
# Pwn2Own 2018

**Description:** This week we discuss the aftermath of CTS Labs' abrupt disclosure of flaws in AMD's outsourced chipsets; Intel's plans for the future and their recent microcode update news; several of Microsoft's recent announcements and actions; the importance of testing, in this case VPNs; the first self-driving automobile pedestrian death; a SQRL update; a bit of closing-the-loop feedback with our listeners; and a look at the outcome of last week's annual Pwn2Own hacking competition.

High quality (64 kbps) mp3 audio file URL: http://media.GRC.com/sn/SN-655.mp3
Quarter size (16 kbps) mp3 audio file URL: http://media.GRC.com/sn/sn-655-lq.mp3

---

SHOW TEASE: It's time for Security Now!. Steve Gibson is here. An update on the AMD flaws, some more details. Turns out this is probably the real deal. We'll also talk about the big Pwn2Own conference, just went on last week. All the machines that are hacked, all the machines that couldn't be hacked, and a sterling play-by-play, all coming up next on Security Now!.

**Leo Laporte:** This is Security Now! with Steve Gibson, Episode 655, recorded Tuesday, March 20, 2018: Pwn2Own 2018.

It's time for Security Now!, the show where we get together with the smartest guy I know when it comes to this stuff and figure out what's going on in the world around us, Mr. Steven Gibson of the GRC, Gibson Research Corporation, dot com. Hi, Steve.

**Steve Gibson:** Leo, it's great to be with you once again.

**Leo:** Live long and prosper.

**Steve:** I'm looking at our Security Now! Episode 655 and thinking, ooh, in 11 weeks we're going to be at 666.

**Leo:** Devilish. The devilish episode.

**Steve:** It is.

**Leo:** We've never done a 666, have we.

**Steve:** We never, no. And that also puts us at two thirds of our way through the history of this podcast because of course that's going to end at 999 because I only have three digits.

**Leo:** You're really serious about that. I was hoping you were just joking.

**Steve:** I've got everybody worried about that right now. They're like, no, no. Someone suggested use negative numbers. It's like, no, that doesn't work.

**Leo:** It's about six more years. But you know how fast this first 13 went. It could go like that.

**Steve:** Yes, yes, yes. And you still have your hair. So we talked last week about the fact that the day after last Tuesday's podcast, which would have been, what, the 14th, Wednesday the 14th. And the 15th and the 16th were the Pwn2Own conference, which has been since 2007 an annual event up at Canada, hosted by Trend Micro's Zero Day Initiative, and also in this case sponsored by Microsoft and VMware. We have the results of that competition, for better or for worse, and sort of a little bit of both. So that's going to be our - we're going to wrap that up.

And we're also going to require from you, Leo, a dramatic reading of the two-day event because, with a little bit of massaging of what they provided in a couple of their blog postings, I pulled it together and pumped it up a little bit so that it's a little bit more like an Olympic judging contest.

**Leo:** You're not going to make me read this whole thing, though, are you?

**Steve:** Well, if you don't, I will, because I think people will find it interesting. But I think it'll be pretty good.

**Leo:** I don't know if I can do the whole thing in a funny voice, but I'll do my best.

**Steve:** You don't have to do it - if you need to fall out of character, that's fine. As I was reading it, I was thinking, okay, this would be good, the way this thing was put together.

**Leo:** Cool.

**Steve:** But we're also going to talk about the aftermath of last week's surprise announcement by CTS Labs that I know you've been talking about on other shows.

**Leo:** With some suspicion, I might add.

**Steve:** Yeah, well, about motives and intent and so forth. And in reaction to that, their CTO has formally produced a statement that I want to share because there's something to empathize with also. And of course we just now 90 minutes ago have a reaction, the first official reaction beyond the first little sentence from AMD themselves. So we want to talk about that.

Also Intel now has announced plans for their own future. And we have some recent microcode update news that was actually, I should have looked last week to see what was happening with that KB4090007 page because it's much expanded. So that's good news. We also have several Microsoft recent announcements and actions. Also another lesson, and they just keep on coming, about the importance of testing, in this case three popular VPNs, all of which were found wanting when tested.

Also I do want to touch on, just because it's an interesting bit of technology, I know you were just talking about it over on MacBreak Weekly, this last evening's first death caused by a self-driving automobile. Also I have a bit of an update on the progress with SQRL. And as I was waiting for the podcast to start, I was thinking how anxious I am to get back to it because we're right down to the last few little things to be resolved. It's getting very exciting.

Also we'll have a bit of closing-the-loop feedback with our listeners. And then we're going to talk about the mixed results from this year's Pwn2Own, which significantly had no Chinese participants. And that's a little bit of a tease for why it was a bit of a lackluster result.

So our Picture of the Week I got a kick out of. It's the result of last week's Pwn2Own. We talked about how there would be the award given, the Master of Pwn. And this is the guy, Richard Zhu, or I guess maybe it's Zhu, I don't know...

**Leo:** Zhu. Zhu.

**Steve:** Richard Zhu, who won the contest. And this is him wearing his Master of Pwn sweatshirt, giving himself some props, which he has earned and which we will be talking about later in the show, when we talk about the result of the Pwn2Own 2018.

**Leo:** He got more than a nice jacket, though, I must say.

**Steve:** Yes, he did.

**Leo:** Yes.

**Steve:** Yes, he did. So as you commented last week, one of the best-named exploits to come along, the Ryzenfall, which was of course one of the four classes. The biggest one was the hardware problem in the Chimera vulnerability. These were announced as we were literally going to press. As I was putting the podcast together Monday evening, as I

was assembling everything, the topic was different. And Tuesday morning it was, oh, crap, this is important. We need to change the title of the podcast. And so I quickly rearranged things.

So in the week that has ensued, the controversy of course, which we discussed last podcast, was that the CTS Labs, a relatively new, unknown security firm, about a year old, did not provide AMD with what has now become customary, well, customary is 90 days. As we know, Google's Project Zero gave Intel 200 days because they understood the nature of the severity and knew that Intel would take this seriously. Anyway, this was not the case with CTS Labs. There's been a bunch of fur flying over the last week, some people saying that there was no proof of this.

The good news is that one independent researcher, Dan Guido, whom we referred to also last week, who's the CEO of Trail of Bits, did independently verify the report and tweeted to that effect on Twitter. He said: "Regardless of the hype around the release, the bugs are real, accurately described in their technical report, which is not public as far as I can tell, and their exploits work." So at this point we're beyond wondering whether there's something here, and we are now at the point of, like, okay, so what's the impact of this to end users? How does the world feel about the way CTS Labs acted, and what is AMD's position?

So first of all, the consensus is, much like Spectre and Meltdown, these are difficult to exploit. They do require - and this is still a little unclear because we don't have details. But what we're being told is that administrative access is required for these to be exploited, which is not true of Spectre and Meltdown. Which caused much more concern, that is, any unprivileged process was able to leverage, if it was clever enough, the Spectre and Meltdown vulnerabilities in order to get cross-process information leakage, which was the big concern. What we're being told is that this requires administrative access. So you have to first have some privilege elevation in order to get that access and then exploit it.

On the other hand, later this podcast we'll be talking about elevation of privilege attacks, all of which the attackers at least week's Pwn2Own were able to achieve. So we shouldn't take too much, I mean, AMD is trying to play up the fact that, oh, you have to be at admin. Well, that's not looking like that's that difficult for an unprivileged process to obtain for itself. So AMD, as I mentioned before, is on record now acknowledging the four problems, that is, basically saying we're not happy with CTS Labs for not giving us any time; but, yes, we acknowledge ourselves that these are problems.

Okay. But there was a lot of pushback about, like, against the way CTS Labs handled this. So I want to share the letter than the CTO just published on the AMDflaws.com site addressing this because, I mean, this is an issue for the industry moving forward. And I think he raises some interesting and worth discussing points. He starts by saying, by way of a disclaimer: "I am a technical guy," he writes. "I love building things and researching things. I do not like the whole world of PR." He says: "It is too messy for me. I have written this letter in my own language, without PR proofing. Please forgive me if there are any grammatical errors, or not written according to correct writing standards."

On the history of their publication of these exploits he said: "We have started researching ASMedia chips about a year ago." And that's about when this group pulled themselves together and declared themselves an entity. "After researching for some time, we have found manufacturer backdoors inside the chip which give you full control over the chips." And what's interesting, too, is that a lot of us have these chips on our motherboards. So the focus of their disclosure was AMD because AMD had formally used a descendant of these in their own chipset. But this may still have some farther reaching consequences

which have yet to have been articulated.

Anyway, he says: "...hardware manufacturer backdoors giving you full control over the chips." And that's the ASM1042, ASM1142, and ASM1143. And if someone's interested, although this hasn't been leveraged as far as we know yet into exploits, you can probably determine if those are the ASMedia chips on your motherboard.

He said: "We wanted to go public with the findings, but then saw that AMD have outsourced their chipset to ASMedia. So we decided to check the state of AMD. We bought a Ryzen computer and whimsically," he writes, "ran our exploit PoC [proof of concept], and it just worked out of the box. Full Read/Write/Execute on the AMD Chipset, as is, no modifications. To be honest," he says, "we were a bit shocked by it. How they have not removed the backdoors when integrating ASMedia IP [intellectual property] into their chipset," he writes, "is beyond me. So then we said, okay, what on earth is going on in AMD, and started researching AMD."

He says: "It took time to set up the working environment to start communication with the AMD Secure processor. But after reaching a full working setup and understanding of the architecture, we started finding vulnerabilities, one and another and another. And not complex, scary, logical bugs, but basic mistakes, like screwing up the digital signatures mechanism. At that point, about once a week we found a new vulnerability, not in one specific section, but across different sections and regions of the chips. It's just filled with so many vulnerabilities that you just have to point, research, and you'll find something." He says, parens: "(Obviously a personal opinion). After that," he writes, "we decided we have to go public with this. I honestly think it's hard to believe we're the only group in the world who has these vulnerabilities, considering who are the actors in the world today, and us being a small group of six researchers."

So he then starts a next section titled "Responsible Disclosure." He writes: "I know this is an extremely heated topic for debate, where everyone has a strong opinion. Unfortunately, I also have a strong opinion on this topic. I think," he writes, "that the current structure of responsible disclosure has a very serious problem. If a researcher finds a vulnerability, this model suggests that the researcher and the vendor work together to build mitigations, with some time limit - 30, 45, 90 days, whatever - at the end of which the researcher will go out with the vulnerabilities. The time limit is meant to hasten the vendor to fix the issues.

"The main problem in my eyes with this model," he writes, "is that during these 30, 45, 90 days, it's up to the vendor if it wants to alert the customers that there is a problem. And as far as I've seen, it is extremely rare that the vendor will come out ahead of time notifying the customers, 'We have problems that put you at risk. We're working on it,'" he writes. "Almost always it's post-factum: 'We had problems. Here's the patch. No need to worry.' The second problem is, if the vendor doesn't fix it in time, then what? The researcher goes public with the technical details and exploits, putting customers at risk? How we have accepted this mode of operation," he writes, "is beyond me, that researchers advertise at the end of the time limit the technical details of the vulnerabilities because the vendor didn't respond. Why should the customers pay for the vendor's lack of actions?"

He says: "I understand this is the model today, and people follow suit. But I think we can do better. I think that a better way would be to notify the public on day zero that there are vulnerabilities and what is the impact, to notify the public and the vendor together, and not to disclose the actual technical details ever unless it's already fixed; to put the full public pressure on the vendor from the get-go, but to never put customers at risk. This model has a huge problem. How can you convince the public you are telling the

truth without the technical details? And we have been paying that price of disbelief in the past 24 hours." Oh, boohoo, but still.

He says: "The solution we came up with is a third-party validation, like the one we did with Dan of Trail of Bits. In retrospect, we would have done this with five third-party validators to remove any doubts, a lesson," he writes, "for next time. I know there are many questions, and a whole lot of confusion. We are trying our best to answer reporters, update our site with Q&A, and clarify what's going on. So far the media focus was on CTS, and I think I understand this. But very soon we will have to deal with the fact that a huge company with products spread throughout millions of computers in the world is riddled with so many problems that it's unclear how to even address this. If you have any technical questions, please contact me at [his email address]. I'll try to answer as many as I can."

So that's his position. One of the attacks that they have come under is this issue of the AMD stock price, this sort of being the equivalent almost of insider trading, where if they were convinced that the public would react negatively to this news, then they could go short on AMD stock and then make a hyperbolic announcement which actually, I mean, this follow-on statement is a little stronger and worrisome than their original AMDflaws website and technical disclosure that was a little more sober than what they came out with. And, I mean, I would argue, I guess my position is I can sort of see what he's doing.

Certainly AMD is under massive pressure, although it's difficult to believe that they would have been any less motivated to fix this in a timely fashion had this disclosure been done privately. That is, he's right about the size of the problem, how widespread it is, how bad it is. He had proof-of-concept code. He had one person he could point AMD to. AMD could certainly validate it just as quickly as Dan at Trail of Bits could. And especially on the heels of Spectre and Meltdown they would understand the severity of this. So my take is this was not them behaving correctly. It's not as if AMD wouldn't have immediately taken action. I think it's clear that they would have for something like this.

So it looks like they were wanting to be glory hounds in an industry where common practice has been to give the company some opportunity to fix this because in fact their behavior has demonstrably put users at risk. I think that's clear. Given the nature of the problems, the fact that, if it's the case that the chips are riddled with security vulnerabilities of the kind they say, and having so clearly pointed us to where they are, there is now a window which is wide open which would not have been if firmware updates were available. And almost all, there's the Chimera, remember, had two aspects, a firmware problem and a hardware problem. They claim the hardware problem cannot be fixed, that it's in hardware. AMD has in their own disclosure some talk of mitigation, although they say that they're working with the vendor, so they're also not saying that this isn't the case that there is this kind of problem. But all of the rest is firmware fixable, meaning that patches can be created.

And in fact in AMD's statement from about two hours ago they are saying firmware patches are underway and will happen absolutely as fast as possible. Given that this is a third-party chipset, you know that AMD is putting pressure on ASMedia, and ASMedia is going to be moving as quickly as they can to preserve their relationship with AMD. So looking at this whole thing, I have to say that, if we say that the goal is to truly protect end users, to protect the people who have the systems, then as long as there are measures in place to patch firmware - and we now know that both Linux is able to and Microsoft are able to, as part of the boot process, to apply a dynamic firmware update. As long as that's in place, that should have a chance to happen before the discoverer of the flaw is able to or does make a public disclosure. I think that's really where we have to

come down.

**Leo:** Should we be worried about other ASMedia chips in Intel machines, a USB controller or anything like that? I mean...

**Steve:** I think so. I think that one of the takeaways is that they've already found what they call "manufacturer backdoors" inside three other chips - ASM1042, 1142, and 1143. And, I mean, I know, I'm not sure what the vintage of those chips are, but I'm sure I've seen when I'm looking at device drivers...

**Leo:** Oh, yeah, they're all over the place, yeah.

**Steve:** Yeah. And so this could be much farther reaching. So they already had backdoors in these other multiple ASMedia chips. And it was when they discovered that AMD had subcontracted this chipset family to ASMedia, that these guys said, whoa. And they got a Ryzen machine, and they ran their proof of concept, and it took over the computer. And so that's when they said, whoopsie, this is bigger than one audio chipset vendor. This is AMD and everybody who's using those chipsets. So I don't think we've heard the end of this ASMedia problem. And what we may be seeing is driver updates from ASMedia that fixes the firmware in their own chips.

**Leo:** It's just those four ASM chips that we know of so far.

**Steve:** Right, right, right. And we will of course on this podcast be covering that as we get more news and information.

**Leo:** Yeah, and AMD indicated that they thought they could have a firmware fix pretty quickly, which was interesting.

**Steve:** Yes, yes. Which is to say that they verified this themselves. I'm sure they stomped on the Taiwanese ASMedia hard and said, okay, look, fix this tonight. And these things should not take that long. I would argue also, this is much more of, like, a software feeling fix than the Intel microcode stuff. I mean, it's firmware. But this is a higher level software-y feeling fix than Intel's. So it makes sense that it would be, like, whoops, you guys aren't checking a digital signature properly. That's a higher level, easier to fix thing than microcode in the Intel architecture.

And speaking of which, an editorial was just posted from the Intel CEO, and it looks like his name is Brian Krzanich?

**Leo:** Krzanich.

**Steve:** Krzanich, thank you. Brian Krzanich. And he's talking about, okay, where Intel goes from here. "Advancing Security at the Silicon Level" was his editorial. And basically what he wrote comes down to, in the second half of this year, the next silicon coming

from Intel, which is the Intel Xeon scalable processors which are codenamed Cascade Lake, as well as the eighth-generation Intel core processors, both expected in the second half of this year, will be incorporating next-generation hardware fixes for the Spectre and Meltdown problems.

So as we know, right now they're scurrying around with microcode patches, essentially, to the existing, what is it, fifth-, sixth-, and seventh-generation architectures. They'll then be doing essentially everything for the last five years. But naturally anything next they do will incorporate, from manufacturing, improvements. And because they have some time now, there's no articulation yet. And this has caused some annoyance in the industry. He's talking about security, like, walls, protective walls.

In his editorial he says: "While Variant 1 will continue to be addressed via software mitigations, we are making changes to our hardware design to further address the other two," meaning the much more concerning Spectre problems. He said: "We have redesigned parts of the processor to introduce new levels of protection through partitioning that will protect against both variants 2 and 3," which are the Spectre variants. "Think of this," he writes, "this partitioning as additional protective walls between applications and user privilege levels to create an obstacle for bad actors." So no more details than that at this point.

And the processor watchers in the industry who are really wanting to know what Intel is doing are expressing their annoyance at not having any more detail yet. I'm sure that'll be forthcoming. And he's at the CEO level, so he's just reading what engineering told him to read. But what this does say is that, as would be expected, whereas what they've having to do is essentially to emergency patch the existing architectures of the existing chips, next generation and all chips moving forward will get probably better mitigation, not only built in, but have the opportunity to be better.

And as I mentioned at the top of the show also, I forgot to have us take a look at where we stood with 4090007. That's that KB page. If you google KB4090007, that's the Intel microcode updates which Microsoft is publishing in their own updates. And whereas the first time we looked there were only two, and that's what then motivated me to update my InSpectre app so that it clearly displayed the CPU ID because you know whether there's something for you based upon those IDs. Now there's a ton. It's actually eight different CPU IDs spanning Skylake, Kaby Lake, and Coffee Lake. And one of the other things that Brian said in his editorial was that they now have firmware updates available and published for all processors released in the last five years. So Skylake, Kaby Lake, Coffee Lake, as well as Haswell and one other that I can't remember. But everything in the last five years.

So they've jumped on this. And at this point it's still a "go get it" from Microsoft. But Intel has made those available. Microsoft has incorporated those into an evolving update that you can find by googling KB4090007. And then you can use InSpectre to get your CPU ID and see whether it is now among that growing list, in which case installing that update will give you, essentially will modify Windows so that, every time it boots, your processor is freshly patched with the firmware changes that allow the worst of the Spectre vulnerabilities to be mitigated in hand with changes to Windows.

And speaking of changes to Windows, this month's Patch Tuesday had a bunch, I think it was 15 different vulnerabilities that were fixed. But among the things that were fixed was that Microsoft released updates for, finally, and I was hoping for this, I'm going to have to change the language once I confirm. In fact, I'm just turning around. I have a Windows 10 x86 machine...

**Leo:** I was wondering what that was behind you.

**Steve:** Yes.

**Leo:** I've never seen a laptop behind you, yeah.

**Steve:** You can see it back there. It's been updating for about five hours. These updates are not - when you do a major move, it can take a while, and I haven't had this machine on for a while. So it looks like it's done. And I wanted to have - my hope was that it wouldn't be able to take that long, and I'd be able to verify. But I assume the news is correct, which is that Windows 10 x86 now got updates in March, that is, it already has them, for microcode running the Win 10 Fall Creators Update, as well as Intel's sixth-gen Skylake processors. And now we know - so that would have been built into the updates, which is news. That means that the automatically delivered updates included Skylake, instead of having to go get it yourself. Not all the other ones yet.

So it sounds like Microsoft is staging the release of this. Right now, if you really want to go get it, you can, for example, to get also Kaby Lake and Coffee Lake coverage, which is what's in the KB4090007. But given the fact that just this month, a couple weeks ago, they did incorporate the microcode updates for Skylake, which were the first ones to appear at that patch-it-yourself page. This suggests that, as soon as Microsoft is absolutely sure that the firmware isn't going to cause any other problems, that this will get rolled into future - probably April's Patch Tuesday. So that means that ultimately everybody's going to get themselves fixed. But now, so far, the Windows 10 Fall Creators updates had been alone. They moved it to x86. But now we also, as a consequence of Patch Tuesday, we have the x86 also versions of 7 and 8.1, as well as Server 2008 and 2012.

So Microsoft's doing the right thing. They are getting everything covered. And that means I'm going to be changing the language in InSpectre here in a second, after the podcast is done. And I've verified it myself to confirm that the x86 variants of these Windows OSes have been fixed because in InSpectre at the moment it just says I hope Microsoft does this. The good news is they are doing it. So that's great. And in which case InSpectre will be able to show you some better news.

Okay. So Microsoft has followed Intel in beefing up their bounties. Remember we talked about this in February, so last month. Intel beefed up their own bug bounty program. And I don't quite understand why both of them, Intel and Microsoft, have a year-end deadline or expiration. But both Intel and Microsoft's bug bounty programs are only valid through December 31st. Which I don't get because it's not like bugs are going to stop happening.

**Leo:** Bugs expire in December. Didn't you hear? No more.

**Steve:** Yeah, we wish.

**Leo:** We fixed them all.

**Steve:** So Intel has announced last month that they will pay out up to a quarter million dollars - yes, $250,000 - for reports of new, similarly serious to Spectre and Meltdown side-channel bugs. In other words, they want to know about these. If the severity ratings are 9 or 10 on the CVSS, the Common Vulnerability Scoring System that is, like, top-level severity, then you get a quarter million dollars if you bring that news to Intel. If the severity is between 7.0 and 8.9, meaning just shy of 9, then you can get as much as $100,000. And below 7.0 you're down at $20,000. But still nothing to sneeze at, I mean, if it's some sort of a side-channel problem.

Well, so Intel announced that last month. Now Microsoft has virtually duplicated that program. Same sort of story applies. And I'm not sure what happens if you co-release the news to both of them. Do you get two paydays? I mean, I hope. So if anyone else is able to find anything wrong like Spectre or Meltdown, Microsoft says we're going to pay you for that information. And similarly it could be worth a quarter million dollars.

Their announcement said: "Today, Microsoft is announcing the launch of a limited-time [same expiration] bounty program for speculative execution side-channel vulnerabilities. This new class of vulnerabilities was disclosed in January 2018 and represented a major," writes Microsoft, "advancement in the research in this field. In recognition of that threat environment change, we are launching a bounty program to encourage research into the new class of vulnerability" - which is cool - "and the mitigations Microsoft has put in place to help mitigate this class of issues."

So they're saying a quarter million dollar jackpot is sitting here until the end of the year for anyone who can find anything else like this, or something that we missed in these vulnerabilities. So they have a four-tier. They said new categories of speculative execution attacks get you a quarter million dollars. Azure speculative execution mitigation bypass is worth 200,000. Windows speculative execution mitigation bypass also up to 200,000. And then instance of a known speculative execution vulnerability in Windows 10 or Microsoft Edge.

And they say: "This vulnerability must enable the disclosure of sensitive information across a trust boundary." In other words, some way of using a known vulnerability that they missed, you can get $25,000. So this is all good. I mean, we've talked about this. Google, what, is there like at $1,600 or something. I mean, and we've talked in the past about these bug bounties offering such little money that it wasn't clearly going to incent a lot of research. But a quarter million dollars, that's going to get you maybe thinking about other clever ways of leveraging our current processor technology.

So that's what we need. We've seen, for example, with cryptojacking, where the ability to turn virtual cyber currency into money has created a huge amount of pressure to install crypto mining unwittingly into people's browsers or servers. In other words, the fact that there's money at the end of the trail makes a difference. And so now we've got money at the end of the speculative execution trail, both from Microsoft and Intel. Maybe, if you can get both, half a million dollars for finding some new problem. And that's what it takes. It takes somebody wanting to find a problem to do so. And these companies have the money, so I think this is good. And I just don't know why it expires on New Year's Eve.

**Leo:** Well, maybe they, you know, they don't want to be liable forever.

**Steve:** I think that's probably it, yes, is that otherwise it's an open door, and they could be held to it maybe forever. So, yeah, I think you're probably right. The attorneys

probably got involved and said, eh, you need to not let this thing be open forever.

So, okay. Speaking of my laptop that took five hours to update this morning, I did want to inform our listeners that Microsoft is changing the way Windows does its twice annual major updates. By Microsoft's estimate, the Creators Update, which was released almost a year ago, took on the average of about 82 minutes, during which time those rollercoaster dots - is there any formal name for those things, Leo?

**Leo:** I like that. I nominate that for the name.

**Steve:** It always reminds me of a rollercoaster where you kind of go over the top, yeah. It's too bad they didn't use the infinity symbol, though, instead, for the rollercoaster, because that would be a little more representative of how long you're waiting. But anyway, so the rollercoaster dots, while you are offline, was 82 minutes. Now, in the Fall Creators Update that some of us have just suffered through, they improved that so that the rollercoaster dots were spinning, by their computation or their estimate, about 51 minutes.

So we're coming into, and we referred to this in the last couple weeks, it hasn't been officially named, but it's provisionally being called the Spring Creators Update. I can't remember what the context was for me talking about it last week or the week before, but there was something that they were going to be doing with this thing called the Spring Creators Update. I don't remember now what it was. But they're now saying that they're planning for what I would call the "system unusable time" to be nothing more than about half an hour. So it's not gone, but they're trying to squeeze down to 30 minutes.

And it's not that they've produced some miracle. What they've done is they have been progressively moving more of the work of performing one of these major updates, which essentially it's a hidden reinstall of Windows. I mean, it is a whole new Windows that is coming in and then moving everything over from the old Windows into the new Windows and saying, okay, here we go. So what they've been doing is they've been moving more and more of that into the so-called online phase, where you can still use your computer. And they're saying that it's in the background with low priority, so it won't interfere, it won't make things sluggish while it's doing it. It's not like running cryptocurrency mining in your machine. But as a consequence, the total time required has actually inflated significantly because they can't commandeer, they just don't want to commandeer your machine in this offline phase for as long as it would take.

**Leo:** They do something sort of like this right now. When you install Microsoft Office, it continues for a while in the background. But it might let you run Word partially.

**Steve:** Well, and remember, even booting. You know, famously, they were under - it took so long to boot [crosstalk].

**Leo:** No, you can't do anything. But look, you're seeing the desktop.

**Steve:** Look, oh, here's your desktop, exactly. It's not quite here yet. Things are still popping in, you know, populating. And it's like, uh, wait a minute, what?

**Leo:** It's ironic because the initial installs are actually much faster now because they use images, and they blast them [crosstalk].

**Steve:** Right, right.

**Leo:** It's the updates. And if you think about it, an update is a nontrivial thing to do.

**Steve:** Oh, I don't even want to - like I'm glad that's somebody else's problem because that's not a problem anybody wants to have.

**Leo:** I wonder, though, if you could just do a diff, you know, just take the fully - but you can't because every system's different, I guess.

**Steve:** Right.

**Leo:** That's the problem. If you could say, oh, just take - if you had a known system in the diff, and you just applied the vectors, the changes, even if it was nonsensical, it'd be just a little bit here, a little bit there, you could do it. But the problem is that doesn't work, yeah, because you don't know the starting point.

**Steve:** Right.

**Leo:** Oh, man, I don't know how they do it. That must be some genius's full-time job.

**Steve:** Oh, and when you look at how many files there are, I mean, the whole Windows Update thing, the idea that you can go in and remove one that you don't like? What?

**Leo:** I always think of it as - the analogy I use on the radio show is, you know, you've seen the magician who whips the tablecloth out, and all the silverware and the plates are the same. Well, this is doing that, and then putting a new tablecloth in. Right?

**Steve:** That's good.

**Leo:** You're whipping it out, and then you're putting a new one in without disturbing anything. Good luck.

**Steve:** And in the process you change the forks to chopsticks.

**Leo:** Just 'cause. Just 'cause you can.

**Steve:** Yeah.

**Leo:** It's spring, let's put in a yellow tablecloth today.

**Steve:** That's right. So we know well the lesson that you know nothing about security until it's been tested. I've used the example for years about Ballmer prancing around the stage before the release of Windows XP, declaring before its release that this was the most secure operating system Microsoft had ever produced. And I was here then. And it's like, uh...

**Leo:** No.

**Steve:** No. That's not something you can declare. You can want it to be true, but history is the judge of these things. So that is to say, only when something has been tested can you affirmatively say, I mean, and even then you can't say we know it's true. But you can say smart people tried to break it and couldn't. So that's a lot better than if that never happened, if no one tried to break it, and everyone's just saying, oh, yeah, Ballmer's entertaining, so let's all use his operating system.

So we had an independent organization, vpnMentor.com, that contracted with three external ethical hackers, one who wished to remain completely anonymous; one, well, he might as well be anonymous because his handle is "File Descriptor," and I don't know who that is, so he's anonymous; but Paulos Yibelo is the third guy, who was the manager of this VPN testing team. The File Descriptor guy is a reputable ethical hacker working for Cure53, who is a company we've referred to in the past. We've covered some of their work. And that was the company hired by TunnelBear to identify and fix issues with their VPN applications. And Cure53 is a leading security research firm. Paulos has previously found vulnerabilities in popular VPNs and published them in the past, and his work has been covered in ZDNet, Slashdot, and other places. So these guys are real.

Well, they took a look at three VPNs - Hotspot Shield, PureVPN, and ZenMate VPN - and found significant, readily exploitable IP leaks, that is, IP address leaks in all three.

**Leo:** The user's IP address?

**Steve:** Yes, yes.

**Leo:** That's not good.

**Steve:** Readily obtainable. Now, we know that you don't use a VPN necessarily for complete anonymity.

**Leo:** As you shouldn't. That's not what a VPN's supposed to be, yeah.

**Steve:** Right. That's what TOR promises more than a VPN. It is the case, though, a VPN gives you encryption. And you get some anonymity because your traffic is going to emerge from an endpoint at your VPN host server out onto the Internet. And hopefully, you know, and so the point is that at that point the packets are being rewritten with - your source IP is rewritten with that server endpoint source IP so that, when they go out to a server, the IP address that the website that you're visiting sees is the VPN server IP. The packet then goes back. The VPN server that's been keeping track of this rewrites the destination from itself back to you, and then the packet goes through the tunnel and so forth.

So it's sort of wrong to expect really strong anonymity from a VPN. But neither should it have really bad, readily exploitable problems. And all three did. Of the three organizations, only Hotspot Shield responded promptly and the way we would want them to. They had three problems, but they just - immediately they stepped up, worked with the security researchers, recognized, acknowledged, and immediately fixed the problems.

At this point, today, that has not happened with PureVPN and ZenMate VPN, both of whom have outstanding problems they've been informed about and haven't resolved. So as we know, mistakes happen. Anybody can make them. I think it's significant that Hotspot Shield said, okay, we're on this. It would be nice if Hotspot Shield had themselves contracted for a security audit, the way for example TunnelBear did, get themselves audited and be able to say, hey, you know, we had people check us out, and we passed. But the mistakes are interesting.

So there were three problems in Hotspot Shield, all now fixed. One of the things that Hotspot Shield, a feature, is a Chrome extension to make it just sort of like, oh, look, it works, easy drop-in. Well, no, that's not the way I'm going to run a VPN. I'm going to run a VPN by installing a client, like OpenVPN, and then using it to establish a tunnel, and then everything in my system goes through the tunnel. But I get it.

**Leo:** I'm surprised these companies don't use OpenVPN, actually.

**Steve:** I know. I know.

**Leo:** Boy, that's weird, isn't it. You think they use a custom-built VPN?

**Steve:** They might well use an OpenVPN protocol. But in this case it was a Chrome extension that had a problem. There was a proxy autoconfig script. And it turns out that the proxy autoconfig script was looking for a query parameter in the URL that was act=afProxyServerPing. And if that parameter was in the URL, it routed all traffic to the proxy hostname provided by that parameter. Which means any site could give you a URL, like that you click on, and your traffic is hijacked, and all of your VPN traffic runs through this hijacking site. So, whoops, not what you want from a VPN.

Also they noted that the DNS lookup was using the system's DNS, which that's an often-made mistake. I've referred to this for years. Not all VPNs capture all of your Internet traffic, that is, some of them just capture your TCP connections or your web surfing

connections, saying, oh, yeah, real easy VPN, whatever. But the DNS lookups don't go through the tunnel. Well, that was also the case. And the problem with that is that DNS leaks your IP. I of course have the DNS Spoofability page where GRC pretends to be a DNS server for the user, and I end up being able to check the action of the DNS server. But it's possible to leak all the way back to your IP.

There's a site, www.dnsleaktest.com, which operates very similar to mine. In fact, I was wondering if they were using mine, like proxying my page because it looks - they copied it, essentially, sort of a subset of its behavior. But it demonstrates that it is very easy to determine who anybody's DNS servers are. I mention this to our listeners because this is a test you can easily perform. If you're a VPN user, fire up your VPN and then go to www.dnsleaktest.com and see whether it shows the same DNS servers, that is probably your ISP's DNS servers, or whether it's showing you the DNS servers belonging to your VPN provider, which is what you would want. You would want for the DNS queries generated by your system to be tunneled through the VPN also and not to just ignore it and make a standard system-level DNS query, which can easily and readily leak your IP.

And then the last thing they found was, again, in this proxy autoconfig, a very broad whitelist. And just because I thought some of our listeners might get a kick out of it, I have it in the show notes here. But, I mean, it's amazing what is on the whitelist that bypasses the proxy, so much so that, for example, localhost dot anything, if localhost appears in the domain name, it bypasses the VPN. So, like, localhost.evilsite.com, and you're not VPNed.

So it's like, whoops, again, all these have been fixed. It's worrisome that there were these problems. But this is a perfect lesson of you really just can't assume something is secure until somebody has looked at it, I mean, somebody who knows what they're doing has tried to find problems with it. And this is why, going back to these bounties, why I'm really happy to see Microsoft and Intel creating some significant cash windfall bounties for hackers, to motivate hackers to try to find more problems. When we look, we find problems. And it's not until you have really, really tried hard and found nothing that you can begin to say, okay, this seems good. So anyway, people using these other two VPNs, I would look for updates hopefully coming from them of their fixes. And Hotspot Shield has immediately stepped up and fixed the problems that they were having. So hats off to those guys.

And I did want to just touch on, although this is not a Security Now! topic, just on last night's news of a pedestrian death, just because it's a novel event. This is the first pedestrian death involving a self-driving car. It occurred in Tempe, Arizona at 10:00 p.m. last evening. There was a driver, a so-called "backup driver," in the auto. It was an SUV owned by Uber as part of their self-driving autonomous vehicle project. There was, remember, what was it, in 2016 Tesla had a problem. Their autopilot software, it was considered, contributed to the death of a Tesla driver, his name was Joshua Brown, although he had ignored repeated warnings to put his hands on the wheel and didn't. So he certainly had some responsibility there.

In this case it's not looking like, I mean, this just happened. So camera footage is being checked, the Uber's self-driving software I'm sure is black boxes and green boxes and red boxes, all the boxes are being probed. But the video footage that's been seen, the driver's report, nobody was impaired. The driver wasn't, you know, the backup driver was not impaired. Neither was the pedestrian. But a 49-year-old woman stepped out with her bicycle in the middle of the block, directly in front of the car. The feeling was from the police reports that anybody, like a human driver driving along, it was a 35-mile-an-hour zone, the car was going 40 miles an hour, so about as much over as we all typically drive. It'll be interesting to see who decides where fault is.

But I really liked what Andy had to say at the end of MacBreak Weekly, too, Leo. I agree with him that downstream I wouldn't be surprised - it's going to take a couple generations, maybe a generation, 20 or 30 years. But I wouldn't be surprised if ultimately we get this stuff nailed. And it doesn't look like it was a technology fault. It looks like something...

**Leo:** I don't think you can blame the car at all.

**Steve:** No.

**Leo:** I mean, if somebody, I mean, I don't know what happened. We're just going by the police report. And the police were the ones who said - the Tempe Police said this couldn't have been avoided even with a human driver. But first of all, I'm surprised it wasn't going the speed limit. That, they should be dinged for that. Autonomous vehicles should travel exactly the speed limit, and maybe even a little less. But even going 35 miles an hour, if somebody jumps out in front of you, there's physics, you know.

**Steve:** Yes.

**Leo:** Autonomy does not overrule physics. It's not even reaction time. The reaction time of an autonomous vehicle should be instant. It's physics. You can't stop. So I don't think you can blame - but of course this will go against the account of autonomous vehicles, which is in a way a shame because I don't think it should.

**Steve:** Well, maybe if it introduced another note of caution, that might not be a bad thing.

**Leo:** Sure.

**Steve:** Just, you know, it's like, let's slow ourselves down a little bit.

**Leo:** What are you going to do? What would you do in the software except go down, go slow down.

**Steve:** You're right, that's a very good point. That's a very good point.

**Leo:** What would you do to fix it? You can't fix that.

**Steve:** Yup.

**Leo:** Can't fix humans, that's the problem. You can fix a machine. You can't fix a human. And you know what, it's going to happen. Sometime a kid's going to - I was telling Lisa this when we were talking about the story yesterday. Some kid's going to chase a ball out from behind a car and get clobbered by an autonomous vehicle. Now, it happens all the time by human-driven vehicles. But when it's a robot, that's going to be a problem.

**Steve:** Well, it's very much like how upset we get when an airliner crashes.

**Leo:** Right.

**Steve:** But we're not as upset by the distributed deaths occurring on the highways all over the country every day. I mean, the famous line is that statistically it's far safer to travel by air. Nobody really likes that because it takes...

**Leo:** But humans are not rational. That's the real problem.

**Steve:** Right.

**Leo:** We're not. And this is why I'm kind of stepping up to defend this robot vehicle.

**Steve:** I think you're right, Leo.

**Leo:** Because it'll be a shame if we don't get autonomy because of an irrational fear of autonomy.

**Steve:** Right. Right. And we won't talk about Terminator and Skynet because...

**Leo:** Well, it's built into the human psyche. We're afraid of these things, yeah.

**Steve:** That's another problem, yeah. Okay. So I just wanted to give our listeners an update on where I'm spending still full time, and that is on SQRL. Something I've been meaning to talk about since I mentioned it maybe a month ago, I called it the "sensitive data monitor." I did a number of things - there are a number of things now looking back I wish I had done differently. Not many, but like internal architectural stuff that turned out to not evolve as well as I wanted.

But one of the things I love is that every single string of text is in a separate file with a numerical number, meaning that it's super simple to change the wording of anything, and of course to make it language-specific. That, I nailed that. Also because it needs to be language - I wanted to give it the opportunity of being multilingual. Different languages require different amounts of space. So I always built in a scaling to the UI. Well, that turned out to be a win because of this DPI awareness that we need now

because our screens have gotten such high resolution that, if the app didn't scale itself to the resolution of the monitor, it would be too small on high-resolution monitors.

And in fact the InSpectre app doesn't have this technology, and I've had complaints from people saying, "I can't read it," because they've got 4K monitors, and the thing's the size of a postage stamp on their monitor. It's like, well, I'm sorry. Get a magnifying glass. But anyway, so I got that right. The other thing I got right from day one, and I'm only saying this because I want other people writing secure code to hear this, mostly what I'm about to say next, but the other thing I got right was creating containment for everything that was sensitive. That is, there is one single location where everything that is sensitive ever resides. And I created it upfront so everything I ever did was in that container. And that has turned out to be the biggest win, of course, because it is easy to lock it in memory and to wipe it. And so that ended up being another really - I'm very pleased with that design decision.

But something I did about a month ago I wish I had done at the beginning. And so anybody who is going to write SQRL components, we've got clients and server-side stuff in the works. There's a Java side. There's a JavaScript. We've got browser plugin. Android and iOS clients are on the way. So a lot's happening.

About a month ago - and I mentioned this on the podcast, but I never followed up - I created a window which I called the "sensitive data monitor." And it is a window that dynamically shows the content of that sensitive data area, and labels it, and formats it so I can see it. And as a consequence of that - I had always intended to do this at the end of the project. Now it's ended up being so useful, I wish I had always had it. And that is that, while you're - and this is only, of course, for R&D mode and the developer edition, where it logs all of the guts of its crypto in order to help other developers confirm their internal operation, and it logs all kinds of stuff. It also shows you the contents of the sensitive data area.

And as a consequence of that, not only of my seeing it, but also letting the people who are testing the client have that, we found some edge cases where, if you did this, and then you did that, and then you did this, then, oh, look, a key that should have been proactively zeroed didn't get zeroed in that particular case. And as a result, I then have been subsequently far more aggressive in proactively wiping things.

But then I caused myself some problems because I was overwiping. I was like, I broke some things by removing information that was necessary, which is why I wish I had done it from day one because then, as I was moving through the development of this, I would have always had my eye on that, making sure that when I was done with some data, it was immediately eliminated. And again, I'm being over cautious. Nobody else, I mean, evidence demonstrates that people are leaving stuff in RAM all the time. So I'm wanting not to make that kind of mistake. But I just wanted to say to anyone else developing security-aware software, doing this, just having it there onscreen while you're working with it is incredibly useful.

And I'm almost done. I fixed a problem with SQRL running in Sandboxie, which now it works just perfectly. Some people didn't want to use the screen darkening. SQRL has a UAC-style screen darkening to demonstrate that you're using SQRL to keep web browsers from being able to spoof SQRL. And when we turned that off because some people didn't want it because it also breaks copy and paste, which is good because you don't want that for your password, but some people are, you know, they're wanting that. And so I allowed screen darkening to be disabled, but that created a keyboard focus problem which I'll be fixing probably tomorrow.

And then it's, like I hesitate to say because something that's this big and involved doesn't - it never ends instantly. But I will then turn my attention to bringing up the web forums which I'm going to have online before the formal announce because there's no way I can handle the consequence of all of our listeners grabbing it and starting to play with it. We need someplace for people to go to self-help and answer questions and so forth. But it's very close to being time for me to come up to Petaluma, Leo, and sit down with you and maybe Father Robert, and go through this.

**Leo:** Just remember, I'm going out of the country April 19th through May 5th. So either before then or after then. Well, Robert could do it. You don't have to have me here.

**Steve:** Oh, but you're a perfect foil. You're wonderful. [Crosstalk]. What does that do? What does that button do again?

**Leo:** What the hell's that for? Hey, this just in. Expedia is announcing that 880,000 payment cards have been exfiltrated by bad guys from their site.

**Steve:** Darn.

**Leo:** Yeah.

**Steve:** Well, what was that sponsor you were just telling us about, Leo? Capital One?

**Leo:** Yeah, everybody's got to use Capital One. Between January 1st of 2016 and December 22nd of 2017 the breach for its partner platform, and then between January and June for its consumer platform. Phone numbers, names, email, billing addresses. Orbitz was not affected, but Expedia was.

**Steve:** Boy.

**Leo:** Anyway, just, you know...

**Steve:** Yup, it happens.

**Leo:** It happens again, and again, and again.

**Steve:** The good news is, if the world chooses to adopt SQRL, we will never again have a theft of passwords. That's just gone because SQRL gives websites no secrets to keep.

**Leo:** Yup.

**Steve:** Which is nice. I got an interesting note from an anonymous sender. He used our web form for sending feedback. And the subject was "SpinRite Story in 652" - so, what, three weeks ago - "could have been social engineering." And his submission was dated March 7, so a couple weeks ago. He said: "Hi, Steve. I just caught SN-652, listened to the SpinRite story." And so then he quotes a dialogue, or he makes one up.

"Customer?: I lost my copy of SpinRite. I'm incapable of recordkeeping. I have no backups. I'm an unorganized person. By the way, the email I bought SpinRite with no longer works <wink>. Here's my new email address. Can I have a new copy, please?"

And so he quotes GRC Support saying: "No problem. We don't need to verify anything. We trust you. Here's a link to the download."

**Leo:** Isn't that nice.

**Steve:** "Feel free to change your contact information. We'll just email you the secret PIN in plaintext."

**Leo:** You shouldn't announce this on the air, Steve.

**Steve:** The customer responds: "Really? Yoink! If anyone asks, I was never here. And, no, I am not wanted in five states for fraud." So then this guy says: "How was this different from someone calling your cell carrier or your bank and claiming to be you?" He says: "This could have been social engineering. It is often said the customer is always right. But what if they aren't a customer?" Anyway, and then he says: "Looking forward to the end of SQRL" - which as I just said is approaching rapidly - "and the beginning of SpinRite 6.1," which is also approaching rapidly.

So I just wanted to say we have a database dating back, well, actually it's in FoxPro.

**Leo:** That tells you something. Almost dBase II.

**Steve:** Actually, it was in dBase II, and we updated to FoxPro, running in a DOS box.

**Leo:** Wow.

**Steve:** And it's got SpinRite 1 customers in it. So I'm sure that Sue did look up this person's name. And she does some due diligence. But at the same time, let's remember who we are. I mean, the company's policies are an extension of mine. And we often share stories of somebody who let their friend run their copy of SpinRite. Last week we said some guy fixed his dentist's RAID array and asked his dentist to buy a copy, and he believes they did, as a way of thanks and sort of payment for the services that the software rendered.

**Leo:** Nice, yeah. That's the right thing to do.

**Steve:** That's the way to be. I remember back when there was competition, there were other hard disk utilities. Every single license said "can only be run on a single hard drive." And that just never seemed reasonable to me.

**Leo:** You're just not greedy enough, Steve.

**Steve:** It's just not practical. Well, and that, too. When I do something like InSpectre, everyone says, oh, just charge a dollar, charge a dollar.

**Leo:** Yeah.

**Steve:** Well, yes, I would have made $547,000 or something. But on the other hand, many fewer people would have actually used it. So I think my policy is right. Greg and Sue are gainfully employed. I get to have cabernet. And I've had time, thank you everybody, to develop SQRL and offer it to the world. SpinRite 6.1 will follow, and we'll go from there.

**Leo:** And this show, yeah.

**Steve:** And the show, exactly.

**Leo:** We'll send you some - we have a new wine sponsor, and we have a wine from them that I - because I know you love cabs.

**Steve:** I do.

**Leo:** I think you might really like - we're going to send a couple of bottles down, I think.

**Steve:** Cool.

**Leo:** That was Lisa's idea. So that'll make up for that lost revenue.

**Steve:** I don't feel it's lost.

**Leo:** No, I agree, I agree.

**Steve:** I mean, our customers, so many of our listeners have said they bought a copy of SpinRite to support the podcast, and then to support my efforts. And hopefully they're getting some value in return. It's not a charity, it's like here's some software. And the other thing I'm not doing, you notice, I mean, people complain it hasn't been updated in

a long time. That's true. On the other hand, I'm not dinging people for upgrade fees every year, trying to milk their loyalty to SpinRite.

**Leo:** That's a lot of the reason things get upgraded. Not for real value...

**Steve:** Exactly.

**Leo:** Just because then they can charge you.

**Steve:** Yes, exactly.

**Leo:** It's an annuity. No, that's what we love about you, Steve.

**Steve:** Well, anyway, I got a kick out of him saying, "That guy could have been making it up." And it's like, yes, he could have. And, you know, fine. I'm not going to lose any sleep over it. We're doing fine, and I appreciate all the listeners who are honest.

Closing the loop, two things. A Kari Mattsson said: "Any comments on this?" and then quoted: "EV certificates under DigiCert's root CA without human intervention." And he or she provided a link to TrustCubes.com/ev-ssl and asked the question, "Shouldn't EV certs be more scrutinized?" And that's interesting because it's a great question. The answer is no, if the scrutiny is placed where it should be. For example, I have loved the fact, and I've mentioned this on the air, that I've been able and can issue my own EV certs with DigiCert. I've done it in the middle of the night when I've needed something. And I love that fact. Where the scrutiny needs to be applied is on me and my account are GRC. That's the identity binding that needs to be absolutely verified.

And what's cool is, independent of my certificate recycling and renewal, I will get contacted by someone from DigiCert saying, hey, we need to reverify you for EV status. Which I think is brilliant. So what they've done is they've decoupled the timing of the identity binding to the account from the certificate issuance, allowing the account holder to issue EV certificates whenever they need to, and then while at the same time not degrading the integrity of the process because they are asynchronously making sure you're still you. So I loved that question, and I just wanted to note that there is a way that you can get the best of both worlds, and that's the service that DigiCert in my experience alone has been offering me, and I really appreciate it.

And then I love this one other tweet from Itinerant. Itinerant? Yeah. Wait, is that the way you pronounce it?

**Leo:** Yeah, itinerant, yeah.

**Steve:** Itinerant. For some reason that just seemed wrong to me. Itinerant Engineer. Anyway, I love this. He tweeted both to me and to SwiftOnSecurity the observation: "Perfect Forward Secrecy" is the "I guess you just had to be there" of encryption. And of course that's exactly what it is.

**Leo:** And somebody just bought a copy of SpinRite.

**Steve:** And somebody just bought a copy of SpinRite. Thank you.

**Leo:** Thank you. All right, Steve. On we go with Pwn2Own. I'm going to get my sportscaster voice ready here.

**Steve:** Yeah, we've got to - the way they wrote up the blow-by-blow, I just thought you'd have - our listeners would get a kick out of you running through...

**Leo:** I don't know if I have a sportscaster.

**Steve:** You've got a million voices, Leo.

**Leo:** I'll find one.

**Steve:** Bottom line is that China was sorely missed. China's hackers routinely win at Pwn2Own, typically sweeping the board. And notably, we've talked in previous years, the Tencent and the Keen teams have been the big winners in the last couple years' competition. In last year's competition of Pwn2Own the top five winners were from China, and three of them were from the Tencent group. And as I mentioned when we were on the front side of Pwn2Own, the Chinese government has decided that they're going to keep their hackers at home. And so there were no Chinese entrants this year, and it showed.

There are potentially many bugs and cash prizes to be had. Last year, which was the event's 10-year anniversary, since the first one was 2007, the Zero Day Initiative awarded a total of $833,000 to white hat hackers, 833, 833,000 to white hat hackers, exposing 51 different zero-day bugs. And most of those were found by Chinese researchers. This year, by comparison, despite pulling together a purse of two million available dollars, this year's hackers took home only $267,000 out of that total of two million, despite the fact that it was a target-rich environment.

Oracle's VirtualBox, VMware Workstation, Microsoft Hyper-V Client, Chrome, Safari, Edge, Firefox, Adobe Reader, Microsoft Office 365, Microsoft Outlook, NGINX, Apache Web Server, Microsoft Windows SMB, and OpenSSL, all of those were valid targets for exploitation. And despite the fact, very few of them were exploited. Mostly the browsers took a hit. Actually, it was browsers and in one case VirtualBox. And $267,000 in total this year, despite the fact that the largest prize was a quarter million dollars for a sandbox escape from Hypervisor to the kernel under Hyper-V.

**Leo:** They had one of those last year.

**Steve:** No takers. Yeah.

**Leo:** Isn't it possible, though, that this software was more secure, like these things had been patched? Or do you think the Chinese team just has all these exploits and no one else does?

**Steve:** I think that they're good. I think that...

**Leo:** I think this is bad because now we don't know what they can do.

**Steve:** Oh, it's definitely bad, Leo. It's bad.

**Leo:** I mean, they're not exactly friendly adversaries.

**Steve:** Unh-unh. No. And also the timing was a little unfortunate because several non-Chinese competitors dropped out at the last minute after the immediately previous day's Patch Tuesday, which closed some holes they'd been planning to exploit.

**Leo:** Oh, man.

**Steve:** So it was like, whoops. Yeah.

**Leo:** So I have some misgivings about this whole thing because people hold onto exploits for a whole year instead of...

**Steve:** Yes, yes, yes, for the competition.

**Leo:** ...responsible disclosure.

**Steve:** Yes.

**Leo:** They're trying to make money. Now, it says here that the Pwn2Own, the CanSecWest people buy these exploits. What does that mean?

**Steve:** Yeah, and it's weird, too. And that's the language that they use is that they purchase...

**Leo:** Does that mean they hold onto them?

**Steve:** Yes. Well, no, it means that they purchase them. The Zero Day Initiative is run by Trend Micro. So Trend Micro protects their users immediately as a benefit of using

Trend Micro's antimalware products. And then they do then responsibly disclose them to the vendors of the products.

**Leo:** Right away.

**Steve:** So they're not - yes, yes, right away. So this year's winner succeeded in two instances. And you'll be reading his name in a second, Richard, and you pronounced it correctly.

**Leo:** Zhu.

**Steve:** Zhu won the contest with 12 points for hacking Edge and Firefox and took home, out of that 267,000, 120 of that. But anyway, I thought our listeners would get a kick out of you sort of taking us through the competition.

**Leo:** This is a lot. I'm only going to do a few paragraphs. I'll let you do the rest.

**Steve:** Okay.

**Leo:** "Day One: The first day of Pwn2Own 2018 has come to a close, and so far we've awarded $162,000 USD and 16 points toward Master of Pwn." Oh, I love this. "Today saw two successful attempts, one partial success, and one failure. In total, we purchased three Apple bugs, two Oracle bugs, three Microsoft bugs," and a partridge in a pear tree. No, I added that. "The day" - oh, I love this. This play-by-play is pretty funny.

**Steve:** Yes.

**Leo:** "The day began with Richard Zhu (@fluorescence) targeting Apple Safari with a sandbox escape. Unfortunately, he couldn't get his exploit chain working within the time allotted due to a failure in the heap spray technique." Oh, no. "Despite this, the bugs he brought to the contest were certainly interesting and were purchased through the regular ZDI program." I guess that's what we were talking about. They gave him money even though he didn't do it right, or it didn't work.

**Steve:** Correct.

**Leo:** "Undaunted, Richard later returned to target Microsoft Edge with a Windows kernel elevation of privilege exploit (EoP), bringing a flair for the dramatic. After his first attempt failed, he proceeded to debug his exploit in front of the crowd while still on the clock. His second attempt nearly succeeded, but the target blue-screened just as his shell started. Oh, Richard! But his third attempt succeeded with one minute and 37 seconds left on the clock." How long do they give them?

**Steve:** Ten minutes.

**Leo:** Ten minutes? Whoa. "In the end, he used two use-after-free bugs in the browser linked to an integer overflow in the kernel to successfully run his code with elevated privileges. Nice going, Richard!" That's you; right? You wrote that.

**Steve:** Uh-huh. I did.

**Leo:** "His dramatic last-ditch effort earned him $70,000 and seven points toward Master of Pwn." All right. That's quite enough of that. Quite enough of that. But that is, it is dramatic.

**Steve:** Oh, it is.

**Leo:** Have you ever watched one of these?

**Steve:** No, I have not.

**Leo:** Be kind of fun.

**Steve:** Yes.

**Leo:** I take it they have an audience.

**Steve:** Yup, they have a bunch of people watching. So it's done in front of a group. Actually, I misspoke. It's 10 minutes average, so it's three attempts in 30 minutes.

**Leo:** Oh, half an hour, I mean, you know.

**Steve:** Yeah, yeah, yeah.

**Leo:** Not a lot of time.

**Steve:** Yes. And so essentially...

**Leo:** I love him debugging it on the fly.

**Steve:** Yes. So, like, you know, the clock is ticking, and he's like, oh, crap, why didn't that work? And so these guys know their stuff.

**Leo:** That's pretty good. Pretty good.

**Steve:** Yeah. So anyway, essentially there were several Apple Safari escapes. All of them were elevation of privilege exploits. There was also one found in Edge that we just talked about, and one in Firefox.

**Leo:** I'm going to read - I'll read you the next one because it's too good.

**Steve:** Oh, okay.

**Leo:** I'm sorry, I'm just looking at it.

**Steve:** It really is, Leo.

**Leo:** "Up next, Niklas Baumstark, or @_niklasb, from the phoenhex team targeted the Oracle VirtualBox. Apparently not one for added intrigue, his exploit immediately popped not one, not two, but three different calculator apps to rub his success into Oracle's face. His demonstration qualified as a partial success as he used an out-of-bounds read and a Time of Check / Time of Use to still earn him $27,000 and three Master of Pwn points. It was a great demonstration, and we look forward to more of Niklas' research in the future." What is he, 12? Probably; right? This kid's going somewhere, folks.

"The final attempt on Day One saw Samuel Gross, or @5aelo of phoenhex, targeting Apple Safari with a macOS kernel elevation of privilege. Last year his exploit involved a Touch Bar component, and this year proved to be no different, folks. Sam deftly leveraged a combination of a Just-In-Time optimization bug in the browser (whoops, Apple!); and then a macOS logic bug to escape the sandbox; and, finally, a kernel overwrite to execute code with a kernel extension and successfully exploit Apple Safari. This lovely exploit chain earned him $65,000 and six points toward Master of Pwn. And as he did last year, he left a message for us on the Touch Bar once he was complete." What was the message? They don't say.

**Steve:** I know.

**Leo:** Something like FU, Apple; right? Take that. Holy cow. He leveraged the Touch Bar.

**Steve:** Yes, yes.

**Leo:** Unbelievable. Unbelievable.

**Steve:** Yes.

**Leo:** If you're one of these big companies, you've got to look at this day and go, oh, boy. Is there video? Wait a minute. There seems like there might be video.

**Steve:** Yeah, there are videos of this happening.

**Leo:** Of them doing this.

**Steve:** Yeah. On the second day, it says: "The day began with the return of Richard, this time taking aim at Mozilla Firefox with a browser kernel elevation of privilege. He eschewed all drama today and successfully popped Mozilla Firefox on his first attempt with his clever use of an out-of-bounds write in the browser, followed by an integer overflow in the Windows kernel, to earn himself another $50,000 and five more Master of Pwn points..."

**Leo:** I want to go next year. I want to see this.

**Steve:** "...to bring his event total to" - yeah, yeah.

**Leo:** Look, and he gets like a drone trophy. No, it's a bug.

**Steve:** And he's very happy.

**Leo:** That's funny. Oh, that is so funny. Good for Richard Zhu. Yeah. Well, you know what, I'll extend this offer to CanSecWest. If they want me to do play-by-play next year, I'm available.

**Steve:** Well, that was great. So the follow-up news is, and I don't know what this means: DEF CON China is May 11, 12, and 13 coming up. And so it's going to be held in Beijing. And I don't know what this means in terms of the Chinese government policy relative to having a security show of that sort. Maybe if it's within China's borders they'll allow their hackers to attend. Because it's a standard-looking DEF CON layout and format and presentation.

**Leo:** Well, I think that's the issue. No hackers want to come to the U.S. anymore. You can get arrested.

**Steve:** I know.

**Leo:** Isn't it ironic that people will go to China? It's safer?

**Steve:** Yup.

**Leo:** Here's the message, by the way. This is a picture of the Touch Bar. It says "Pwned by 5aelo," and a happy face.

**Steve:** Nice.

**Leo:** That's cool. And all the codes above it. No fake code there.

**Steve:** So, my friend, that's our podcast.

**Leo:** Our time is up for this fabulous episode of Security Now!. Thank you, Steve Gibson. We do this show every Tuesday, round about 1:30 p.m. Pacific, 4:30 Eastern, 20:30 UTC. If you want to watch live, you can, at TWiT.tv/live, by the way. And if you're going to do that, you might as well get in that chatroom there, where there are some fabulous people chatting along and providing us with images and all of that stuff: irc.twit.tv.

Now, Steve has the on-demand audio of the show, and he has something unique at GRC.com: transcripts. Elaine Farris will be listening to this and trying to write it down. And then those handmade transcripts are great, though, for searching. So if you hear something, or you want to go back through one of the previous episodes - do you have transcripts for every episode?

**Steve:** Every single one. She didn't start at the beginning, but after a while it was proving so useful that I said, okay, let's go back and start from number one.

**Leo:** So you could search for "honey monkeys," it would take you right to number one.

**Steve:** Yup. Got it.

**Leo:** Yeah, I don't think we've mentioned them since. We have audio and video, oddly enough, of the show. You can get that at TWiT.tv/sn. While you're at Gibson Research Corporation, Steve's site, GRC.com, don't forget to pick up SpinRite, right, the world's best hard drive recovery and maintenance utility, and check out all the other stuff going on over there. It's a beehive of activity. Steve wraps up SQRL, gets back to SpinRite 6 and all sorts of other fun projects. InSpectre is still there. Actually, you probably do want to get InSpectre these days, right, see what CPU ID you have.

**Steve:** Yeah, exactly, in order to see whether you've got new microcode available.

**Leo:** That's GRC.com/InSpectre, spelled R-E. Or, you know, if you Google "InSpectre" you'll find it right away. It's number one.

**Steve:** Yup.

**Leo:** What else? I guess that's about it except thank you, Steve. And we'll be back next week, I hope you will, too, for another gripping edition of Security Now!. Bye-bye.

**Steve:** Thank you, Leo.