either we're going to get you, or they're going to get you. So we'll see. I have to say, though, it does raise the issue of why authorities want backdoors in devices. I mean, this is a goldmine for them. And immediately in Australia more laws were passed to encourage backdoors in encryption because...

Steve: Yeah, I mean, that's one of the big focuses on the podcast is the question, what's going to happen with this?

Leo: I think it's a big PR victory for the AFP and the FBI. And I think that's really what this is mostly about. But we'll see. Let's watch the prosecutions. I mean, it's a lot of this is pot. Come on, man. It's actually legal in many states in the United States.

Steve: Yeah. And of course the reason for the success, I'm sure our listeners already got this, is that this was distributed explicitly to criminal gangs. And every message, every transaction was being monitored. That's not something that our FBI could do under any circumstances in the United States. The most that I can see happening is that the equivalent of a search warrant, a highly targeted search warrant, where a judge, hopefully a judge with judgment, agrees that there is probable cause to believe that something wrong is going on. And so selective decryption or tapping, very much like an old-school phone tap, right, where the FBI says we need to bring up a phone tap on these people because we need information that we have reason to believe we will get there, not we're going to monitor every conversation of everyone in the country. Which with the much smaller ANOM group is what was happening. So it's going to be really interesting how this shakes out.

Leo: Yeah, no kidding.

Steve: So, Errata.

Leo: Never. Never.

Steve: Windows 10, the last Windows ever?

Leo: Yeah?

Steve: Apparently not.

Leo: No.

Steve: I was absolutely certain that we were clearly and formally told by representatives of Microsoft...

Leo: Yes, I know.

Steve: ...that following Windows 8.1, the next Windows, to be called 10, would be free, would create - and remember this? - new revenue opportunities for Microsoft. In other words, that users of this next Windows would become profit centers themselves for Microsoft. And that it would be the last version of Windows ever.

Leo: We've all been saying that.

Steve: Yes.

Leo: Over and over.

Steve: And I was stunned last week during Windows Weekly, in her conversation with you, Leo, and Paul, when Mary Jo Foley said, "Uh, nope. Microsoft never said that."

Leo: Unh-unh.

Steve: And so I put this under Errata since...

Leo: Jerry said it. Jerry said it.

Steve: Exactly. Since if I had been spreading an unfounded rumor, at least I wasn't alone.

Leo: No, I said it, too, many, many, many times. You probably got it from me.

Steve: Yeah, well, that's what we all believed. Rich Woods of XDA Developers, he tweeted on the 11th last week, he said: "The most mind-blowing news story from this week was broken by @maryjofoley on Windows Weekly..."

Leo: Oh, thanks, Rich.

Steve: "...and it's that Microsoft never said Windows 10 was the last version of Windows. One developer evangelist said it, Microsoft never corrected it, and everyone ran with it. It became lore," wrote Rich.

Leo: I said it on the radio all the time, so I guess I'm going to have to do an errata, too. Thank you. You're right. Error.

Steve: So it turns out that back in 2015, Microsoft's developer evangelist Jerry Nixon stated that Windows 10 was the last version of Windows. Quoting him exactly, he said: "Right now we're releasing Windows 10. And because Windows 10 is the last version of Windows, we're all still working on Windows 10." In other words, he said that "we're releasing it" and "we're still working on it," clearly implying that Windows 10 would be an ongoing effort. And indeed, that what's we've seen. It would be difficult to say with a straight face, however, that this has all gone smoothly, what with Windows 10 2004 being released in 2010. What? Anyway, what a mess.

Last Wednesday, Mary Jo Foley explained that Microsoft themselves never publicly said, in plain language, that Windows 10 is the last version of the Windows operating system. Their developer evangelist Jerry Nixon said it, and backed it up with evidence at the time. And as Mary Jo noted, Microsoft's PR team never denied it. So no one said he was wrong. Maybe they thought it was the last one.

Leo: You can even read, though, now that I'm reading the actual quote, you could even read that like the latest version of Windows. Like it's not necessarily we're not - it's not explicitly we're never releasing another version.

Steve: True.

Leo: It's the last version of Windows. Well, it is. It's the last version of Windows; right? I mean...

Steve: So far.

Leo: It is still the last version of Windows. Not the last of all ever. Just the latest, as in last as in latest. So I'm not even sure Microsoft realized that he had said something that people would interpret that way. So I don't know how, but this is very common in our industry, any industry; you know?

Steve: And you know, Leo, it may have just been we're just so tired that we were hoping it was true.

Leo: We want it to be.

Steve: Please, oh, god, please.

Leo: We all want it to be the last ever.

Steve: Just please stop. Stop the madness.

Leo: That's why it took off. We all wanted it to be the last version. I think that's exactly what happened. We were hoping it was the last version.

Steve: Oh, oh, does this mean what I think it means?

Leo: I don't think - and by the way, if you listened to that show, you heard Paul, Mary Jo, and I talk about what Windows 11's going to be, and none of the things you and I would like it to be, which is more secure, more reliable, rewritten from the ground up, all of that. It's just going to be a nice thin glaze on top of the old doughnut.

Steve: Yes. Apparently if by mistake you bump into the corner of a window, it will no longer draw blood.

Leo: Yeah. That's it. That's exactly right. No sharp corners. It's a childproof version of Windows. It looks very Mac-like, according to the releases that have come out in the last day.

Steve: Well, good. I like the way the Mac looks. It could use a little bit of an upgrade.

Leo: It's pretty, yeah.

Steve: I did want to follow up once again, real briefly, on "Project Hail Mary." Lorrie loved the book.

Leo: Oh, good.

Steve: She blew through it, as I knew she would. At one point she was standing - I just had to chuckle. She was standing in front of the stove, reading the book with it open in front of her as popcorn was popping.

Leo: See, that's why I do audiobooks. I invariably pop my popcorn listening to an audiobook. I'm completely with her on that.

Steve: Yeah. And I will say she, because she's just such a dear heart, she got a little choked up at the end.

Leo: Oh, yes, very touching.

Steve: With the way everything turned out. I thought, oh, honey, that's cute.

Leo: But actually Lisa - so Lisa had the same reaction. Loved it. And she said, "Give me another one like this."

Steve: I know.

Leo: What do you suggest?

Steve: Well, and this is - perfect segue. I wrote in my notes, seeing that she loved it so much, I gave her the first of the Honor Harrington books. But after a couple of hours she asked whether the entire book would be about military space ordnance.

Leo: By the way.

Steve: And I said...

Leo: Yes.

Steve: Uh, yeah. There's a lot of that.

Leo: It's not a woman's sci-fi book, I don't think.

Steve: No. So now she's reading Daniel Suarez's "Daemon."

Leo: Good choice.

Steve: And that one may be a hit.

Leo: I think so.

Steve: She did, she's read enough of it that last night she said, "This is creepy." And I said yeah, yeah.

Leo: The thing that Lisa likes about Andy Weir's books, both "The Martian" and "Hail Mary," is the humor, the humanity of it.

Steve: Yes.

Leo: And I've been trying - and first I thought, maybe the "Ringworld" series by Larry Niven. But she said, "No, it's too old. I want something newer." So I said maybe Peter F. Hamilton's "Fallen Dragon." But I think it's a little dry. I think she wants something more human scale. So I'm open to any suggestions for her, something recent.

Steve: Yeah, and "Hitchhiker's" is too slapstick.

Leo: Yeah, I don't know if she'd like that. Yeah. I know what she likes because the thing about "Hail Mary" is very similar to "The Martian" in the sense that the protagonist has this great wry style and sense of humor. Very much that sense of humor is what she likes.

Steve: What about "Artemis" in the middle?

Leo: "Artemis" is good. I think she gave up on it. She tried it, didn't like it.

Steve: Yeah, and I'm the same way. I don't know of anything else like that. Normally sci-fi is, well, I mean, there's like New York Times Bestseller sci-fi. I read something awful, can't even remember now what it was.

Leo: It's always got lurid covers with monsters' wrapped tentacles around buxom women and stuff. It's not good.

Steve: Yeah, it's not good.

Leo: It's not good. Actually Stacey Higginbotham has recommended a series that I just bought on Amazon. I'll let you know. You might like it. I think unfortunately there's a considerable amount of space ordnance in it, which means you and I will enjoy it. But I don't know if it's for everyone. Let me see if I can find the name of that series because she really thought it was good.

Steve: Well, and of course Suarez's "Daemon" is not about that at all.

Leo: No.

Steve: It's on Earth, and it's like creepy possible future.

Leo: It's so fast moving and fun and just wonderful. She recommended the - it's by Arkady Martine. It's a series, the first book of which is called "A Memory Called Empire." And the series is the Teixcalaan series. But just remember Arkady Martine. It's written by a woman.

Steve: You know, Lorrie may have never read the "Foundation" series.

Leo: See, I think that's wonderful. And I know Lisa hasn't. But she'll say it's too old.

Steve: Oh, gosh.

Leo: I know.

Steve: And it's going to be, have you seen the previews of what's coming from...

Leo: Oh, no.

Steve: The "Foundation" previews?

Leo: Oh, I'm going to go look at those as soon as the show's over.

Steve: Oh, Leo. Oh, oh, oh, oh. It's - oh.

Leo: Now, have you read it recently? Because I think, as great as that series is, it may...

Steve: I need to reread it.

Leo: It may feel a little dated. Somebody told me that they went back - this is not unusual. You go back to old sci-fi, and times have changed a little bit. Heinlein's really a good example of that, where it just feels dated. So I don't know if - I don't know. But I'm going to go watch those. That's great.

Steve: Okay. So I think our listeners will find this interesting. A colleague of Lorrie's had his laptop stop booting. It contained irreplaceable data - of course, old story - that had not been backed up.

Leo: Of course.

Steve: He had a whole bunch of EEGs which had been recorded and were there. His Ph.D. thesis. And it just - it wouldn't boot. So he and Lorrie were chatting, and she said send it, you know, Steve will take a look at it.

Leo: Wow. That was nice of her.

Steve: And of course I was happy to.

Leo: Oh, good.

Steve: And so it was actually three weeks ago today because it arrived on a Monday. And so I brought it with me here on podcast Tuesday. And it was sitting right here to my left. And I had turned it on in the morning when I was still working on the podcast. And it came up. Everything looked good. I got the little 3D blue Windows thing. It had Windows 10. Oh, it was a Lenovo, a P51s.

Leo: Oh, that's a nice laptop. That's a good, solid laptop.

Steve: Yes. Absolutely beautiful solid laptop.

Leo: Yeah.

Steve: And so I got the little spinning white dots, the little rollercoaster dots, and they spun a while. And then it came up with that unhappy smiley face or frowny face, with an error I'd never seen before: WHEA_UNCORRECTABLE_ERROR. And I thought, oh, okay. So then it said, don't worry, we're going to fix this for you. And so it rebooted itself, and it went into the automatic repairing Windows, and it did that for a while. Then it came up and said, "Could not fix your problem. Try recovering your drive." And so now I got the screen where you had multiple options of, like, restoring from a system restore, uninstalling recent updates, backing up to an image that had been made, those sorts of things.

Oh, and I forgot, I had googled "whea uncorrectable error." And of course as Mr. SpinRite, "uncorrectable" sounds like a sector; right? And I didn't know. But googling didn't produce anything definitive. For example, Microsoft was saying, oh, this could be caused by a recent update, so back out of any updates. So I thought, well, okay. So I tried that. It wouldn't work. I tried uninstalling, I mean, like nothing succeeded. Very quickly, when I tried any of those things, it said can't do. So I thought, okay, what are my BIOS options? Rebooted the system. It had diagnostics built in. So I thought, oh. So ran the diagnostics. And everything except the mass storage, the storage entry, it failed. It failed retests. And so I thought, okay, well, that sounds like the hard drive is having problems. So, yeah. So then I booted my little Ubuntu Linux in order to...

Leo: Oh, you're just warming the cockles of my heart there. You've got your little USB key with Linux on it.

Steve: And it's got a nice persistent file system. So I'm able to install other things on it and sort of build a tool set over time.

Leo: That's not the standard installer disk.

Steve: Correct.

Leo: The live boot CD. You actually made a real Linux system on there. Nice.

Steve: Correct. And in fact, because I'm still sort of a newbie, I stumbled around doing that. And for anyone who's interested in following along, the latest Rufus has...

Leo: Yeah, Rufus is great, yeah.

Steve: It has a slider on it where you're able to slide how much space you want to make persistent store. And it does all the work for you.

Leo: Oh, nice.

Steve: When it sets up a bootable thumb drive. So after trying to manually create a file system like three or four times, and it kind of didn't work or it took too long or something, I stumbled on Rufus, and it's like, yeah. You just slide the thing from the left where it's no persistent storage. You slide it over to - I think I cut it in half. So because it was a 32GB thumb drive, I made it 16GB of persistent store. And it works perfectly.

Leo: I'm going to have to try that.

Steve: It's just an overlay. It's an overlay file system.

Leo: So you actually are using the live CD ISO.

Steve: Yes.

Leo: You're not installing it on the key. Because that's what I would have done is just run the Linux installer. And instead of installing it on a hard drive, install it on the USB key. But you're actually using the live CD with some storage because normally it's a read-only volume.

Steve: Yes. Exactly.

Leo: Oh, interesting. What a good idea.

Steve: It's really cool, and Rufus makes it like full GUI, just click here, thank you very much.

Leo: Rufus.ie, a really good product. Free. Nice. Really does a good job.

Steve: And it's being maintained continually. I don't use it often, and typically it sees that there's a newer version of it and says, hey, you know, you might as well use that one. I go, oh, okay. And you don't even install it. It just - it's an EXE. It's like one of my...

Leo: It's like one of yours. It's probably written in assembler.

Steve: Okay. So actually it's not. It's open source, and I did look at some of its source when I was working on SpinRite's AHCI controller, AHCI driver to see if there was anything I could learn from it. And I ended up - or maybe it was the - I think it was when I was working on the USB boot stuff because Rufus is also a formatter. But it turns out mine is able to fix some thumb drives that Rufus still won't install on.

Leo: Oh, interesting. Ah, that's also good to know.

Steve: Okay. In any event.

Leo: Yes.

Steve: The Smartmontools stuff, it turns out, isn't useful for NVMe. So you need to load an NVMe command that isn't normally part of the Ubuntu set. So I did that, and I was able to look at the NVMe drive. But it still seemed it was having some problems. So open up the laptop, take out the little NVMe. It was a nice Samsung OEM NVMe. I stuck it into an NVMe-to-PCIe adapter and put it in a different machine. Booted that up in Linux, and it could see it. Oh, I also stuck it into an NVMe-to-USB adapter and plugged it into my Windows 7 machine. Up it came, showing the drive looking fine. It showed three partitions. The UEFI boot partition in the big middle was the C drive with BitLocker. So it was BitLocker encrypted. And then at the very end was the WinPE, the recovery partition.

So the drive hadn't died, or at least didn't look like it had a problem. Looked like it was okay. So it's like, oh, okay. So but I couldn't do anything with it because of course it was BitLocker encrypted. And this laptop was Secure Boot, BitLocker encrypted. It apparently came from Lenovo with Windows 10 installed and Secure Boot and BitLocker. So that was a problem because I did have some dialog asking whether he had the backup key, the BitLocker recovery key.

Leo: Yeah, it uses a certificate system for encryption. And if you don't back that up, uh-oh.

Steve: Yes. And because it uses TPM, it was tied to that hardware.

Leo: That's right. That's right.

Steve: So only being in that laptop would allow BitLocker to decrypt. I asked him, do you have paperwork come with it? Because I figured if Lenovo gave him a laptop that had BitLocker already enabled, they would have printed out that - it's a 48-digit BitLocker recovery key, which you need in order to decrypt a BitLocker partition if you don't, you know, if it's not in the native machine that it came from. Anyway, he was absolutely sure he didn't have it. I said, okay, what about OneDrive? Because one thing that happens is, if you log into OneDrive, Windows 10 is supposed to back up the BitLocker recovery key to your OneDrive. He looked. It wasn't there. I said, what about an earlier - because this was now 2018. How about earlier? He checked a different Microsoft account, also no BitLocker keys. So he was absolutely sure he didn't have it anywhere.

Okay. Meanwhile, I ran, to get a sense for the condition of the drive, I then ran the DD command in Linux using /dev/null as the output. So basically it just did a read scan of the entire NVMe. It was a 1TB Samsung NVMe. I set the block size to 60MB because that's a good size. And it read through the entire thing without error. So plugged into...

Leo: Ooh, that's good.

Steve: ...that other machine, booted Linux. Yup, absolutely good. Booted Linux, worked fine, without error. So Linux is saying, just from a simple read scan...

Leo: Hardware's good.

Steve: Hardware's good. There is no problem with this drive. And plugged into the USB adapter, my Windows 7 machine said, yeah, I see three partitions. The middle one's BitLocker. That's going to be a problem unless you've got the key. But we're okay. I mean, so the boot sector and all that stuff is there. So I think, okay, plug it back into the laptop. Doesn't work. Go to the diagnostics. Oh, he did not have the latest BIOS. I updated his BIOS on the laptop. It gave me much better diagnostics than it was originally shipped with, more detailed and many more tests. And again, would not pass a retest. It failed failed on the reads. And then there were some other tests of just the NVMe itself, and those it passed. But it would not read. So then I thought, okay. So maybe the NVMe slot has died. So I took a different NVMe drive, plugged it in, works perfectly. So this drive works fine elsewhere. Another NVMe drive in the laptop, it works fine. But this drive and the laptop won't work together.

Leo: I have a feeling I know where the end game is on this one. Go ahead.

Steve: So here's the problem. He needs the data. It's BitLocker-encrypted. He's sure that the key, the recovery key, doesn't exist anywhere.

Leo: You've got to get this drive working. There's just no way around it.

Steve: Got to get the drive working. And the only instance of the key known to exist is in the Trusted Platform Module on the motherboard.

Leo: You have to get it working on that P51. No way around it.

Steve: Exactly. Exactly.

Leo: Well, I know what I would do.

Steve: And so the only thing I could think was that, like, how do you explain this weird drive behavior is that something bad happened, like age, just an age-related problem where the signal levels of that, that is being generated...

Leo: Maybe just one cell. Just one cell.

Steve: Well, or a signal line is a little bit out of spec.

Leo: Oh, on the whole drive.

Steve: Such that this drive, yeah, like this drive and this particular interface chip in the laptop are no longer talking to each other.

Leo: Interesting.

Steve: A different drive, no problem. And it in a different machine, no problem. But not those two.

Leo: Yeah, because that WHEA error is a Windows hardware error. So that's why you would think, well, there's a hardware problem here.

Steve: Right.

Leo: That's interesting.

Steve: Right. And so what was happening was the drive was just going offline. It was not passing its diagnostics.

Leo: Yes, can't talk to me, yeah.

Steve: So I told him, I said, look. Talk to Lenovo. If it came from them, maybe they have a record of the BitLocker recovery key. You'll have to prove that it's your ownership and so forth in order to get them to do that. So he spent a lot of time on the phone. They instructed him to take it to one of their partners, one of their, like, corporate partner companies, not really even a consumer company. And so I was getting ready to pack it up. And I said, you know, there's one more thing I want to try. It probably won't work. But I've got to rule it out. This other little piece of hardware is coming tomorrow. I'm going to see if it does the job. Because it had a Thunderbolt 3 port. And I remembered that Thunderbolt - and we talked about this on the podcast.

Leo: DMA access, yes.

Steve: Yes. Thunderbolt is actually the PCIe bus. It is a hybrid of the PCIe bus and display port. And of course what's happened with PCIe is the reason we went from PCI to PCIe is that interconnect became too expensive. As more stuff was happening, it no longer made sense to have - you couldn't have a 64-bit address bus on a 64-bit system, and a 64-bit data bus. That was just - that would be 128 bits of, like, individual wires. So we went from a parallel bus architecture to a serial bus architecture, where now, instead of putting a whole bunch of things on a bus, you flip it over, and you make the bus be serial, and you then have a one-to-one relationship between the processor chip and each individual device in the system. Well, the Thunderbolt port is one of those. Amazon delivered a Thunderbolt 3 to NVMe case. And the first one I ordered was wrong because it worked on USB. So the second one, I had to make sure, and they did say this will not work on USB. It looks like USB3. It's not USB3. It's Thunderbolt.

Leo: Good. Good. Good idea.

Steve: So that's like, yes, yes.

Leo: That's what I want.

Steve: That's the one I'm trying. I put it in the case. I plugged it in, and it booted.

Leo: So your diagnosis was something wrong with that NVMe slot, that M.2 slot, timing on the pins. So putting in an external connected NVM3 reader on the PCIe bus...

Steve: Because that's Thunderbolt.

Leo: Thunderbolt. TPM would still say, oh, I recognize that?

Steve: It did. It recognized the drive.

Leo: It decrypted it?

Steve: Yup.

Leo: And you were able to read it.

Steve: Yes.

Leo: Unbelievable. What a story. See, I thought you were going to end with SpinRite, you were going to SpinRite the thing. But it really was a hardware error.

Steve: It was a hardware error. The drive was fine. They just wouldn't talk to each other on that connector.

Leo: That is bizarre.

Steve: Yeah. And I said to him, I said, you know, frankly...

Leo: You're lucky.

Steve: Yeah.

Leo: You're lucky I exist, is what he would say. No one else is going to solve that, no. That's a great story, Steve. Boy, he is really lucky he knows you because...

Steve: Well, and of course the first thing I did, I made an image of it because, I mean, it was like this data was so crucial.

Leo: Yeah, yeah. Yeah, yeah.

Steve: I made an image of it. He was desperate for some of it, so I installed the remote utilities and let him access it remotely to pull the files off that he needed.

Leo: Wow.

Steve: Then after I had an image safely, I removed BitLocker, just to get rid of it because, thank you anyway, this is too important. And I ended up shipping - I called it "the sidecar." I actually, because it was running a little warm, I glued some...

Leo: Heat sinks. Nice copper heat sinks on there.

Steve: Some nice copper heat sinks to the top of the case.

Leo: You're such a geek. Wow.

Steve: Because he was going to be using it from now on.

Leo: Oh, no. Really. I guess he has to; right? He can't fix that laptop.

Steve: Well, maybe he could put a different NVMe drive in. But I wouldn't trust it. I would never trust it again. So he has the laptop. And he's also not financially constrained.

Leo: Get a new one, yeah.

Steve: I mean, he could instantly - he'll get a nice Carbon X1 or who knows what.

Leo: Lenovo should fix that, though. The problem is proving to them that your obviously correct diagnosis is what's going on. That's just amazing.

Steve: At this point it's old, and it's time. But so anyway, he has it back. It's got its cute little - basically the drive has been externalized. So, I mean, and boy is it fast. And that's another lesson.

Leo: Sure, it's 40Gb.

Steve: I mean, it's running at the full speed of this drive.

Leo: Yeah. I love Thunderbolt. That's amazing.

Steve: Yeah. So what I told him is, use it like this as long as you want to. You'll have to keep them together wherever they go. Eventually...

Leo: You're going to look a little weird.

Steve: Eventually, when you do replace, yeah, replace the laptop. And then what's cool is you plug this into another laptop that also has a Thunderbolt 3 interface, and you'll just be able to copy the stuff over to migrate yourself to a new laptop. So anyway, a cool story, which I thought you and our...

Leo: Really wonderful story. And kudos to you. He's lucky because basically you're a hard drive wizard. I would never have thought of that being, oh, there's a timing issue on the pinout. I mean, that's remarkable. But you did all - had you already ordered that M.2 external? Or you ordered it because you thought this might be the solution?

Steve: I ordered it because.

Leo: Yes, okay.

Steve: I just thought, you know, Thunderbolt is - we've got Thunderbolt, and it's a PCIe. Maybe it'll fool the laptop into thinking, oh, it's still plugged in.

Leo: Thinks it's an internal drive, yeah.

Steve: Yup.

Leo: So it booted to it and everything.

Steve: Booted right up. Didn't even complain. I mean when I saw the little thing spinning, I thought, okay, well, here comes the blue screen. And instead I got the blurry logon screen. It's like [gasp], you know.

Leo: It worked.

Steve: And I, I mean, and he was actually in tears. He got choked up on the phone with me. He was in tears with Lorrie.

Leo: Wow. You saved his life.

Steve: Because, I mean, it was his Ph.D. thesis was there. A whole bunch of irreplaceable biomedical data was there.

Leo: He's very, very lucky that you were the guy he brought it to. No one would have gotten that. That's wild. That's a great story. Bravo. Golf clap. Would you like to take a break?

Steve: I'm going to do that, and then the TLS Collision Attacks.

Leo: That was exhausting. I'm exhausted just listening. What a story. I mean, that is really incredible. I'm going to have to tell that on The Tech Guy. That is just an incredible story. Thing is, when I do, because I have to answer questions like this on the radio frequently, is I store away the knowledge that that's even possible. I would not have thought that that pinout could somehow, some little issue with the pinout on the motherboard and that stock M.2 drive made it unreadable over time...

Steve: They just stopped liking each other.

Leo: That's a bizarre error.

Steve: Yeah.

Leo: But now it's in my mind, you know, if that happens again, I'll keep that in mind. That is wild.

Steve: I guess the takeaway for our listeners...

Leo: Back up.

Steve: Make sure you have BitLocker, that BitLocker recovery key. Check right now. And I will absolutely be setting him up with Sync. He was like selectively using Dropbox and OneDrive, but not ubiquitously. And for what it's worth I'm...

Leo: Would you give him Syncthing, or Sync.com? What do you think you'd use? Because he's got HIPAA requirements. I mean, if there's X-rays on there, things like that.

Steve: True. True.

Leo: He's got a requirement. That's probably why he - I bet you he turned on BitLocker. I doubt Lenovo ships with BitLocker on.

Steve: That would be interesting to know. He says it's not something he ever did. But he's also...

Leo: That was years ago. He might not remember.

Steve: Yeah, it was a long time ago. I wanted to remind our listeners that I'm still loving Sync. I use Sync, and I use Syncthing separately. Grc.sc/sync, S-Y-N-C. That is a referral link which will get you an extra gig. Rather than five, you get six if you use that link, grc.sc/sync, S-Y-N-C, if you want to play with it. It perfectly keeps my multiple locations synchronized. The reason I like it is that it's also in the cloud, and it does versioning, like infinite versioning.

Leo: And it's TNO.

Steve: Yes, it is.

Leo: Trust No One. End-to-end encrypted.

Steve: It is Trust No One, encrypted locally. They can't get to it. HIPAA-compliant. And you're able to generate links for things you want to share, which allow other people to have the decryption occur at their end on the fly. So, very cool.

Leo: I'm signing up right now.

Steve: And Leo, before I forget about it, I just got a note from a Twitter follower who told me that Syncthing, the different peer-to-peer syncing that you and I like, just added encrypted folders. So that means you're able to sync to a machine that is untrusted, if you ever have a need to do that. You're able to assign a long password to a folder. And when it syncs remotely, it will be encrypted before it leaves under that key and then decrypted as it comes back. But everything stored in that synchronized folder will be encrypted, which is not a feature that Syncthing has had until now. So it just got it.

Leo: That's good to know.

Steve: Yeah.

Leo: Yeah, I use Syncthing everywhere, all the time. That's really my, well, it's not my primary backup, but it's one of my several backup systems because basically I have, just like you, you have multiple machines, multiple locales, and you want to have your source code directory synchronized everywhere. I have my source code, my sync files, my dot files, my documents, my pictures, and my audio, all synced to all systems. And it's a great way to do it. It's peer-to-peer.

Steve: And if you have a Linux-based NAS, as I do, because I've got Drobos in each location, then I've got Syncthing running natively on the Drobos, and they're just pulling stuff. They sync each other, and I sync to them.

Leo: So that is a pretty good backup strategy. Yeah, I do that on my Synology. They have a community-based version of Syncthing available.

Steve: Nice.

Leo: Looks like I already signed up for Sync.com. I didn't even notice.

Steve: It is, it is...

Leo: It's a terrible name, by the way. They really - they should call it AvocadoSync or something just so that you would - because you can't google it.

Steve: Well, and think how much they must have paid for that domain, to have Sync.com. However, the reason it would be right for John is that I don't think he - he's sort of a nomad. He's not really a deep computer geek. Syncthing is a little - you need to kind of know your way around, you know, things.

Leo: It's geeky, for sure.

Steve: And it just, yeah, and it just does it to the cloud. And so his entire documents folder would just be synced using Sync, and he just would never have this problem again.

Leo: Oh, I remember why I don't use Sync.com. No Linux client.

Steve: Ah.

Leo: Get them to work on that. Now, ladies and gentlemen, it's time for our titular topic of the day.

Steve: So I titled this TLS Confusion Attacks for reasons that'll become clear. Some very interesting and quite depressing research will be presented at the forthcoming 30th USENIX Security Symposium and this summer's Black Hat USA in Las Vegas about six weeks from now, at the end of next month. And even though it's not entirely new, it teaches us some interesting lessons about unintended edge case consequences.

Now, the researchers apparently struggled to come up with a cute name for this.

Leo: Uh-oh.

Steve: Since the problem - yeah. Since the problem they were exploring was the mischief that could be created by deliberately confusing the application layer within TLS-authenticated and protected connections, they had "Application Layer Protocol Confusion," ALPC. Which is, you know, not really anything. But if you add a pair of judiciously...

Leo: Oh, god.

Steve: Yeah, add a pair of judiciously placed gratuitous A's into that, you can force this to become A-L-P-A-C-A.

Leo: ALPACA. At least it's memorable.

Steve: And that explains why I chose to name this podcast "TLS Confusion Attacks."

Leo: But alpacas are so cute, Steve.

Steve: Well, regardless of their name, these attacks, they're a truly interesting instance of an unintended consequence edge case. Or to use the official term, "Oops." Here's what happened. In the beginning, we had the UDP and TCP IP protocols. UDP was mostly used for DNS. But over the TCP protocol we ran TELNET and FTP and SMTP and POP and IMAP and HTTP, you know, and others. Those were the so-called Application Layer protocols running on top or inside of the TCP Transport Layer. And back then the world was simple. But it was not secure. Passive eavesdroppers could listen in on passing Internet packets to see what everyone was up to, so there was no assurance of privacy. And active man-in-the-middle attackers could redirect traffic to other destinations without detection, so there was no authentication.

But we're clever. So we grafted on SSL, which over the years evolved into TLS. The idea was that our existing plaintext application protocols would be given different ports when their TCP connections should be authenticated and encrypted. So HTTP over port 80 became HTTPS over port 443. FTP's port 21 became FTPS over port 990. And SMTP's port 25 became port 587 whenever TLS would be used to secure the SMTP protocol. When using these alternate secured ports, immediately after establishing that famous three-way TCP handshake, a sort of shim would be introduced in between the TCP transport layer and the application protocol layer. Before any application layer traffic would be allowed to flow, the two endpoints would need to exchange another set of packets to cryptographically establish the identity of one or both of the endpoints, thus providing endpoint authentication, and agree upon an encryption key that they would henceforth use to encrypt all subsequent communications, thus providing communications privacy.

Problem solved? Pretty much, but not entirely. The problem was, and has always been, that the SSL/TLS protocol is bound to the TCP connection, but not to the underlying application layer protocol whose traffic it is carrying. In other words, the TLS protocol itself is unaware of the IP and port to which it is connecting. The TCP protocol knows about that, but not TLS. It's naive to that. So it does the same thing if the connection is to HTTPS as if it was to secure SMTP. TLS doesn't care at all. But the underlying application protocol does care which port it's connected to because that determines which service it's talking to. And therein lies the edge case. Since the TLS is not bound to the connection's IP and port, they can be changed, and TLS won't care. It won't even notice. But since the underlying protocol does care, that creates a problem. And this opens what appeared to be a safe and secure system to what the researchers term "cross-protocol attacks."

Here's what the abstract of their paper explains. It says: "TLS is widely used to add confidentiality, authenticity, and integrity to application layer protocols such as HTTP, SMTP, IMAP, POP3, and FTP. However, TLS does not bind a TCP connection to the intended application layer protocol. This allows a man-in-the-middle attacker to redirect TLS traffic to a different TLS service endpoint on another IP address and/or port. For example," they write, "if subdomains share a wildcard certificate, an attacker can redirect traffic from one subdomain to another, resulting in a valid TLS session. This breaks the authentication of TLS, and cross-protocol attacks may be possible where the behavior of one service may compromise the security of the other at the application layer."

Okay, so let me pause for a second. What they're saying is that if, for example, GitHub were to use the same TLS certificate for HTTPS as for their Secure SMTP, it might be possible to redirect an incoming user's web browser connection, which is intended for GitHub's HTTPS port 443, to their secure SMTP port 587. And because the same TLS certificate will be used to negotiate the connection's TCP security either way, it may be possible to take advantage of this protocol level confusion.

Okay. So continuing with their abstract. They say: "In this paper, we investigate cross-protocol attacks on TLS in general and conduct a systematic case study on web servers, redirecting HTTPS requests from a victim's web browser to Secure SMTP, IMAP, POP3, and FTP servers. We show that, in realistic scenarios, the attacker can extract session cookies and other private user data or execute arbitrary JavaScript in the context of the vulnerable web server" - in other words inject script - "therefore," they write, "bypassing TLS and web application security."

They finish: "We evaluate the real-world attack surface of web browsers and widely deployed email and FTP servers in lab environments and with Internet-wide scans. We find that 1.4 million web servers are generally vulnerable to cross-protocol attack, i.e., TLS application data confusion is possible. Of these, 114,000 web servers" - so just shy of one out of every 10 - "can be attacked using an exploitable application server. Finally, we discuss the effectiveness of TLS extensions such as Application Layer Protocol Negotiation (ALPN) and Server Name Indication (SNI) in mitigating these and other cross-protocol attacks."

Okay. So the main takeaway here is, I think, that not only is security difficult, it's even significantly more difficult than we know. In fact, it might be even more difficult than we can know.

Leo: Than we can. That's the problem.

Steve: Yes.

Leo: Yes. Your giveaway was when you said, "But we're clever." Yeah, that's the giveaway. Don't get clever. Yeah.

Steve: Yeah, never get clever. So to put this into context, okay, this is not the end of the world. The attack requires sophistication and the ability to establish a man-in-the-middle position, the ability to actively manipulate the traffic coming and going from a victim in real-time. But it's certainly within the reach of state-level attackers, and it does mean that users are not really obtaining the security that we think we've always had. These attacks also require a very specific set of circumstances.

The researchers described what they uncovered. They said: "In practice, cross-protocol attacks are sensitive to many requirements, such as certificate compatibility, ability to upload, download, or reflect data, and application tolerance towards syntax errors caused by mixing two protocols in one channel." Yeah. So, I mean, it's an edge case. They said: "In our case study of cross-protocol attacks on HTTPS, using Secure SMTP, IMAP, POP3, and FTP application servers, we address these concerns in three evaluations.

"First, we identified 25 popular SMTP, IMAP, POP3, and FTP implementations and evaluated their suitability for cross-protocol attacks on HTTPS in a series of lab experiments. We found that 13 of the 25 are exploitable with at least one attack method. We also implemented a full proof of concept that demonstrates all three attack methods on a well-secured web server, using exploitable SMTP, IMAP, POP3, and FTP application servers.

"Second, we evaluated seven browsers for their error tolerance. We find that Internet Explorer and Edge Legacy still perform content sniffing, and thus are vulnerable to all presented attacks." Okay, so by content sniffing they're meaning, and I couldn't believe this, IE and Edge Legacy look at the protocol text to determine what the protocol is. That is, does this look like HTTP? Or does this look like FTP?

Leo: They just call it "deep packet inspection" and then...

Steve: You're kidding me. That's - you're kidding me. Anyway, they said, "...while all other browsers allow at least FTP upload and download attacks. And, third, in an Internet-wide scan, we collected X.509 certs" - which are the standard web certs - "served by SMTP, IMAP, POP3, and FTP servers. We analyzed how many of these are likely to be trusted by major web browsers." In other words, are they using the same certs? GRC does. It's easy. And they're good, so why not?

"For each certificate, we extracted the hostnames in the Common Name field and the SAN."

Leo: Storage Area Network.

Steve: Server, no, Alternate Names, yeah, Server Alternate Names, yeah, extension.

Leo: Oh, okay. We have unfortunately acronym overload here.

Steve: We do. Well, because when you're only going to use three letters, you're in trouble. "And checked if there exists" - ALPACA, that's pretty safe. "And checked if there exists a web server on these hosts." They said: "We found," and here's where this number comes - "1.4 million web servers that are compatible with at least one trusted application server certificate," meaning that the same certificate is crossing between applications, web and mail, web and FTP, whatever. They said: "... making them vulnerable to cross-protocol attacks. Of these, 119,000 web servers are compatible with an application server that is exploitable in our lab settings."

Okay. So this is bad. What can be done? It's a decidedly mixed blessing. Because while it's unlikely to occur, given the proper conditions, it can. And when it's exploited, it can result in logged-on session hijacking because you're able to read the browser cookie, the session cookie which is supposed to be unreadable, otherwise we're back in the, what was that, it was Firesheep. Remember those innocent days of yore, Leo?

Leo: Yes, yes, yes.

Steve: You could go to Starbucks and just hop onto other people's login sessions. We were so innocent back then.

Leo: Oh, yeah.

Steve: And, okay. And if you can inject malicious JavaScript, then you're running that malicious JavaScript in the context of the fully trusted web server, which means you can do anything. The fact that it's unlikely to occur, unfortunately, reduces the pressure to make any changes. Right? There's not like Heartbleed or, for example, in Dan Kaminsky's famous fix for DNS, which only required that the DNS servers be fixed. The changes that need to be made here have to occur at both ends of the TLS connections. So all the clients and all of the servers.

There is currently a well-defined solution for this. It's called ALPN. That stands for Application Layer Protocol Negotiation. It does exactly what its name sounds like, and what we need. It's an optional extension for TLS connection setup, which allows the endpoints to explicitly agree upon the protocol that the connection will be carrying. That's the thing that's missing from TLS now is that nothing tells TLS what it's going to be carrying. But ALPN, an optional extension, does. And it turns out that it does this without needing any additional packet roundtrips or anything. It's just added to the existing TLS as one of the extensions in the already well-defined packets.

And since support for the HTTP/2 protocol needs to be able to smoothly upgrade an HTTP/1.1 connection to HTTP/2, in other words, that's exactly what we need here. We need the HTTP/2 server to be able to negotiate an HTTP/2 connection with the other endpoint, only if both endpoints agree. This is now being done via ALPN to the degree that HTTP/2 has been deployed. So ALPN support is already emerging.

However, protection from web-based protocol confusion attacks, that is, these kinds of attacks, requires that all TLS connection-accepting services, which could be unwitting participants in protocol confusion, meaning all FTP servers, SMTP, POP3, IMAP, and so forth, also support ALPN and flatly reject and terminate any non-ALPN enhanced connections. And I can't imagine a world in which that's ever going to happen. I just, you know, it's hard enough to get moved from IPv4 to IPv6. This is like, well, probably no one's going to get abused by this, and it's a theoretical attack, and really we have to replace everything, and we don't get any protection until we mandate that everybody do it, and anybody that doesn't gets hung up on? No. I don't recognize that world. That's not the one we're in. So we can imagine that future non-web services might begin to incorporate ALPN awareness, but that's entirely different from, as I said, refusing connections from remote clients that are not offering ALPN-enhanced TLS. The outlook is not good.

They conclude their own paper, saying: "We demonstrated that the lack of strong authentication of service endpoints in TLS can be abused by attackers to perform powerful cross-protocol attacks with unforeseeable consequences." And now this is where Bruce Schneier's "Attacks never get worse, they only ever get better" is ringing in our ears. We may be revisiting some new consequence that has been literally, these guys said, "unforeseeable." So something unforeseen may emerge from this.

They said: "Our Internet-wide scans showed that it is common" - I'm guilty - "for administrators to deploy compatible certificates across multiple services" - why not, seemed like a good idea - "possibly without consideration to cross-protocol attacks." Yeah, I didn't think about that. "We also showed that cross-protocol attacks are practical, although the impact is limited and difficult to assess from lab experiments alone. In the real world, cross-protocol attacks will always be situational and target individual users or groups. However, it is also clear that existing countermeasures are ineffective because they do not address all possible attack scenarios. We have identified one countermeasure that is far superior to others: the pervasive use of the ALPN extension to TLS by both client and server. Luckily, ALPN is easy to deploy with the next software update without affecting legacy clients or servers."

So, yeah, that's true. You could put it out there, get it in place, and wait for everyone to have it, and then flip the switch to make it mandatory. But again, it is only - it's necessary to make it mandatory, to actively refuse any non-ALPN-enhanced connections. And I don't know if that's going to happen in our lifetime. So I wanted to put this on everyone's radar because we may be coming back. There may be ALPACA 2.

Leo: Wow.

Steve: Yeah. Unforeseen. Like nobody ever said, I mean, it seemed like wrapping our TCP connections in TLS...

Leo: Seemed like a good idea.

Steve: What could go wrong?

Leo: Right.

Steve: Well, as I said, the official term is "oops."

Leo: It's subtle, though. I mean, I could see how you might miss it.

Steve: Yeah, yeah, yeah.

Leo: Wow. But that's the problem is we've gotten to the point now where any flaws...

Steve: The stuff is so confusing. It's so complicated.

Leo: Yeah. That's what it really is. It's these multiple interactions. It's the complexity of the system is almost chaotic at this point, in which case you can't - if it's unpredictable, you can't do it right. Interesting.

Steve: Yeah, I agree. I agree. I think that's a good analogy. I think we are approaching chaos. Formal, mathematical chaos.

Leo: That is not - you're right. Now I am depressed. I was all happy because you solved the most impossible problem I've ever heard. And now you just ruined it. Steve Gibson, he's the guy. Man, is he the man. I'll tell you what, go to GRC.com. He's got lots of great stuff there, free stuff, of course his bread and butter, SpinRite, the world's best hard drive, I'm sorry, mass storage recovery and maintenance utility. I was sure that other story was going to end with a spin drive. But you threw me a curve. You threw me a curve there. There's nothing wrong with the drive. Very clever. This is the guy you want working on your hard drive, trust me. Did I say "spin drive"? I meant SpinRite. Apparently I said "spin drive."

Steve: We knew, we all knew. Yeah, the most downloaded thing at GRC until 6.1 is available is the DNS Benchmark.

Leo: Yeah, I'm not surprised.

Steve: 3,000 downloads a day, every single day, day in and day out.

Leo: It's a really good way to - because people I think are starting to realize they don't want to use their ISP's DNS servers. But they don't want to use a slow one, either. And so they want to - so they're thinking about Cloudflare or Quad9's or NextDNS. There are a lot of them now.

Steve: And how would you know?

Leo: How would you know which is the best? So test it. That's a great free download. Steve's very generous. A lot of great stuff there. GRC.com. He's also on Twitter at @SGgrc. That's where you can DM him, slip into his DMs. They're open. You can also message him at GRC.com/feedback; right? Yeah, GRC.com/feedback. His forums are great, too, forums.grc.com. Is that right?

Steve: That's correct, yeah.

Leo: Steve, have a wonderful week. We'll see you next time on Security Now!.

Steve: Thanks, buddy. Bye.