SECURITY NOW!

Transcript of Episode #707

Tesla, Pwned

Description: This week on Security Now! we have the return of "Clippy," Microsoft's much-loathed dancing paperclip; operation "ShadowHammer," which reports say compromised ASUS (but did it?); the ransomware attack on Norsk Hydro aluminum; the surprise renaming of Windows Defender; a severe bug revealed in the most popular PDF-generating PHP library; an early look at Microsoft's forthcoming Chromium-based web browser; hope for preventing caller ID spoofing; a needed update for users of PuTTY; Mozilla's decision to conditionally rely upon Windows' root store; Microsoft to offer virtual Windows 7 and 10 desktops through Azure; details of the Windows 7 End of Life warning dialog; then a bit of Sci-Fi, SQRL and SpinRite news, followed by our look at the results of the much anticipated Mid-March Vancouver Pwn2Own competition - one of the results of which our episode title gives away!

High quality (64 kbps) mp3 audio file URL: http://media.GRC.com/sn/SN-707.mp3
Quarter size (16 kbps) mp3 audio file URL: http://media.GRC.com/sn/sn-707-lq.mp3

SHOW TEASE: It's time for Security Now!. Birthday boy Steve Gibson is here. He is - we're going to call him "Commodore 64" today because it's his 64th birthday, and he's got a lot to talk about including the malware, the big Norwegian aluminum company. You won't believe the sign they put in the door. And a play-by-play of Pwn2Own, all three days, coming up on Security Now!.

Leo Laporte: This is Security Now! with Steve Gibson, Episode 707, recorded Tuesday, March 26th, 2019: Tesla, Pwned.

It's time for Security Now!, the show where we cover your security and privacy online with the man and the plan, Panama, Steve Gibson. Actually, ever wear a Panama hat? You'd look good in a...

Steve Gibson: No.

Leo: You'd look great in a Panama hat.

Steve: Mine is sort of the French beret. I'm sure you remember me in that.

Leo: You look good in a beret. I do remember that.

Steve: I used to wear that all the time.

Leo: That was your hat.

Steve: I just kind of fell out of the habit somehow.

Leo: It was no brim. You need one with a brim.

Steve: Well, no. That's true, it does not keep the rain out of your eyes nearly as well as a big Panama hat.

Leo: So let me add security, privacy, and hat couture.

Steve: That's true.

Leo: What's coming up this week?

Steve: So we have, for Episode 707, the title of the podcast is "Tesla, Pwned."

Leo: Oh, yeah. I saw that.

Steve: But it wasn't a bad pwnage. It was about as mild a pwnage as it could be. On the other hand, they did drive off with a Tesla Model 3.

Leo: Yeah, yeah.

Steve: So they're pretty happy. This, of course, was last week's three-day Pwn2Own. And it's called Pwn2Own because, if you pwn it, you own it. And they pwned it, and they're now driving it.

Leo: And a little bit of cash also. This is Fluoroacetate. They're at it again.

Steve: They are. And what impressed me the most, I have to say, is that the prize money that they won, relative to what Zerodium would have paid, because one of the things they did would have qualified for the half a million dollar recently increased - and we talked about this a couple weeks ago - zero-day in VMware, where they were able to execute code on the host outside of the VM after just browsing to a web page, which is just like, oh, you know, that's like - that's the golden goose from Zerodium's standpoint. But they didn't sell it to Zerodium.

Leo: Good.

Steve: I'm very impressed.

Leo: It's a proof-of-concept for Pwn2Own, really, that shows it's a benefit.

Steve: Yeah. So the problem is normally our show notes are 14 pages, and we run out of time. We have 20 pages today. I mean, there is a lot to talk about.

Leo: I'll shut up.

Steve: We've got the return - and, okay. Now, arguably there's a little bit of padding here because I've had some fun, too. I could not pass up the fact that we have the return of Clippy, Microsoft's much-loathed dancing paperclip, which actually is our Picture of the Week that came to me when Hawaii had their problem with that false alert. We also have Operation ShadowHammer, which reports say compromised ASUS. But I may be the only one in the industry who's a little skeptical. We'll talk about that.

We have the ransomware attack that you brought up, it was occurring during last week's podcast against Norsk Hydro aluminum, which took down across 40 different plants located globally. It was a ransomware attack. We have the surprise renaming of Windows Defender for a purpose, a severe bug revealed in a most popular PDF-generating PHP library, an early look at Microsoft's forthcoming Chromium-based Edge browser, hope for preventing caller ID spoofing, a need to update for users of PuTTY, which is a very popular freeware SSH client.

We've got Mozilla's decision to conditionally rely on Windows root store rather than their own. Microsoft will be offering virtual Windows 7 and 10 desktops through Azure and what that means. Also I heard you and Mary Jo and Paul talking about this last week, sort of some wondering about what the Windows 7 end of life, end of service life, end of update life warning dialog would look like. We now know.

I've got a tiny bit of sci-fi, some SQRL and SpinRite news, and then we're going to take a look at the much-anticipated mid-March Vancouver Pwn2Own competition, three days. And of course the title of our podcast gives away what happened during the third automotive pwning day. But I think another great podcast for our listeners.

Leo: And let me know if you need the sportscaster voice because I'm ready.

Steve: Howard Cosell.

Leo: I could do the play-by-play if you should need it.

Steve: Okay.

Leo: Clippy's back. I missed little Clippy.

Steve: Well, okay. I was going to say you're alone in that, but I don't think you are.

Leo: He was cute. I just don't want him helping me with grocery lists, you know.

Steve: Yes. Anyway, the Picture of the Week was sent to me back when Hawaii had that bogus ballistic missile alert. And anyway, so this is the Clippy from yesteryear. It was introduced with Office 97, I think it was, and it was taken out of commission with Office XP.

Leo: Whew.

Steve: Yeah. Anyway, much maligned and - anyway. So anyway, our Picture of the Week is Clippy saying, "It looks like you're sending out a ballistic missile alert."

Leo: Would you like some help?

Steve: Would you like some help? And that's what it used to do. It used to be, like, watching what you were doing.

Leo: Yeah. Annoying.

Steve: And it would jump in and, oh, my god, so annoying. And, I mean, it was just, I don't know, I guess it wasn't right for the time. But it turns out Clippy is coming back. Well, at least for some places. In a blog posting made on April 11th, 2001 - okay, so back 17 years ago, no, 18 years ago nearly. Microsoft titled it "Farewell Clippy: What's Happening to the Infamous Office Assistant." And their title said "in Office XP." But what they meant was "with Office XP," that is to say, it was introduced in Office 97, and they are saying farewell to it.

So what they wrote was: "Whether you love him or hate him" - and actually the vote was very heavily weighted toward the latter - "say farewell to Clippy automatically popping up on your screen. Clippy is a little paperclip with the soulful eyes and the Groucho eyebrows," they wrote, "the electronic ham who politely offers hints for using Microsoft Office software."

Okay, and I love the way they spun this. They said: "But after four years onscreen, Clippy will lose his starring role when Microsoft Office XP debuts on May 1st. Clippy, the Office Assistant introduced in Office 97, has been demoted in Office XP." And I did enjoy this pun: "The wiry little assistant..."

Leo: Because he's made out of a paper clip, yeah.

Steve: Uh-huh, "... is turned off by default" - I didn't even know it was still there, so I'm glad it was gone.

Leo: You could turn it on? Who would ever want to do that?

Steve: Exactly, "...turned off by default in Office XP. But diehard supporters can turn Clippy back on if they miss him."

Leo: Aw.

Steve: And here's the spin. Lisa Gurry, a Microsoft product manager, explained: "Office XP is so easy to use..."

Leo: Oh, please. The spin machine.

Steve: I know, "...that Clippy is no longer necessary..."

Leo: No, it's easy.

STEVE : "...or useful." That's right. We've finally figured out how to make our UI work, so we don't need a paperclip to come springing out and helping you. Anyway, she said: "With new features like smart tags" - whatever those are - "and task panes" - whatever those are - "Office XP enables people to get more out of the product than ever before."

Leo: Thank god.

Steve: Oh, whew. "These new simplicity and ease-of-use improvements really make Clippy obsolete," she said. And then, finally: "He's quite down in the dumps," Gurry joked. "He has even started his own campaign to try to get his old job back or find a new one."

Now, surprisingly, that was then. A report in USA Today, well, USA Today, not today, back in 2002 stated that Microsoft banked on its customers' contempt - this is actually - USA Today said this back then. "Microsoft banked on its customers' contempt of Clippy to promote Office XP."

Leo: There's a selling point. No more Clippy.

Steve: That's right. No more of that - anyway.

Leo: That's funny.

Steve: "On Thursday," they wrote, "On Thursday, Microsoft is scheduled to unveil the last installment in a nontraditional advertising campaign that aims to sell the newest version of Office, called XP, by encouraging customers' hatred of Clippy." Unbelievable.

So here we are now, today, finally today, 18 years later, and wouldn't you know it, Clippy's lobbying to return to the limelight appears to be paying off. Clippy is about to make a not-long-awaited comeback for Microsoft's Teams app. The effort is open source and on GitHub, so the animations are all publicly available. And I have to confess, Leo, that Clippy has become such a meme from the past that, had I him available to embellish the occasional iMessage on my iOS device, that might be kind of funny. Fun. I mean, and I put down here at the bottom a snap of one of them. We have the beer-drinking Clippy because it's 2019 now, so we can do that.

Leo: They let him drink beer?

Steve: And he's also got a coffee mug. There's one with a coffee mug. There's one where he's holding like a Starbucks-style paper with the little heat guard slip-on dealy-do. Anyway, there's a bunch of them. They're animated, and I'm sure that someone is going to grab them off of GitHub and sprinkle them around. So I've never been much of a big fan of the emojis and things, but if we had this little bank of animated paperclips, almost because it's a dated meme I think it would be kind of fun. So I'll bet it happens.

Leo: Nice.

Steve: Okay. So now here's - this is really odd. It's called Operation ShadowHammer. And first I'm going to share Kaspersky's post about the incident. Then I'll explain what puzzles me so much about this. So Kaspersky wrote: "Earlier today" - and this just happened - "Motherboard published a story by Kim Zetter on Operation ShadowHammer." And I should mention that Motherboard story is based on Kaspersky's research, so they're sort of self-referential here. By Kim Zetter on Operation ShadowHammer, "a newly discovered supply chain attack that leveraged ASUS Live Update software."

And Motherboard's headline read: "Hackers Hijacked ASUS Software Updates to Install Backdoors on Thousands of Computers." And their subtitle says: "The Taiwan-based tech giant ASUS is believed to have pushed the malware to hundreds of thousands of customers through its trusted automatic software update tool after attackers compromised the company's server and used it to push the malware to machines." And for anyone who's interested, I have the link to the whole Motherboard article in the show notes.

So Kaspersky says: "While the investigation is still in progress, and full results and technical paper will be published during SAS 2019 conference in Singapore" - which I think is only like 12 days away, so a week and a half we'll know more. They said: "We would like to share some important details" - this is Kaspersky speaking - "about the attack. In January 2019" - so two months ago, beginning of this year, they write - "we discovered a sophisticated supply chain attack involving the ASUS Live Update Utility. The attack took place" - okay, and not really the utility. That's the thing that reaches back to ASUS, right, to check for any updates.

Anyway, "The attack took place between" - get this - "June and November." Now, not meaning a single event of attack, but meaning for five months this was ongoing. So the attack took place for the span between June and November 2018. "And according to our telemetry, it affected a large number of users. ASUS Live Update," they write, "is a utility that is pre-installed on most ASUS computers and is used to automatically update certain components such as BIOS, UEFI, drivers, and applications." And of course we've talked about this a lot. Lenovo has this. Famously Microsoft invented this. I remember how much at the time it was like, wait a minute, you're going to update my computer without my involvement? That was a thing once. Now it's just like, okay, please bring it on.

"According to Gartner, ASUS," writes Kaspersky, "is the world's fifth largest PC vendor by 2017 unit sales. This makes it an extremely attractive target for APT [Advanced Persistent Threat] groups that might want to take advantage of their user base." Okay, but let me tell you why this doesn't track. We'll get there. Kaspersky says: "Based on our statistics, over 57,000 Kaspersky users have downloaded and installed the backdoored version of ASUS Live Update at some point in time."

Leo: Oh, so they know this because Kaspersky saw it.

Steve: Yes. Their own Kaspersky instrumentation on those users' machines. They said: "We are not able to calculate the total count of affected users based only on our data; however, we estimate that the real scale of the problem is much bigger" - of course it would be because they don't have their stuff in every ASUS machine - "and is possibly affecting over a million ASUS users worldwide. The goal of the attack was to surgically target an unknown pool of users" - and, okay, listen carefully to this.

Leo: Guess who that might be. Okay.

Steve: Well, "which were identified by their network adapters' MAC addresses." Which is really screwball.

Leo: That's by manufacturer, then; right?

Steve: Well, we know it's ASUS because ASUS is the infection channel. But they're selecting targets based on their MAC address. Okay. So anyway, Kaspersky says...

Leo: All a MAC address tells you is who made that device.

Steve: Well, no. The MAC address is 48 bits.

Leo: No, there's extra stuff, but the first part of it is manufacturer; right?

Steve: Correct. Right. So Kaspersky says: "To achieve this, the attackers had hardcoded a list of MAC addresses in the trojanized samples, and this list was used to identify the actual intended targets of this massive operation."

Leo: Hmm.

Steve: I know. It gets weird, Leo. "We were able to extract more than 600 unique MAC addresses from over 200 samples" - which Kaspersky got from their own customers - "used in this attack." They said: "Of course, there might be other samples out there with different MAC addresses in their list. We believe this" - and this doesn't make sense to me, but we'll get there in a second. "We believe this to be a very sophisticated supply chain attack, which matches or even surpasses the ShadowPad and CCleaner incidents in complexity and technique. The reason that it stayed undetected for so long is partly due to the fact that the trojanized updaters were signed with legitimate ASUS certificates, AsusTek Computer Inc." And Leo, not once but twice. We'll get there, too, in a second.

"The malicious updaters were hosted on the official liveupdate01s.asus.com and liveupdate01.asus.com ASUS update servers. Although precise attribution," they say, "is not available at the moment, certain evidence we have collected allows us to link this behavior to the ShadowPad incident from 2017. The actor behind the ShadowPad incident has been publicly identified by Microsoft in court documents as BARIUM. BARIUM is an APT actor known to be using the Winnti backdoor. Recently, our colleagues from ESET wrote about another supply chain attack in which BARIUM was also involved, that we believe is connected to this case, as well.

"A victim distribution by country for the compromised ASUS Live Updater looks as follows." And I've got a picture of the graph in the show notes just because Kaspersky provided it. But remember this is their view into victims, and it's going to be massively skewed by their customer base. And they acknowledge that. They said: "It should be noted that the numbers are also highly influenced by the distribution of Kaspersky users around the world."

Leo: Mostly in Russia.

Steve: Yes. "In principle, the distribution of victims should match the distribution of ASUS users around the world." They said: "We've also created a tool which can be run to determine if your computer has been one of the surgically selected targets of this attack. To check this, it compares MAC addresses of all adapters to a list of predefined values hardcoded in the malware, and alerts if a match was found."

And I have a link in the show notes for any ASUS computer user among our listeners who is listening to this thinking, ugh. It's https://kas.pr/shadowhammer. That downloads a 50k ShadowHammerCheck.zip, which then checks against the hardcoded list. They say you may also check MAC addresses online. And there it's https://shadowhammer.kaspersky.com. "If you discover that you have been targeted by this operation, please email us at" - and then they have their email address, shadowhammer@kaspersky.com.

So things feel fishy to me about this. First of all, ASUS official servers were being used to supply the initial malware. And the malware was signed by legitimate ASUS certificates. And though Kaspersky's brief summary didn't mention it, other coverage noted that ASUS was being uncooperative in the extreme about this, denying that anything had happened at all, 100% stonewalling. What I find so puzzling and curious is that the malware delivered by ASUS's own servers from ASUS and signed by ASUS used the victim's MAC addresses to identify individual specific ASUS machines. And what's most troubling is that no one but ASUS, the manufacturer of those machines, would reliably know what MAC addresses specific machines have.

As we know, MAC addresses, while not highly secret, neither are they widespread. A machine's MAC address is often printed on the label outside the box, and on the label underneath the machine, for example, in the case of a laptop, or on the label on the machine. But the MAC address is inherently highly local because it's not transmitted over the Internet. The MAC address, as we have often described, provides local Ethernet network hardware addressing for use within a single Ethernet subnet.

So, for example, any IP, Internet Packet router, serves as an intelligent link between separate Ethernet networks, with a different network on each of its interfaces. Unless a router is bridging two networks at the Ethernet layer, the MAC address from one network is removed, and its contained IP packet is routed to another interface where it is reencapsulated with an Ethernet packet for that other network containing its source and destination MAC addresses. So my point is how would some random external malicious agency obtain the physical hardware Ethernet MAC addresses that are only useful for ASUS customers because that's the source of this infection, across a large collection of specific ASUS machines.

Okay. So if we're brainstorming, one possibility is that these were wireless laptops. We've been talking about MAC addresses recently, and MAC address spoofing on WiFi. So if they were wireless laptops, they would have been promiscuously broadcasting their MAC address more or less constantly to every WiFi access point within range. And as we know, a MAC address is a 48-bit value composed of two halves, a 24-bit registered manufacturer number, and a 24-bit serial number within that manufacturer.

Leo: So they had the full qualified MAC address, not just the first half.

Steve: Right.

Leo: So they were specifically targeting machines.

Steve: A machine. A machine, yes.

Leo: Well, that is interesting.

Steve: I know. So the fact of them being ASUS laptops would have been evident from their MAC addresses. So there's, I mean, if you were stretching, some possibility that the machine's MAC addresses of specific individuals could have somehow been gathered over time. But it stretches credulity. If they were wired desktop machines, it's difficult to come up with any theory to explain how some random remote third party could obtain those machines' Ethernet MAC addresses. And if some agency was close enough to a wired machine to obtain its MAC address, it probably has physical proximity anyway, so it wouldn't need to go this weird circuitous route to get its malware into this ASUS machine.

Occam's Razor suggests that when confronted with a lack of definitive evidence, the simplest explanation is likely to be the best. And distressing as this is, this suggests that the entire thing was likely a covert and deliberate campaign, if not on the part of all of ASUS, then an insider action within ASUS. Only ASUS has the certs to sign their update downloads. From Motherboard's reporting, Motherboard said: "The attackers used two different ASUS digital certificates to sign their malware. The first expired in mid-2018, so the attackers then switched to a second newer legitimate ASUS certificate to resign their malware after this." So what sort of security are we to believe exists at ASUS if they were not a willing or begrudging collaborator?

Leo: Let me provide a scenario.

Steve: Okay. But let me finish. One more second.

Leo: Finish, yeah, yeah.

Steve: So the attackers first signed their malware with ASUS's protected, guarded, super-secret code-signing certificate. And those attackers placed that ASUS-signed malware onto both of ASUS's software update servers, where it stays undetected for five months. But later, as that first certificate nears expiration, the "attackers," in quotes, obtain ASUS updated newly freshened code-signing certificate, resign the malware with that updated cert, and replace the soon-to-be-expired malware on both of ASUS's software downloaded servers with freshly signed new malware. That's what we're to believe. And ASUS had no knowledge of any of this.

And so the least that seems feasible is that a well-placed person on the inside arranged for all of this except for the MAC addresses. That would be a very different region within this very large company because the MAC addresses would probably be in sales records for those machines, which indicate who owns which machines with which MAC addresses. So let me just finish quickly. The fact that the follow-up malicious backdoor payload was later sourced from elsewhere gives ASUS some plausible deniability, and Kaspersky indicated that attribution was unavailable at the moment, plus it's very easy to plan a bit of misdirection which would have been in ASUS's interests.

So finally Motherboard wrote: "Motherboard sent ASUS a list of the claims made by Kaspersky in three separate emails last Thursday, but has not heard back from the company," as of yesterday. So three separate emails, ASUS doesn't respond. But Motherboard wrote: "But the U.S.-based security firm Symantec confirmed the Kaspersky findings on Friday after being asked by Motherboard to see if any of its customers had also received the malicious download. Symantec is still investigating the matter, but said in a phone call that at least 13,000 computers belonging to Symantec's customers were infected with the malicious software update from ASUS last year."

Liam O'Murchu, director of development for the Security Technology and Response Group at Symantec, was quoted by Motherboard, saying: "We saw the updates come down from the Live Update ASUS server. They were trojanized or malicious updates, and they were signed by ASUS." So I think that's all my coverage. So I'm just, for our listeners' sake, for five months late last year, ASUS was delivering a malicious download which, if you were one of 600 selected people by the MAC address of your machine, that machine then reached out to a trojan supply server to download additional active malware into your machine.

Leo: So of the thousands of people who were infected, only those 600 got anything malicious.

Steve: Correct. Yes.

Leo: Oh, that's interesting. Sounds targeted.

Steve: Yes. They were infected with - yes.

Leo: So here's the scenario. Let me offer a scenario and see if this makes sense. ASUS has this built in. This is a standard updating procedure. They have all the MAC addresses.

Steve: Yes. Yes.

Leo: They have this built in. Presumably they sign the software when they deliver it. It's an update package. If a bad actor got into ASUS's system and replaced the update package with a malicious package, which then got signed and sent on as if it was a regular update package, all of this would fit. Except for that one little bit, which is, in order to target 600 machines you'd have to have 600 MAC addresses.

Steve: Yes. And remember that the malware was updated when its first certificate was nearing expiration.

Leo: So the bad guy's in there. I mean, we know people, you know, bad guys sit in networks.

Steve: Okay.

Leo: So let's say that ASUS, by the way, a Taiwanese company, not a Mainland China company, but let's say a bad actor from North Korea or some nation-state had access to the ASUS network, got in there, was able to put the malware in there.

Steve: Yup.

Leo: Is it conceivable, I mean, when you hear that something's targeting 600 machines, that sounds like a nation-state going after individuals. It's not a mass attack; right?

Steve: Correct. Correct.

Leo: It's a targeted attack.

Steve: Because they're estimating in five months a million people, a million ASUS customers checked in, got this...

Leo: Because it's part of the normal ASUS update process.

Steve: Exactly.

Leo: Yeah. So couldn't a bad guy who had access to ASUS's network perpetrate this - I mean, it sounds like especially a nation-state bad actor - perpetrate something like this?

Steve: Yeah.

Leo: And I could see why ASUS would be very slow to respond because it looks really bad.

Steve: They're, like, going holy crap, what?

Leo: Yeah. They're, I mean, the first thing, if I'm the CISO at ASUS, I'm going, guys, let's find this intruder. Let's figure this out.

Steve: And we like ASUS. I mean, they're a great company.

Leo: Oh, they're very good.

Steve: They make beautiful hardware.

Leo: I just want to make sure that it doesn't mean necessarily that ASUS is malicious.

Steve: Corporate, right.

Leo: Somebody inside might be bad. Or in my opinion, I mean, look at all the companies that malicious nation-state hackers have gotten into.

Steve: True.

Leo: And just sit there.

Steve: An Advanced Persistent Threat where that person has really deep access to, I mean, like, again, it seems to me that the database where customer to MAC address sales records are is different from the software download/update stuff.

Leo: Right, should be, yeah. Were the MAC addresses sequential or just random? And do we know anything about those 600?

Steve: No. In fact, in 12 days Kaspersky's - that thing is Kaspersky's own SAS. It's the Security Analyst Summit in Singapore where they're going to present a paper on this. So in two weeks we should have some more information from them.

Leo: Very interesting. I mean, it could have been us. Could have been the NSA.

Steve: Yeah. But again, it's weird because it's limited to ASUS customers. I mean, no non-ASUS customer is going to...

Leo: Well, yeah. You start with ASUS. But you get whoever you're in; right?

Steve: But there are 600 of them that were of interest to somebody.

Leo: What if you noticed that, I don't know, the Israeli Embassy had just purchased a large number of ASUS computers.

Steve: Yeah?

Leo: I don't know. I think that we need to know more, obviously.

Steve: And that's a good point, too, because it certainly could be that there is a - if this were targeted, and we don't know targeted by whom, but if - for example, ASUS is a major brand. There's probably many enterprises who have standardized on ASUS hardware. That's what they buy. And so if you know that, like, all of your employees are using ASUS laptops...

Leo: Exactly,

Steve: ...and you can somehow get a list of who's using which laptop by MAC address, then...

Leo: Or just target the whole organization; you know?

Steve: Yeah.

Leo: I mean, maybe, who knows, it could be Lenovo doing this. We want to make ASUS customers unhappy.

Steve: Yeah, that'll do it. So speaking of making people unhappy, Leo, we have the Norsk Hydro ransomware attack. I have a picture in the show notes that someone took of the notice scotch-taped to the door of one of the Norsk Hydro plants. And it's dated, I think it's 3/19. So it says: "Warning: Cyber Attack Against the Hydro Network. Please do not connect any devices to the Hydro Network. Do not turn on any devices connected to the Hydro Network. Please disconnect any device (Phone/Tablet etc.) from the Hydro Network. Await new update." And then it was signed "Security." And then there's some note handwritten in probably Norwegian next to the one that's in English. So, and you brought this breaking news to us during last week's podcast. It was just happening.

Leo: It's funny, we saw almost identical - something in the door a couple of years ago during a ransomware attack. Was it Maersk? I can't remember who it was. I think it was Maersk, the shipping line. Same kind of thing in the door. Don't connect to our network.

Steve: Yeah.

Leo: Wow.

Steve: You'll get hurt.

Leo: Yeah.

Steve: So paraphrasing from Ars - everybody had pretty much the same coverage. Paraphrasing from Ars Technica's coverage, they said: "One of the world's biggest producers of aluminum has been hit by a serious ransomware attack that shut down its worldwide network, stopped or disrupted plants, and sent IT workers scrambling to return operations to normal. Norsk Hydro of Norway said the malware first hit computers in the United States on Monday night. By Tuesday morning, the infection had spread to other parts of the company, which operates in 40 countries, on every continent.

"Company officials responded by isolating plants to prevent further spreading. Some plants were temporarily stopped, while others, which had to be kept running continuously, were switched to manual mode where possible. The company's 35,000 employees were instructed to keep computers turned off, but were allowed to use phones and tablets to check email," maybe using WiFi and not using their network. Or maybe they figured out...

Leo: LTE, not WiFi.

Steve: Yeah, not going to affect them. Chief Financial Officer Eivind Kallevik said during a press conference Tuesday: "Let me be clear. The situation for Norsk Hydro is quite severe. The entire worldwide network is down, affecting our production as well as our office operations. We are working hard to contain and solve this situation and to ensure the safety and security of our employees. Our main priority now is to ensure safe operations and limit the operational and financial impact."

According to Kevin Beaumont, who's an oft-quoted security guy and, Ars said, "tweeting in his capacity as an independent researcher and citing local media reports, the ransomware that infected Norsk Hydro is known as" - and this has been confirmed - "LockerGoga [G-O-G-A]." He said: "LockerGoga doesn't rely on the use of network traffic or on domain name system or command-and-control servers, which all allow ransomware to bypass many network defenses."

An independent research group calling itself MalwareHunterTeam pointed to a LockerGoga sample uploaded to VirusTotal from Norway on Tuesday morning. At the time the malware was first scanned, it was detected by only 17 of the 67 biggest AV products, although detections increased once awareness of the Norsk Hydro infection grew. The malware had also once been digitally signed by security company Sectigo [S-E-C-T-I-G-O]. So the malware had been digitally signed by the security company Sectigo, but the certificate was revoked at an unknown time.

In the statement, Sectigo Senior Fellow Tim Callan wrote: "As a policy, Sectigo revokes certificates used in malware attacks and does not issue certificates..."

Leo: Oh, that's a relief.

Steve: "...to known malware" - it's like, oh, thank you. You know? But wait, Leo. It gets better - "to known malware purveyors." He said: "We encourage security researchers to report instances of malware employing Sectigo certificates at signedmalwarealert@sectigo.com." Okay. Now, when I first read this, I thought to myself, who the heck is Sectigo? Leo? Guess who? Our old friends, Comodo.

Leo: Oh, lord.

Steve: Now operating under a shiny new name.

Leo: Yes. The old one got a little tarnished, yeah.

Steve: They so thoroughly ruined their previous name.

Leo: Oh.

Steve: So horse of a different color. Company by a different name. But yes, Comodo issued the certificate that signed the LockerGoga malware. So a text file that the attackers included in the malware, it's a longer file with a bunch of nonsense. But it starts out saying there was a significant flaw in the security system of your company. "You should be thankful that the flaw was exploited by serious people and not some rookies. They would have damaged all your data by mistake or for fun. Your files are encrypted with the strongest military algorithms, RSA-4096 and AES-256. Without our special decoder ring" - no, not ring, just decoder. "Without our special decoder, it is impossible to restore that data. Attempts to restore your data with third-party software such as Photorec, RannohDecryptor, et cetera, will lead to irreversible destruction of your data."

Leo: Oh, god.

Steve: Okay. So the Norsk Hydro CFO said the majority of the company's plants were operating normally, but that the network shutdown prevented plants from receiving future orders from customers. He said the losses at the moment were "minimal," but he conceded they would grow over time if automated systems aren't restored. Kallevik - that's the CFO - was unable to provide any timetable for how long it would take to disinfect the network. He said company IT teams are working to remove the ransomware - and actually I heard in some separate reporting that Microsoft, a team from Microsoft had flown over to help with that.

He said: "Company IT teams are working to remove the ransomware from infected systems. Once that's done, the teams plan to restore lost data using company backup systems," which he described as "good." Asked by a reporter if the company would rule out paying the demanded ransom, the CFO said: "The main strategy is to use backup."

Lawrence Abrams at BleepingComputer, who is of course everyone's go-to site for ransomware details, added, he said: "It should be noted that while this ransomware has had high-profile targets, it is not the most active one out there targeting companies and has not seen wide distribution." He said: "Furthermore, it's very noisy as it consumes a lot of CPU, causes Windows Explorer to crash repeatedly, and borks the system," he wrote, "enough while encrypting that you can't run normal programs." In other words, it's not very stealthful while it's doing its deed. He says: "Unless it's launched on an idle machine, it would have a good chance of being spotted."

So anyway, what I had heard in subsequent reporting is that they have removed it, and they are restoring from backup. So no mega payout. Oh, and there was no fixed price given, either. These guys, in their note, they instructed the infected company to contact them and strike up a dialog, and that the amount requested would be a function of how long it took to reach out and contact them.

Leo: Oh, yeah. Don't delay. Call today.

Steve: Call today. That's right. So anyway, that's the background on that attack. And we're going to talk about Microsoft renaming Windows Defender. Microsoft has renamed Windows Defender Advanced Threat Protection, you know, APT, to the more generic Microsoft Defender Advanced Threat Protection. Why? Because they're offering it for the Mac. It was a post last Thursday on...

Leo: Oh, that's so interesting. Wow.

Steve: Yeah, it really is. A Microsoft blog posting was titled: "Announcing Microsoft Defender ATP for Mac." They said: "Today we're announcing our advances in cross-platform, next-generation protection and endpoint detection" - oh, and I should mention that Linux is coming - "endpoint detection and response coverage with a new Microsoft solution for Mac. Core components of our unified endpoint security platform, including the new Threat & Vulnerability Management also announced today, will now be available for Mac devices.

"We've been working closely with industry partners to enable Windows Defender Advanced Threat Protection customers to protect their non-Windows devices while keeping a centralized [they called it] 'single pane of glass,'" meaning everything monitored in a single location. "Now we're going a step further by adding our own solution to the options, starting with a limited preview today. As we bring our unified security solution to other platforms, we're also updating our name to reflect the breadth of this expanded coverage: Microsoft Defender ATP.

"There are two key parts for cross-platform support for Microsoft Defender ATP on Mac." They said: "A new user interface on Mac clients called Microsoft Defender ATP. The user interface brings a similar experience" - meaning like look and feel - "to what customers have today on Windows 10 devices." And then: "Reporting for Mac devices on Microsoft Defender ATP portal." And then they said: "Microsoft Defender ATP can be installed on devices running macOS Mojave, High Sierra, or Sierra which you want to manage and protect."

And then they said: "In a limited preview, this app provides next-generation antimalware protection and allows end users to review and perform configuration of their protection, including," you know, and then they had a list of all the standard AV things - quick scan, full scan, deep scan, quarantining, blah blah blah.

And then they said: "Users will also be able to configure advanced settings: Disabling or enabling real-time protection, cloud-delivered protection, and automatic sample submission; adding exclusions for files and paths; managing notifications when threats are found; manually checking for security intelligence updates." And they said: "Microsoft AutoUpdate service is also installed, which ensures that the app is kept up to date and properly connected to the cloud."

So Leo, I am completely out of the loop on Mac AV. Are there multiple vendors offering Mac AV?

Leo: Oh, yeah. It's just [crosstalk] the same.

Steve: So it's the same as on the PC?

Leo: Yeah, I mean, you know, I think you probably agree with me, I'm not a recommender of AV in general.

Steve: You know I agree that, I mean, I've got my little fort with the flag on it looking at me from the tray. So I've got Microsoft's integrated Defender in my system where it's not bothering me.

Leo: Well, and it comes with Windows 10. It's kind of, because it's operating that way, you're not...

Steve: Yeah.

Leo: But that's now installing a third-party standalone AV on Mac, which I wouldn't - I don't think I would recommend. Apple does have, not an antivirus, but has some pretty sophisticated security stuff on there, including malware scanning and Gatekeeper. So I don't think you need it.

Steve: Yeah, I mean, and I agree. And I wonder if - do you know if Microsoft's portal stuff integrates with Apple's so that you could stay native and still get this single pane of glass thing? Or do you think they're just, like, separate? I don't know.

Leo: I have no idea what the implementation specifics are. But this isn't surprising. Remember Microsoft put out that Chrome and Firefox plug-in. You talked about it last week.

Steve: Yeah.

Leo: So basically there's an Edge sandbox inside Chrome and Firefox.

Steve: An Edge takeaway. Oh, you're about to go on the Internet. Let's switch you over to Edge.

Leo: But in their defense, their sandbox technology's on Edge, it's based on Edge, so that's why they do that. But I think this is the new Microsoft. They don't care that much about Windows. Windows is not the crown jewels by any means of the company anymore. And so why not put a Microsoft everywhere? It is bizarre. I would never have thought that we'd be looking at antiviruses for Macintosh from Microsoft.

Steve: No, it is. And I wonder, I mean, homegrown? Or maybe I wonder if they acquired somebody that was already there?

Leo: Oh, that's a good question. I mean, Defender's based on Giant antivirus, remember that, way back when they bought it. But by now it's so different than Giant originally was that it's a unique product.

Steve: But that's generally how Microsoft acquires a big new technology, you know, something that's really alien to what they already have is they just, you know, they acquire it because money is not a problem at Microsoft.

Leo: Right. So, yeah, I'd love to know more about this. And for now, I don't know about you, but I wouldn't recommend it.

Steve: No. I just wanted to let our listeners know. And maybe there's an enterprise need for...

Leo: That's probably it; right?

Steve: Yeah, because that is where, you know, the Azure Cloud rigmarole. Okay. So for our listeners who may be responsible for a website based on PHP, which generates PDFs on the fly, a severe security bug was found six months ago in the most popular PHP library for creating PDF files. The three most popular libraries used by web servers to create PDF files for like invoicing, purchase receipts or whatever on the fly are TCPDF - which I think is a clever name - TCPDF, MPDF, and FPDF. And as we know, well, I'm sorry. And now we know, after the very responsible disclosure by a security researcher who waited, not six hours, not six days, not six weeks, but six months after the flaw was disclosed privately so that it could be fixed, that now today we know that a serious remote code execution flaw exists in the one of those three which is the most popular, which is TCPDF.

The vulnerability is a variation of another researcher's discovery which was first found by a guy named Sam Thomas, a researcher at Secarma who in a series of experiences last summer showcased a deserialization bug - and we've talked about this, and I'll remind our listeners about that in a second - affecting PHP apps over the summer of 2018. He released a research paper detailing PHP serialization attacks against WordPress and Typo 3 CMS [Content Management System] platforms, but also the TCPDF library, which is embedded in the Contao CMS.

Then in a blog post just this past weekend, an Italian researcher who goes by the online handle Polict, P-O-L-I-C-T, revealed a new PHP serialization flaw impacting TCPDF, like in the same way as the one discovered by Thomas last summer. Polict says the vulnerability he found can be exploited two ways. The first is on websites that allow user input to be part of the PDF generation process, such as for example when adding a name or an email address or other details which would then be bound into the resulting PDF.

The second would be on websites that contained cross-site scripting vulnerabilities where an attacker is able to plant malicious code inside the HTML source code that will then be fed to the TCPDF library to convert it into a PDF. One way or another, the requirement is to supply deliberately malformed data to the TCPDF library, which causes that library to call the PHP servers, something known as the "phar," phar:// stream wrapper, which later then abuses the PHP deserialization process to run code on the underlying server.

So what the upshot of all this is, it is a very potent remote code execution vulnerability. He notes that it's a complex attack requiring advanced PHP coding skills to exploit. And as we have discussed previously, deserialization attacks are difficult to uncover and are the bane of many programming languages. Ruby, Java, .NET, and PHP all have these problems. Recall that when we serialize something, we convert a complex structured data into a linear, thus a serialized byte stream. I mean, that's like for storage. In order to store a complex data structure, it's serialized into a blob. Then when we deserialize in order to read it back and reconstruct it, we're reversing the process to convert that byte stream back into the original complex data structure.

But all too often, coders who write the deserializer, well, they're typically the same people who wrote the serializer. So they inherently assume that the byte stream they are being fed back is the one that their serializing code generated, that it was created by a well-meaning serializer. But of course a deserializer is necessarily an interpreter. It reads the incoming byte stream and interprets its meaning in order to reconstruct this complex data structure. And as we have often noted on the podcast, interpreters are inherently fraught with problems.

So Polict said that he reported the vulnerability, and it was given a 2018-CVE-17057, to the TCPDF library author last August, so a long time ago. The TCPDF team then released an updated TCPDF 6.2.20 a month later, in September, to address the issue. Unfortunately, when the TCPDF team did that, they accidentally reintroduced the earlier vulnerability reported by Sam Thomas while attempted to patch the one reported by Polict. Like I said, interpreters are finicky beasts.

So finally both issues were resolved in v6.2.22. Polict published the details about this vulnerability last week after waiting six full months after the patch's release, due to the bug's severity, the fact that it is remotely exploitable. It runs the attacker's code on the attacked server, and it affects any PHP-based website that uses TCPDF to render its PDFs on the fly, which are a lot. So he waited six months to allow updates to actually get out to the endpoint.

In their coverage of this, ZDNet noted that the TCPDF library is one of today's most popular PHP libraries and has been used, they wrote, "all over the place - in standalone websites, in content management systems, CMS plug-ins, CMS themes, enterprise Intranets, CRMs, HRMs, invoicing solutions, many PDF-centered web apps, and others. And," ZDNet says, "patching isn't as easy as it sounds. In some cases this means replacing a file and editing a build instruction, but in other places it might require rewriting large swaths of code."

So anyway, and we know that PDF libraries are sort of that unsexy back end that are exactly the sorts of things that tend to be overlooked by web developers and website maintainers. So the takeaway for our listeners is, if you're responsible for any website that does generate on-the-fly PDFs, make sure that, if you're using TCPDF, that it is at v6.2.22 or later. Because now that this is public, bad guys are going to know, I mean, we know what the pattern is here now. Something like this that's juicy is revealed publicly, the bad guys roll up their sleeves, and they start looking for opportunities to exploit. So you want to make sure that in the, I mean, ideally in the intervening six months, just in the natural course, that library would have been updated because you've had plenty of time. If not, then don't delay.

News is from Bleeping Computer that Microsoft's leaked Edge browser based on Chromium is looking really good. Bleeping Computer reported that over this past weekend a leaked build for the Chromium-based Edge browser has been released that is providing users with their first look at the upcoming browser from Microsoft.

I think it was Lawrence who said: "If you are currently using Chrome, the reports indicate that this Edge preview browser feels, performs, and basically offers, not surprisingly, the same feature set." Oh, yeah, it was Lawrence. He writes that: "Microsoft has been quiet regarding their upcoming Microsoft Edge Insider browser, but a slow trickle of leaks has provided quite a bit of information. With this leaked build, users get their first look at the upcoming Edge browser, which from all reports feels like it has the best chance of putting a dent in Google Chrome's market share.

"Users who have tested the leaked build have also stated that the browser performs really well when browsing the web, and that it is more than ready for public preview. Microsoft has modified the layout of the browser to make it feel more like a Microsoft app. For example," he says, "the Settings pages have a left-hand navigation panel similar to other Windows 10 apps." I like the way the Control Panel looks because I saw a picture of it. "Microsoft also included their own services into the browser," not surprisingly. "For example, Google Safe Browsing has been removed in favor of Microsoft SmartScreen.

"In addition to setting up a dedicated Microsoft Extension store, Edge" - and this is really interesting - "also allows users to enable the installation of extensions from Chrome's web store. While Microsoft states that these extensions are unverified as a warning, it provides an enormous pool of extensions for users to install right off the bat, right from release." So this feels significant. We've got Google with currently the largest browser share, and we've got then the largest browser built into the largest desktop and laptop share, effectively merging into one.

So, I mean, this will only be Windows 10. But still it'll automatically - we know what Microsoft will do. As soon as this thing is ready, all Edge will be converted to this, and a month later everybody running Windows 10, which is a little over half of the Windows install base now, will be essentially merged with everybody who is - whether you're running Windows 10 or Google's Chrome, and everybody not running Windows 10 who is using Google Chrome, effectively all using the same browser. So that's a lot of weight for that browser. And it makes me glad that we have a SQRL plug-in running on Firefox and Chrome because that'll soon also then be running on Edge. And that'll be cool.

Leo, I titled this next piece "From our pained abbreviation department" because, boy, this one is a stretch.

Leo: Yeah, yeah. I know what you're going to talk about, too.

Steve: They really wanted to call this SHAKEN/STIR.

Leo: Little Bond reference, yeah.

Steve: Yeah, exactly. SHAKEN, S-H-A-K-E-N. And, oh, this one was a reach. Signature-based Handling (so we've got the S and the H) of Asserted (now we have the A) Information (okay, we're going to forget about the I) Using (forget about the U) toKENS.

Leo: Oh, boy.

Steve: And we're going to get the K-E-N to finish the SHAKEN. So, ouch. And then of course STIR, that comes from Secure Telephone Identity Revisited. So it's like, okay, you guys. The military, I think, has the best acronym people. I don't know where, how that happens.

Leo: Well, Congress is pretty good, too. They come up with some wild acronyms.

Steve: Yeah. Anyway, it's...

Leo: They're mostly retronyms, you know.

Steve: Yeah. Horrible as those abbreviations are, taken together, SHAKEN and STIR do deliver a protocol for authenticating phone calls with the help of cryptographic certificates. The U.S. Federal Communications Commission has been pushing for SHAKEN/STIR's adoption and has imposed an end of 2019 hard deadline for networks to implement the protocol. I have a link to the FCC.gov/call-authentication. And I had to cut out some of the self-serving Ajit Pai nonsense from it. But what I did keep reads - it was titled "Combating Spoofed Robocalls with Caller ID Authentication." And notice it's spoofed robocalls, not robocalls.

So this reads: "FCC Chairman Ajit Pai: 'American consumers are sick and tired of unwanted robocalls." Amen. "This consumer among them," he says of himself. "Caller ID authentication will be a significant step towards ending the scourge of spoofed robocalls. It's time for carriers to implement robust caller ID authentication."

And then in the same announcement: "How Will Caller ID Authentication Help Consumers? Caller ID authentication is a new system aimed at combating illegal caller ID spoofing." Okay, I didn't know it was illegal, but apparently it is, but everyone does it. "Such a system is critical to protecting Americans from scam spoofed robocalls and would erode the ability of callers to illegally spoof a caller ID, which scam artists use to trick Americans into answering their phones when they shouldn't." Oh, okay. Well, I'm not going to pick this apart. I'll keep reading.

"Additionally, consumers and law enforcement alike could more readily identify the source of illegal robocalls and reduce their impact." That's true. "Industry stakeholders are working" - and I didn't know this, and I'm glad for this - "to implement caller ID authentication, which is sometimes [unfortunately] called SHAKEN/STIR. Once implemented, it should greatly help the accuracy of caller ID information and should provide consumers with helpful information for determining which calls are authenticated.

"SHAKEN/STIR is a framework of interconnected standards. SHAKEN/STIR are acronyms for Signature-based Handling of Asserted Information Using toKENs and the Secure Telephone Identity Revisited standards. This means," they write, "that calls traveling through interconnected phone networks would have their caller ID 'signed' as legitimate by originating carriers and validated by other carriers before reaching consumers." Okay, now, of course, this is what we've had for HTTP ever since - what was SSL?

Leo: Secure Sockets Layer? What?

Steve: The original Netscape browser. Netscape 4 or something, at least. So, yes, we've had, I mean, all they're talking about is that the originator of a call sign the caller ID. Thank you. And that the signature be verified by the receiving network. Okay. Doesn't seem like rocket science. They're going to do it, which is good news.

"In November of 2018, Chairman Pai" - this is, again, their announcement - "sent letters to voice providers asking those that apparently had not yet established concrete plans to protect their customers using the SHAKEN/STIR standards to do so without delay. In February 2019, Chairman Pai welcomed many carriers' commitment to meeting his timeline for implementation, called on others to catch up, and made clear that the FCC would consider regulatory intervention if necessary."

So the news is last Wednesday, almost a week ago, AT&T and Comcast announced that they had successfully tested what they believe to be the first SHAKEN/STIR-authenticated call between two different telecom networks.

Leo: And of course the content was "Watson, come here, I need you."

Steve: So apparently, work on the SHAKEN/STIR protocol has been underway for a while, and telecom operators have used it internally, but only for calls originated and terminated within their own networks. So they've been getting ready to reach out and touch someone, and they finally did.

Leo: I'm so glad you said that.

Steve: So once broadly adopted, incoming calls not signed, and that's of course I guess still a possibility, could simply be dropped as a subscriber option. And spoofed caller ID for signed calls would become impossible. So this will not, by itself, stop robocalling; but it will certainly chill the callers, who then significantly know that law enforcement, if they were to use signed caller ID, they can't be spoofed anymore. So anyway, I just want, you know, the robocalling is a problem. The spoofing of caller ID is a problem because you can't block - I think we've talked about this recently, in fact. You can block a robocall, but then it just comes back on a different number because they just make them up.

Leo: Right, right, right.

Steve: So there's no benefit at all to doing that. And so anyway, progress. And then maybe at least it'll give us initially some more control. And then we just need legislation, if that ever happens. So we'll see.

I wanted to quickly note to all we users of PuTTY, P-u-T-T-Y, and of course that's an old play. The original teletype was known as the TTY, pronounced "titty."

Leo: Really.

Steve: Yes.

Leo: I never heard that.

Steve: Absolutely.

Leo: Maybe that was your group.

Steve: Us old-timers, it would be, "What happened to the titty interface?"

Leo: No, no, no, no. TTY. No, no, no, no.

Steve: I'm sorry, TTY, that's what it's called.

Leo: I always said TTY.

Steve: All of us old-timers.

Leo: That's, see, you're two years older than me, so that's why.

Steve: That's right. So it is a very popular SSH and Telnet client for securely with SSH connecting to remote systems over a network. They have just announced and released 0.71. And I don't know what's - why they're not at 1.0. It's like, get off, I mean, just make it 1.0. I mean, it's been around for a long time. Anyway, for whatever reason it's at 0.71. This update patches eight high-severity flaws which affect - they're very much like we were talking about the RDP exploit where, if you remote desktopped, to make a verb of it, if you remote desktopped to a Windows server, if it was malicious it could, because your RDP client assumes a benign server, like a friendly protocol, it could get you. Well, that's what these eight high-severity flaws are. So it's unlikely that you're going to PuTTY to a remote server. But should you do so with any version of PuTTY before 0.71...

Leo: I believe that's PUTitty. Right? I might be wrong there. Sorry.

Steve: I really don't want to say...

Leo: No, it's not [crosstalk].

Steve: I mean, so everybody should update. I've got a copy, and I updated when I ran across this news.

Leo: Surely there's a better terminal for Windows by now. This thing has been decades - it's decades old.

Steve: It is very old. There is - I'm looking at the link to it right here. I also use - I can't see it on my desktop. It's here somewhere. Anyway, there are other SSH. But it's very popular. And it's, you know...

Leo: Oh, yeah. I installed it for years, along with Cygwin.

Steve: Yup.

Leo: But seems like there must be - maybe, of course, now you can put Bash on there with LTS, so...

Steve: Yeah, yeah.

Leo: Or WSL, rather, yeah.

Steve: So Firefox, I wanted to also note quickly, they're going to conditionally use - they're experimenting with conditionally using the Windows cert store to avoid the AV SSL scanning issues that they have been subjected to recently. When we were exploring the recent mess that was caused by Firefox Update, where the AV vendors who had added their own AV HTTP TLS interception roots into the Windows root store had not done so for Firefox. And at the time we noted that Firefox did have an option, which was disabled by default, to import the Windows root store. That was security.enterprise_roots.enabled, which is normally false, but you can set it to true. And if you do, all of those problems immediately vanish because you're using the Windows root store.

Now Mozilla is considering automatically flipping that preference proactively in detectable situations to avoid future problems with third-party AV. I have a link to their Bugzilla posting, and they called it "Retention/Engagement impact of enabling the Enterprise roots feature in presence of an AV." And the bug says: "Several AVs recently broke HTTPS with their HTTPS scanning features that require their certificates to be added to our certificate store." And we discussed Avast, but there's also Bitdefender had a Mozilla bug filed for it, and Kaspersky also.

They said: "The security team confirmed that having the preference security.enterprise_roots.enabled set to true would have fixed all of these issues without known regressions; and we want to validate that, in the presence of an AV, enabling this preference would have a positive impact on retention" - I guess that means Firefox user retention, like they're not going to lose people to Chrome or Edge because of a problem that Firefox is causing - "and engagement." And they said, parens: "(We cannot detect a change in certificate error page displays through telemetry since telemetry is sent over HTTPS," this person writes, "that breaks in these instances.)"

So then they said, and this is what I thought was interesting: "Description of the impacted population." First bullet point: "Win 10 and Win 8 release users." And they said, parens: "(The API allowing detection of an AV registered with the system was only available since Window 8)." And then also: "An AV is registered with the system and is not Windows Defender. This information is available on the telemetry under 'sec.antivirus.'" So the point being, from Windows 8 on, there is a means of determining whether a non-native, that is to say, not what we used to call Windows Defender, which is now Microsoft Defender, when it's not that, but it's a third party, there is a way for Mozilla to detect that.

So in those instances, what they're exploring is proactively flipping this so that, if you are on Windows 8 or 10, if you have a third-party AV, not the built-in Microsoft AV, and you are using Firefox, you will not have a problem. So anyway, just good news for people who want to use Firefox and a third-party AV on Windows 10 without the hassle of needing to, well, actually, you can just go manually flip that switch if you're a listener to the podcast. Everybody else will have it done automatically.

Also a quick note that last Thursday Microsoft announced that their new Windows Virtual Desktop product was now available for public preview. The technology, this Windows Virtual Desktop product, which is perversely named Remote Desktop - which seems like a name collision to me. But maybe they're going to amalgamate the existing Remote Desktop with what is called Windows Virtual Desktop into a single thing called Remote Desktop. So there is no collision because there won't be two different things. Anyway, it is a technology that allows enterprise to move their desktops and applications to Azure for hosting on Windows 10 or Windows 7 Virtual Machine operating systems that are always secured with the latest updates.

What's interesting is that this will continue to get, if you choose to use Windows 7, it will automatically be getting updates through 2023. I will remind our listeners that an Azure trial subscription is available for free for a year. And apparently this looks like a move by Microsoft to come up with a compromise with users, enterprise users, where there are some things they want to do that they have a hard time moving from Windows 7 to Windows 10.

So what Microsoft is billing this as is as a way where desktops can be migrated to Windows 10, while at the same time those applications that are stubborn and need still to stay on Windows 7 can be accessed through the cloud on this Azure Windows Virtual Desktop that will just become Remote Desktop. And it will be part of your Azure subscription. You create a tenant, an Azure tenant, and then you publish desktops and applications to it, and then you're able to use it just like the application was running natively and locally on your desktop. So anyway, and that gets updates, as does the extended service of Windows 7, through 2023. So just another benefit.

And, finally, as I said, Leo, you and Mary Jo and Paul were talking about, were wondering out loud last Wednesday about what would Microsoft do about bugging people about Windows 7 end of support date. We now know, thanks to coverage by Lawrence at Bleeping Computer. The update is KB4493132. The good news is so far it is not selected in Windows Update for installation by default. So seems unlikely that Microsoft is not at some point going to flip that on, but we'll see. It doesn't identify itself as providing end-of-life notifications. It states that it is an update to "resolve issues in Windows." Yeah. Like you're still using Windows 7.

Leo: No, we need some more money from you, that's the issue.

Steve: That's right. So we now know what the dialog looks like. It's a big dialog, and it reads: "After 10 years, support for Windows 7 is nearing the end. January 14th, 2020 is the last day Microsoft will offer security updates and technical support for computers running Windows 7. We know change can be difficult." Yeah, and of course it would be a lot less difficult if you didn't make Windows such a - but anyway. "We know change can be difficult. That's why we're reaching out early to help you back up your files and prepare for what's next." And then the good news is, in fine print, itty-bitty down in the far lower left, there is a checkbox that says "Do not remind me again." Which of course you can turn on, and then I guess you close the box. It doesn't say okay.

Leo: Yeah, there's no okay button, yeah.

Steve: It just, yeah, there's a close. So we'll hope that works. Lawrence has torn this thing into bits in his posting. I've got a link in the show notes. I mean, so he's figured out what it is that it downloads, what's the CAB file. It puts two entries into Windows Scheduler. Where it runs. It looks like it's got multilingual HTML assets so they can change the pretty picture and wording anytime they want to.

But the good news is it isn't currently being installed by default. It's hard for me to believe Microsoft will not turn that on once they're sure it's not causing problems, or once they get feedback from people who have turned it on, who always turn on all the optional updates. So it's one of the optional updates which is not, you know, that's optional. Eventually it seems certain that they're going to feel they need to warn people that the end is nigh. So we'll see what happens with that. But at least we know what it looks like. And they are saying stop bothering me about this in the future.

I just wanted to remind our listeners of the excellent three season series "The Expanse." It first aired on Syfy Channel. It's now available from Amazon Prime. And Lorrie and her son were out of town together traveling for several days last week. And so I thought, okay, this is my chance to watch the final season.

Leo: Lorrie doesn't like sci-fi.

Steve: No, no. She actually really, really, really does like sci-fi. But she's not big on violence, and...

Leo: She didn't watch the first two with you, either. So she's kind of behind.

Steve: Precisely. So there's really no - it didn't make sense to try to catch her all up. Oh, my god, is it good. It's just - it is. I remember when it was on Syfy thinking, where did this come from? How is this possible? Because Syfy Channel, much as I love them, they just produce crap, I mean, just horrible, barely watchable stuff. Now "Galactica," that was an exception, too. But I don't know how this was produced, but this was just excellent. And so it's three seasons. I don't remember how many episodes the previous two had, but this one had 13. And it is top quality. I mean, it demonstrates that our computer-generated graphics has really come a long way because you can kind of sense that some of it is CG, but not rubbery looking. I mean, it sells it.

And, but I mean, like I'm watching as they were under acceleration, and somebody was trying to get to somebody else, and they were having to roll the ship. And the physics was perfectly done of this guy hanging onto this bar hand-over-hand while loose tools were flying from one side of the room to the other, and they were having to duck them as the ship maneuvered. I was just - I was stunned. I thought, these guys have some serious science advisors. Anyway, now, the only downside is it's a little bit into politics. There's Mars and the U.N. and the Belters are sort of the three political factions. And so if you're really intolerant of any, like, political back story or political machinations, it might turn you off a little bit. Or you can just kind of ignore that part and just watch the fun because - anyway, it was great.

So I just wanted to put it on people's radar. If you're an Amazon Prime person, it's free for you. And it's probably 39 hours of good - and it was an hour-long episode. It wasn't like 45 minutes where it used to be for TV and they cut out the commercials. So I recommend it. If the politics doesn't turn you off, because there is that. And I did want to remind, or I just wanted to mention that the next book in the never-ending 75-book series...

Leo: Ryk Brown is crazy.

Steve: Ryk Brown.

Leo: He's nuts. You didn't even have to tell me.

Steve: R-Y-K, Ryk Brown. It's "The Frontiers Saga." I am in love.

Leo: I don't think there's a 75-book series anywhere else. That's it. That's got to be the one.

Steve: No, no. And you know, Leo, when I was writing the show notes I thought, how would I describe this? And I said, well, we all - we're familiar with comfort food. This is comfort reading. I mean, it's not, you know, earthshaking, world-changing, bleeding-edge sci-fi. It's just a bunch of people who you get to know really well. The writing is excellent. The plot twists and stuff, I mean, there's an arc in place. It's just so pleasant. And so, yes, I am now on book number 26.

Leo: Twenty-six, not 76.

Steve: No, he's going to write - so what he's laid out is five 15-book arcs.

Leo: Seventy-five.

Steve: Seventy-five. And so the first one was 15.

Leo: That's insane.

Steve: I'm now on book 11 of the second 15, so I'm on book 26. And it is so good. It is just - it is so good. I was reading something else last week, again because Lorrie was out of town, so I had a little more spare time in the evenings. And so I thought - I continued to read what I was reading, and I thought, where is that next book? And I checked, and it had been released. It's like, ah. And so I immediately switched to it, and I'm sadly about a third of the way through. It's just - I just wanted to say "The Frontiers Saga," if you just like comfort sci-fi, you know, it's just great.

Over on the SQRL side of the world, everything is moving forward apace. I'm working to get the static content of the web forums finished. Then I get to take my Windows client to final. A couple people, I mean, this is all working out well. I know it's taking a long time. But, for example, now that more people are using it, the observation was made that users are used to, when they change their password, it's like everything is cloud-synced; like if you change your password with LastPass, then LastPass, you know, like it's changed everywhere. Or if you change your password with a website, then of course when you log in somewhere else to that site, it's the changed password.

Well, SQRL is different, of course, because the slogan I've come up with, the best way of thinking of this, is that SQRL logs into websites for you. That is, so it offloads all of the mess. It logs into websites for you. But you have to remind SQRL or prove to SQRL that you're you and not somebody else using your SQRL identity. Anyway, the point being that if you change your SQRL password, that is, the password you use, the only one you need, the one password for SQRL - and yes, it can be your face or your fingerprint, and people are loving that - that doesn't automatically change your password on a different device, like on your Android or your iOS device, where you also have your SQRL identity.

So the point is I'm going to add a little reminder in the wizard for changing your password, just to remind people that, to prevent confusion, they should make the same change they have just made here on other places where they have and use SQRL. So those little final touch-ups. So anyway, I have to do that. There's a bunch of development code still in the client that gets removed. And so I'm about to take it to 1.0 because it hasn't been yet. Then I need to update the online documentation of the spec because there's lots of pressure now that we've got now someone working on a pure JavaScript implementation. We've got the Android client, a lot of work on that, and the iOS client coming along fast, and the web extension. Someone's doing something for React that'll be available for NPM. And so lots of pieces coming together.

The point is that I did see - we were discussing, there'd been some discussion in the SQRL forums about remaining logged into a site, which is not in any way related to SQRL because SQRL's role is just to authenticate you to the site. And then of course the site maintains a cookie is how your browser session stays connected. But what happens is it's so easy to sign in, almost fun to sign in using SQRL, that it does arguably sort of shift the balance away from necessarily remaining signed in everywhere because it's so easy now to reassert one's identity.

Anyway, so day before yesterday I saw a post. It was followed up yesterday, and I thought I would just share these. So day before yesterday at 1:00 p.m. someone calling himself "Gristle," he said: "I am loving logging in with SQRL so much that I actually don't want the cookie to persist longer than an hour, or maybe even minutes. It's just so fun and simple to use SQRL."

And then I think he came back and saw his own post, and yesterday he quoted himself, saying: "I'm not joking. I find myself logging out and in and out and in, just for fun. It's crazy, I know, but it's just so cool. Reminds me of the first time I tried unlocking a phone with my fingerprint. I kept locking and unlocking it, unbelievable that there is so much crypto behind such a simple gesture." So anyway, I will - we're not far away from getting this thing off the ground. I mean, it's no longer just me pushing forward on the client and the protocol. All of the other pieces that we need to create a functioning ecosystem are, I mean, the Android and the iOS client both work. People are using them like crazy now. So we're getting there.

Oh, and I talked to Ralf, I can't pronounce his last name, it's Wondratschek or something. I'm sorry, Ralf, for mangling your name. But he was the person who wrote that Android client a long time ago, and some people were getting confused about it because it's still on the Google Play Store. I asked him if he would just make a mention that this was no longer current for the SQRL protocol as it exists today. And so he said he wanted to keep it there for his rsum; but, yes, he would make a change, and he did. So thank you, Ralf, for that.

I'm wondering if I want to skip talking about SpinRite because of where we are with time.

Leo: Oh, we've got all the time in the world, unless you've got a date.

Steve: No. Okay. So...

Leo: That's right, Lorrie's out of town. Go ahead.

Steve: This is some good techie stuff that I thought our listeners would find interesting. It came from a reply that I wrote. And as I was writing, I thought, okay, I'm going to make this a little more general reply that we can use. So one of our customers said: "I have a few questions about SpinRite." He said: "Those sectors you read with surface scan errors, do you try to overwrite them so that sector mapping by S.M.A.R.T. takes place?" He says: "Recall the remapping will only take place when the data is overwritten, which in a file system may never happen until the disk is completely, sector by sector, reformatted." He says: "Imagine a file that is read often, but never rewritten, and resides on one or more sectors with surface scan errors."

Okay. Well, that's full of misunderstandings. But that prompted me, when my tech support guy forwarded it to me, it prompted me to explain. And I said: "If you think about this a bit, you'll see how this can work." I said: "Surface defects do not manifest when sectors are written because writing to the disk is an entirely blind process. The drive gets no information about the readability of a freshly written sector. That only occurs when a subsequent attempt is made to read back that sector. Because a read-after-write would be prohibitively slow, all modern drives incorporate Error Correction Code (ECC) technology which creates a safety margin to defend against the possibility that, due to some problem which may have occurred during writing, or a surface anomaly that interferes with reading, the data that was written to a sector cannot be read back without the aid of algorithmic correction."

And I wrote: "ECC operates by appending additional carefully designed redundant data to the end of the sector, based upon the sector's intended content. This extra data allows the location of a read-back problem to be identified along with its length and its error mask. From this data the drive is able to determine, within limits, what data was originally written to the sector. Those error recovery limits are the number of separate errors occurring within a single sector, and each error's bit-run length."

I wrote: "On today's drives, data densities have grown so high that some level of background error correction is occurring more or less continuously. It's only when the extent of correction required for read-back recovery begins to approach the drive's inherent limits on recovery that the drive gets worried and decides to remove that worrisome sector from service."

And then he wrote: "If you do overwrite them and force remapping, how do you stop a disk from being rendered useless because the max limit (say 100) of remapping has taken place due to your reading all sectors and forcing all those remappings?"

And so I wrote: "A sector can be misread due to its own aging or transient vibration or physical shock which forced the head slightly off track when the sector was last written or when it is now read. To determine whether the error was transient and not caused by an actual defect, SpinRite first writes inverted data, then reads it back. It then rewrites the original data and reads that back. If any of those three reads - the original read, the inverted read, and the reinverted read - result in sufficiently threatening data read-back errors, the drive will take that sector out of service and remap the sector."

And then he asked: "When a sector is found with a surface scan error, are you able to tell me what NTFS (Windows in my case) file is using that bad sector?"

And I wrote: "Since a bad sector is replaced with a good sector at the drive's physical interface level, this falls beneath the operating system's file system. In essence, all surface errors and remappings are transparent to the OS and the file system, so nothing needs to be done there. Before this autonomous drive-level sector-replacement technology existed universally as it does now, SpinRite did this itself. All early versions of SpinRite managed surface defects for the drive. SpinRite would determine which cluster the defective sector occupied.

"It would then determine what role that cluster was playing in the file system, if any. If the cluster was in use by a directory or a file, it would obtain the closest nearby free cluster, copy the bad clusters data into the new cluster after recovering that data, and relink the replacement cluster into the system's file system. It would then mark the defective sector as bad in the low-level format, and mark the defective cluster containing that sector as bad in the system's cluster management tables.

"All of that technology still exists in today's SpinRite for possible use if it should encounter the need. But SpinRite is able to interact with all modern drives to perform these data relocations underneath the file system. Other than on floppy disks, SpinRite's high-level defect management is never needed today." So there's a little kind of interesting background on what SpinRite does to keep people safe and to manage to help to work with drives to help them manage the health of their sector pool and how they relocate and replace sectors when they are truly damaged. And how, because we don't immediately trigger on a single problem, how we separate transient errors from true physical surface defect errors.

You know, remember, our long-term podcast listeners remember how you can shout at a drive, and suddenly it will slow down the rate at which it's able to transfer data. Well, that shouting is vibration, which forces the heads off track just enough to cause them to misread what's there. And so the surface, the platter has to spin around again and try to do a retry. And so, I mean, it really is the case that drives are very vibration-sensitive. So one tip is make sure that your drives are not, well, aside from being shouted at, are operating in an environment where they are protected from undue vibration. They really do try to protect themselves.

So last Wednesday, Thursday, Friday was Trend Micro's organized ZDI, their zero-day initiative, with support from Microsoft, Tesla and VMware: the Pwn2Own competition. The first day resulted in four successful hacks and one partial win, with the contestants earning a total of $240,000 U.S. in cash awards, plus because it's the "own" part of the Pwn2Own, the laptops which were used to demonstrate their research. So in their traditional blow-by-blow style, Trend Micro described the first day as follows.

They said: "The contest started with the team of Fluoroacetate" - who always dominates this competition - "Amat Cama and Richard Zhu, targeting the Apple Safari web browser. They successfully exploited the browser and escaped the sandbox by using an integer overflow in the browser and a heap overflow to escape the sandbox. The attempt nearly took the entire allowed time because they used a brute force technique during the sandbox escape. The code would fail, then try again until it succeeded. The demonstration earned them $55,000 and five points toward Master of Pwn.

"The Fluoroacetate duo returned targeting Oracle VirtualBox in the virtualization category. Although their first attempt failed, the second attempt successfully used an integer underflow and a race condition to escalate from the virtual client to pop calc at medium integrity. It wasn't the race condition that caused their first failed attempt. Their memory leak was working, but their code execution failed. Everything aligned on the second attempt, which earned them $35,000 and three more Master of Pwn points.

"Next up, Pwn2Own newcomer anhdaden of STAR Labs also targeted Oracle VirtualBox. He also used an integer underflow to escalate from the virtual client to execute his code on the hypervisor at medium integrity. Interestingly, he used a unique integer underflow different than the previously demonstrated underflow. His first foray into Pwn2Own netted him $35,000 and three Master of Pwn points. This also marks the first Vietnamese winner at Pwn2Own. We hope," they wrote, "to see more of him in the future.

"In their final entry for Day One, the Fluoroacetate duo targeted the VMware Workstation. They leveraged a race condition leading to an out-of-bounds write to go from the virtual client to executing code on the underlying host operating system. They earned $70,000 USD and seven additional Master of Pwn points. This brings their Day One total to $160,000 and 15 Master of Pwn points."

And I'll note that this is a full VMware escape, executing code on the host OS for which, as we recently noted, Zerodium has upped their bounty to half a million dollars. So it seems clear that these guys are the good guys who are interested in increasing security, rather than using their talent to indirectly attack others, which of course is what happens if you sell your zero-day exploit to Zerodium.

"The final entry in Day One saw the phoenhex & qwerty team targeting Apple Safari with a kernel elevation. They demonstrated a complete system compromise. By browsing to their website, they triggered a JIT [Just In Time] bug, followed by a heap out-of-bounds read, used twice; then pivoted from root to kernel via a Time-of-Check-Time-of-Use (TOCTOU) bug. Unfortunately, it was only a partial win since Apple already knew of one of the bugs used in the demo. Still, they earned themselves $45,000 and four points toward Master of Pwn.

"On Day Two the day began with Fluoroacetate duo back again, targeting the Mozilla Firefox web browser. They leveraged a just-in-time bug in the browser, then used an out-of-bounds write in the Windows kernel to effectively take over the system. They were able to execute code at system level just by using Firefox to visit their specially crafted website. The effort earned them another $50,000 and five more points toward Master of Pwn.

"The prolific duo returned with perhaps their greatest challenge of the competition. Starting from within a VMware Workstation client, they opened Microsoft Edge and browsed to their specially created web page. That's all it took to go from a browser in a virtual machine client to executing code on the underlying hypervisor." And I'll just stop for a moment to say, do we realize how much that exploit would have meant to Zerodium? Zerodium would have moved heaven and earth to obtain that hack. I mean, you visit a page, and you're running code on the hosting hypervisor. Unbelievable.

Anyway, ZDI Trend Micro continues: "They started with a type confusion bug in the Microsoft Edge browser, then used a race condition in the Windows kernel, followed by an out-of-bounds write in VMware Workstation," so a serious exploit chain. "The masterfully crafted exploit chain earned them $130,000" - so no small potatoes there, either - "and 13 Master of Pwn points. They now have a commanding lead with 33 points total. In the two days of the competition, they racked up a total of $340,000 as a result of their phenomenal work. Tomorrow, they will attempt to cap their week off with a successful demonstration in the automotive category."

Then the blow-by-blow continues: "The third attempt of the day had Niklas Baumstark target the Mozilla Firefox web browser. He used a just-in-time bug in the browser, followed by a logic bug, to escape the sandbox. In a real-world scenario, an attacker could use this to run their code on a target system at the level of the logged-on user. The successful demonstration earned him $40,000 and four Master of Pwn points.

"The final attempt for Day Two had Arthur Gerkis of Exodus Intelligence targeting Microsoft Edge. The newcomer to Pwn2Own wasted no time by using a double free bug in the renderer, followed by a logic bug to bypass the sandbox. His debut entry earned him $50,000 and five points toward Master of Pwn."

And they write: "That brings Day Two to a close. We awarded $270,000 for nine unique bugs today, which brings the Day Two total to $510,000." They said: "Join us tomorrow as we debut the automotive category with the two final entries of Pwn2Own Vancouver 2019."

So Day Three, the automotive category. Two months ago, mid-January, writing for Forbes magazine, Thomas Brewster wrote: "Think you can hack a Tesla? Now's your chance. And you could win more than $900,000 in the process. For the first time ever," Thomas wrote, "Pwn2Own, perhaps the world's best-known competition for ethical hackers, will have a Tesla Model 3 opened up for participants to break. Prizes range from $35,000 to $250,000. The more difficult the hack, the greater the prize. The lowest prize," he writes, "will go to an as-yet-unspecified attack on the car's infotainment system. The top $250,000 reward will go to the first person or team who can break any of the three critical Tesla internals: the Gateway, the Autopilot, or the VCSEC.

"The Gateway," he writes, "acts as the central hub for controlling data flowing around the Tesla. Taking control of that system would give a hacker power over many of the car's functions. Manipulating Tesla's Autopilot could lead to all-too-obvious problems. Imagine if the hacker simply shut Autopilot down without the driver noticing. The VCSEC is the part of a Tesla responsible for a variety of security functions, including the alarm. Again, it's not hard to guess just what a hacker could do if they commandeered that part of the car." And he concludes: "Another $100,000 is on offer for the first to hack the doors by breaking the key fob or mobile app unlock tech. Starting the car without owning the legitimate key will also land a lucky hacker" - although I would say no luck is involved - "$100,000."

So what happened on Day Three? ZDI writes: "The day began not with a bang, but with a whimper, as the Team KunnaPwn withdrew their entry from the automotive category. Although they did not demonstrate any of their research at this contest, we hope they submit some of their research to our program in the future."

And I found through other research, this was supposed to be the day when Team KunnaPwn demonstrated a hack of the Tesla Model 3's VCSEC security component. So that was the expected or possible quarter million dollar hack. But they withdrew from the competition. And again, VCSEC is an abbreviation for Vehicle Controller Secondary, and as we know is responsible for security functions such as the alarm.

However, Fluoroacetate was next up. ZDI wrote: "When their scheduled time arrived, the dynamic Fluoroacetate duo of Richard Zhu and Amat Cama thrilled the assembled crowd as they entered the vehicle. After a few minutes of setup, and with many cameras rolling, they successfully demonstrated their research on the Tesla Model 3, which is a Chromium-based web browser. They used a just-in-time bug in the renderer to display their message on the Tesla's infotainment system and earned $35,000." Of course, this is Pwn2Own, so they also got to drive away in the car.

So they wrapped it up saying: "Overall, the three days of Pwn2Own Vancouver 2019 have been a great success. We have," they wrote, "awarded a total of $545,000 for 19 unique bugs in Apple Safari; Microsoft Edge and Windows; VMware Workstation; Mozilla Firefox; and, in its inaugural year, the Tesla infotainment system. And it should come as no surprise that the Fluoroacetate team of Richard Zhu and Amat Cama have been crowned the Master of Pwn for 2019. Their amazing research earned them $375,000 over the contest and resulted in 36 Master of Pwn points. They dominated Pwn2Own Tokyo and have carried that wave through to the spring. We can't wait to see what's next for this pair of talented researchers."

So that's the story. And hats off to everybody involved. This is a great way to incentivize researchers to ethically hack and to fix problems without exposing end users to them because of course there is a "full disclosure" room at the Pwn2Own conference where everything is disclosed in return for the award money. And then the manufacturer, the affected products get the benefit of getting patches before they are made public. So yay to everybody involved.

Leo: Yeah, especially Fluoroacetate. They made some big bucks.

Steve: Boy, these guys, they've got skillz, S-K-I-L-L-Z.

Leo: Yeah. Do you figure this is like a full-time job for them? All year long they search for these exploits, and it all culminates in March in Vancouver?

Steve: Yeah. I mean, as we said, what was it, SPOILER was last week, and I think week before was hacking as a career.

Leo: Yeah.

Steve: I mean, you can, if you're good, you can support yourself now. We're in an industry where there is money, either responsibly or, I would argue, irresponsibly disclosed; there's money if you can find problems. And, boy, the target richness of the environment is not decreasing.

Leo: Yeah.

Steve: It is growing exponentially.

Leo: I remember they won a lot of money in Tokyo, too, for Pwn2Own Tokyo.

Steve: Yup, exactly. So not even annual, it's biannual.

Leo: Yeah. It's an interesting - I wonder if - it must be a certain brain type and certain skill set that just makes you well suited for this kind of stuff. Because probably you would enjoy it, but most people would just go nuts staring at hex dumps and fuzzing readouts and, I mean, it'd be horrible.

Steve: I could do it. I really - I prefer creating.

Leo: Yeah, and reverse engineering and, yeah.

Steve: I love finding my own mistakes. I just...

Leo: Debugging is fun. I agree.

Steve: I love debugging, yes.

Leo: It's fun if you're not tearing your hair out. If when you find the bug, let's put it this way, when you find the bug it's fun. It can be less fun if you can't find it. That's really frustrating, when you say, no, this should work, this should work. What's going on? I don't understand it.

Steve: Yeah, one of the things that I created that I mentioned before was an API for servers, which is what Rasmus used to create the SQRL login for our XenForo forums. And you don't have to know anything about SQRL. The problem is, of course, I code in MASM, and so I implemented the first one in assembly language. We have a guy who's in the process of - I've given him my source, and so he's using that as a template. And I also have a full spec which is online, and so he's using that. Anyway, Paul is rewriting it in what will then be Open C, so it will be completely cross-platform, and anyone will be able to host a service API on their server in order to easily, trivially, add SQRL support to a server.

The point is that he's found several bugs in my code, and I've received a couple of pieces of email saying, Steve, if this happened here, wouldn't that just not return a response to the user? And it was like, ooh, you're right. So, I mean, I love having someone carefully rereading my own code because, as we know, it's very difficult to see your own mistakes. But he's found a couple. So I've been really grateful for that side effect also, even though it's not clear to me anyone will ever use mine because it's MASM. But at least it's fixed.

Leo: Hey, we offer a handy-dandy assembly language API for those of you.

Steve: That's right.

Leo: Well, Steve, it's been fun. And I don't mind 20 pages, 25, 30, it's always great. Security Now!.

Steve: We just had a lot to - I threw a bunch of stuff out because there just wasn't time to get to it all.

Leo: Yeah, some days there's a lot to say.

Steve: That's right. Thank you, industry, for never giving us a boring podcast.

Leo: You'll find copies of this show at Steve's site, GRC.com, along with transcriptions. So if you like to read along while you listen, they're all there. Along with SpinRite, the world's finest hard drive recovery and maintenance utility. And SpinRite. Did I say SpinRite? I did say SpinRite.

Steve: Yeah.

Leo: And SQRL, that's the other thing. And a lot of other things. In fact, it's a black hole. You'll go there, and you won't come out for hours.

Steve: That's a black hole, right.

Leo: Read everything you can. It's fun. GRC.com. We have audio and video on our site, TWiT.tv/sn. It's also on YouTube. It's everywhere. And, you know, the best thing to do would be get a podcast program, there are so many good ones out there, and subscribe. That way you'll get every episode. And if you want to go back in time, every one of the 707 episodes are stored at TWiT.tv/sn, so you can get them bit by bit. And there are a number of scripts out there, I have a few on my blog, PowerShell scripts and so forth, that will suck the entire set to your server. From our server to yours. Thank you, Steve. Have a great evening, and I'll see you next time on Security Now!. Bye-bye.

Steve: Thanks, Leo. Bye.