SECURITY NOW!

Transcript of Episode #840

0-Day Angst

Description: This week we look at Microsoft's decision to finally disable Excel's legacy XLM by default, but not for everyone. We look at Google's warning sent to more than 14,000 of its Gmail users and at their move toward enforced two-step verification. We look at recent hacking and ransom payment legislation and at last week's massive breach at Twitch. We cover the emergency Apache web server update and the mass exodus from WhatsApp during last week's Facebook outage. We look at new Windows 11 side effects and at Patch Tuesday. We close the loop with some listeners, and I quickly update on SpinRite's progress. Then we settle down to consider the true significance and import of the various year-to-date zero-day counts.

High quality (64 kbps) mp3 audio file URL: http://media.GRC.com/sn/SN-840.mp3
Quarter size (16 kbps) mp3 audio file URL: http://media.GRC.com/sn/sn-840-lq.mp3

SHOW TEASE: It's time for Security Now!. Steve Gibson is here. It's Patch Tuesday. Steve will have the details, including why you might want to disable macros in Excel. That again? We'll also talk about, get Steve's take on the big Twitch hack. And zero-days, Apple is not exempt. You'll be surprised how many they've had this year alone. It's all coming up next on Security Now!.

Leo Laporte: This is Security Now! with Steve Gibson, Episode 840, recorded Tuesday, October 12th, 2021: 0-Day Angst.

It's time for Security Now!, the show with Steve Gibson, the man in charge, the explainer in chief; the show that protects you, your loved ones online, your privacy and security. Good day, Steve. Good to see you.

Steve Gibson: Hello, Leo.

Leo: You did a great job, and everybody should listen to your Fireside Chat on Friday. Thank you for doing that. I really appreciate it.

Steve: I was happy to. It was fun. Gave me an opportunity to air some of my past.

Leo: War stories.

Steve: My war stories from the past, yeah.

Leo: Fantastic. And if you are a member of Club TWiT, it's on your TWiT+ feed. Big show today?

Steve: Yeah. This started out as a discussion of another company's run of zero-days. And as I sort of was fleshing it out more, I'm getting a little more philosophical about, you know, are we really needing to be that worried about them? I thought, you know, this - and I didn't already have like a dominating topic. We've got lots of news to talk about, so a jam-packed show. But this is Security Now! #840 for October 12th. And I titled it "0-Day Angst" to sort of just get a little philosophical after we cover the background facts about what it means, the whole issue of exploits that are found in the wild and how much danger they really represent to typical people.

But we're going to first look at Microsoft's decision to finally disable Excel's legacy XLM macros by default. But not for everyone. We look at Google's warning sent to more than 14,000 of their Gmail users. And also at their move toward enforcing two-step verification. We talked about this back in May when they said, we're going to make everybody use multiple factors of authentication. To which I said, eh, good luck with that.

Leo: Yeah. I already got a call on the radio show from somebody baffled by the whole thing.

Steve: Oh, goodness. What could possibly go wrong?

Leo: Oh, god.

Steve: We're going to look at recent hacking and ransom payment legislation which is, like, moving through Congress with surprising alacrity; and at last week's massive breach of Twitch. We cover the emergency Apache web server update and the mass exodus from WhatsApp during last week's Facebook outage. We look at new Windows 11 side effects. And of course this is Patch Tuesday, which we don't have anything to say about yet because the action all begins now, and we'll be talking about it next week.

We close the loop with some listeners, and I quickly update on SpinRite's progress. Then we're going to settle down to consider the true significance and import of the various year-to-date zero-day counts, what they mean for us. And of course we have a fun - actually, we talked about this briefly in passing previously. But I decided, okay, now's the time to deploy this fun Picture of the Week. So I think a great podcast.

Leo: Oh, I'm excited. Back to you and your magic Picture of the Week.

Steve: So anyway, this was just one of, as we were saying, one of the synthetic O'Reilly covers. This book, if it were real, titled "Web Development With Assembly," as in assembly language. And then sort of the subtitle they have actually above that on the book, it says: "You might as well just kill yourself right now."

Leo: Oh, my god.

Steve: And you know how all the O'Reilly books have an animal of some sort. This thing is some synthetic disaster.

Leo: Gryphon or something, yeah.

Steve: It's something with a forked tail, and the hind end of a tiger, and the front end of, I don't know, kind of a medusa, but with dreadlocks or something. I don't know. Anyway, quite frightening. And of course it's written by, it says down in the lower right, Bob Johnson with His Therapist.

Leo: Drive yourself nuts with Web Assembly.

Steve: That's right. And of course I'm sure I'm not the only person who's ever written extensive web server side technology in assembly.

Leo: You have?

Steve: Oh, yeah.

Leo: Oh, my god.

Steve: All of mine is written in assembler.

Leo: Oh, my god.

Steve: ShieldsUP! and the DNS Spoofability Test. The Certificate Fingerprint stuff. All of the GRC services are in assembler.

Leo: Wow. I didn't know that. I assumed you use some web-facing technologies. Wow.

Steve: No.

Leo: You're brave. Brave man. It's running on IIS, so you know it's an Intel processor. So you know it doesn't need to be portable, in other words.

Steve: Correct. And just like me, it will never port. It will...

Leo: x86 forever.

Steve: That's right. I'll go down with the Intel ship. But it looks like that's not going to happen immediately, so...

Leo: You've got plenty of time. I think you're good.

Steve: We did, toward the end of the AMA with Ant, someone asked something about language, my favorite language or least favorite language. It was something about got me on the topic of language. And so I had a little bit of an opportunity to sort of put RISC and CISC in context. And maybe it was my least favorite processor. I don't remember what it was. But anyway, I was able to - I did spend a little time talking about how, in the case of the DEC minicomputers, like the PDP-11 and the VAX, which was sort of their ultimate instruction set design. It was almost like writing in a high-level language. I mean, unlike a RISC, which is so painful even I won't program that in assembly language, the complex instruction set computers were really pleasant to program in assembly language. And that's what I'm doing with the x86 for as long as it lasts.

Okay. So we have a new section, we'll see how long the section supports itself on the podcast, titled "Windows 11 Watch." The first item got a lot of retweets and hearts or likes or whatever you call them under Twitter, when I tweeted this yesterday. It is the name of a registry entry named, believe it or not, officially supported by Microsoft: "AllowUpgradesWithUnsupportedTPMOrCPU." It's actually the name of a registry whose purpose is made quite clear by its name. And it does, indeed, do what we would think.

So at HKEY_LOCAL_MACHINE\SYSTEM\Setup\MoSetup, believe it or not, M-O-S-E-T-U-P. That's a registry key which already exists. On my Win10 machine I had two subkeys below it. But at that key, MoSetup, you create a "REG_DWORD" value named "AllowUpgradesWithUnsupportedTPMOrCPU." You set it to "1" and reboot so that the change takes effect. With that registry value set, as you would expect, Windows 11 setup will upgrade even over a TPM 1.2, you do still need to have at least 1.2, and without being blocked by what is otherwise a perfectly fine CPU version. It's like, wow. I mean, there are so many hacks emerging, you know, like third-party scripts and weird ways to get around what is clearly an arbitrary restriction because, after all, you set the registry key, and it says, oh, never mind. Fine. We'll run Windows 11 on this.

Our friend Simon Zerafa, after I tweeted that yesterday, he replied: "Yes, but not receiving security updates will be an issue. Probably best to avoid Windows 11 until the dust settles???????" With, let's see, seven question marks. Okay. So first of all, I agree that Windows 11 is best avoided. We'll be looking at a few additional reasons why in a moment. But in reply to Simon's note about security updates, which Microsoft now says such systems may not receive, I tweeted in reply: "Simon. The Win11 requirements were always complete B.S. They tried to make a power play. It failed. And just as that collapsed, there's no reason to think that such systems will not actually receive all security updates. Microsoft just wants to keep pushing back. But they lost this round."

And really, what possible value can it be to Microsoft, after creating an explicit registry entry called "AllowUpgradesWithUnsupportedTPMOrCPU," to then stubbornly refuse to provide those machines with security updates? That would be nuts. I mean, Windows 11 is already broken. It was born broken. So, you know, they're not going to ever fix it for people who install it? That's crazy.

So today is Patch Tuesday. It's probably too soon to know whether those systems which were upgraded over Microsoft's recently and decreasingly strong objections will receive updates. I mean, actually there's one that they were supposed to be receiving today. We'll see. But we'll certainly know next month since by then many unqualifying, that is, like strictly unqualifying Windows 10 machines will have been gently nudged over to

Windows 11 by their users, who are willing to set that key or do any of the other number of things that you can do now to get Windows 11 running on a Win10 machine. We'll find out a month from now if they're going to get what is I'm sure going to be an exciting Patch Tuesday event. So stay tuned.

I also got a kick out of another reply. Bruno Zuber, who, Leo, his Twitter handle is @bzu. So I thought, wow. That's one of...

Leo: That's a good one, yeah.

Steve: That's a good one. He joined in December of '08. So, you know...

Leo: That's why, yeah.

Steve: Yeah. He was on there earlier. Anyway, he tweeted, I love this: "Steve, 2015: Never10. Steve, 2021: How to upgrade to Win 11, even if your hardware is not supported."

Leo: He does have a point there. He has a point.

Steve: Yes, he does. I said: "Touch." So, you know, just for the record, I guess, I mean, this caused me to do some introspection. What is it? Well, it's that I'm annoyed by anyone being, like technology being abused against us. I was annoyed when Microsoft was trying to force Windows 7 users to use 10, even if they were quite happy with Windows 7. And I'm also annoyed if Microsoft makes an entirely arbitrary decision not to allow people who can clearly use Windows 11 with no problem at all, not to. So it's just, you know, that's the thing that bugs me.

So I use Windows 7 through every morning and early afternoon at my primary workspace. I'm sitting in front of it right now. But my microphone and earphones which I'm using right now are connected to a tiny little Win10 machine, which I use for connecting to TWiT. I also sit in front of a Win10 machine on that Intel NUC, the one with the wide curved screen, every evening, I talked about. And furthermore, any new machine I set up would be Windows 10. And now I suppose, yes, it would be Windows 11. You know, Microsoft's going to do what Microsoft's going to do. I harbor no illusions about anyone's ability to affect their actions.

But again, I think it's about giving users choice. And fussing around with an operating system is an entirely reasonable preoccupation. I've spent, and I know you have, Leo, untold hours fussing with various operating system platforms through the decades. It's something that geeks and nerds like to do. But Simon's also right that depending upon how much time one enjoys spending fighting with the system's default settings, and with the possibility of new incompatibilities, it probably would be far saner to wait a bit.

And to that end, speaking of Windows 11 incompatibilities, Joel over at ExtremeTech wrote: "If you're contemplating upgrading to Windows 11 on an AMD system, you may want to hold off just a bit. The semiconductor design firm," meaning AMD, he wrote, "has confirmed that Windows 11 performance is a bit lower on Ryzen CPUs than under Windows 10 right now."

AMD posted a support knowledge base FAQ, it's number PA-400, titled "AMD processors running some apps up to 15% slower." They note that even though 175 different processors which Microsoft says Windows 11 installs on run just great, and the idea that they even have 175 different processors is something of a question mark in my mind, due to an unspecified problem with AMD's L3 caching: "Measured and functional L3 cache latency may increase by on the order of factor of three."

So in their notice under "Impact" they said: "Applications sensitive to memory subsystem access may be impacted. And expected performance impact of 3-5% in affected applications, 10-15% outliers possible in games commonly used for eSports." And in addition to that, AMD said that something known as "UEFI CPPC2," which is the support for their so-called "preferred core," may not preferentially schedule threads on a processor's fastest core. When I encountered this I wasn't aware that not all cores on big AMD desktop machines are created the same. You know, Intel's are symmetrical, but apparently not all of AMD's are. And I hear you guys over on MacBreak Weekly talking about, you know, Apple's performance cores versus their, what's the alternative...

Leo: Efficiency.

Steve: Efficiency, right. Their efficiency cores. And I get it. That absolutely makes sense on a battery-operated platform to be able to run, like keep the system alive using cores that are substantially more efficient and consuming less power, and only fire up the thing that heats the thing up in your hand when you absolutely need that performance. Anyway, to address both these issues, AMD has said that updates to Windows and unspecified software are in development, probably maybe some UEFI updates to fix that, to address this issue with expected availability in October of 2021. And that's one of the changes that I referred to as, like, apparently was going to be part of today's Patch Tuesday. So although maybe that's on all of AMD's processors which did have Microsoft's blessing.

Anyway, at ExtremeTech, Joel concluded his treatment of this by noting, he said: "There's no firm timeline on when fixes will be available for either bug, but AMD is promising they'll be ready this month. This issue is separate from the performance-impacting security features that are baked into Windows 11, which impact AMD Zen performance by 4-5% if left enabled on an OEM system or turned on by an enthusiast." He said: "While these two issues are unrelated, the net effect of them knocks most of a generation's worth of improvement off AMD's CPU cores." That is to say, if you run Windows 11, AMD Zen performance? You fall back a generation.

He said: "Enthusiasts will want to be careful when using OEM PCs, both as far as driver updates and underlying security configurations are concerned. At the same time," he said, "it's not unusual for a brand new OS to have some teething problems. So this isn't likely to represent some kind of long-term referendum on Windows 11's gaming performance. As we've covered," he wrote, "Windows 11 is a bit more 'meh' than some of Microsoft's previous releases. Gamers don't need to be in any hurry to jump for the new OS. And while there's no reason to specifically downgrade to Windows 10, there's no great reason to upgrade to Windows 11, either." And of course that summarizes our position. So thanks, Joel, for the AMD gamer's perspective on Win11.

Also, the Windows 10 taskbar is appearing on Windows 11. New Windows 11 users are reporting that after upgrading over Windows 10, their Windows 11 retained the old Start Menu and that there was no Windows Store. It had disappeared. An active thread over on Reddit suggests rerunning a full Windows 11 reinstall, like on top of what you already have now, or creating a new user and deleting the old user. It appears that something about user profiles are getting messed up in the conversion. So abandoning the old profile that was imported from Windows 10 and restarting with a new profile under Windows 11 appears to resolve the trouble. Unfortunately, all prior user settings, which can be many, I'm often jarred when I'll create a new user. It's like, oh, my goodness, I'm back to new install look, you know, because I've made so many changes over time.

Anyway, all prior user settings and some apps may also need to be restored and reinstalled. The earlier releases of Windows 11 also exhibited this behavior, so it's been seen before by those who enjoy playing with their environment for its own sake. And, again, no idea when this will be resolved.

Microsoft is disagreeing with themselves in some instances. When performing the upgrade, some users are stopped with the message "This PC doesn't currently meet all the system requirements for Windows 11," even when their hardware is compatible, and Windows 10 setup agreed to go along with it. And what makes this more confounding was when those users run the latest PC Health Check app, they are told that their hardware is compatible and will work without trouble with Windows 11. So who knows what's going on?

We have an update on the Windows Explorer RAM leak I mentioned previously. As we know, ever since the release of Windows 11 previews, File Explorer has been experiencing a memory leak causing the application to use too much system memory. For some, the leak has caused File Explorer to use 1GB of memory after opening several folders. And as I previously noted, after File Explorer is closed, that memory is not released back to the system. It remains unavailable until Windows 11 is restarted. The good news is that Microsoft found and fixed the issue in Windows 11 22454 preview build for the Insider dev channel. It's not known when it will be made widely public, but presumably with updates next month.

Also, VirtualBox and Windows Hypervisors do not get along. At the moment, Hyper-V, Windows Hypervisor, and VirtualBox are mutually incompatible. So Windows 11 setup won't agree to set up until they're removed. Oracle and Microsoft are working to get that fixed.

And lastly, there are dropped UDP packets occurring if third-party network optimization is in place. Intel produces something called "Killer," which made me think that naming my Portable Dog Killer that wasn't really such a bad idea.

Leo: No. It's the worst name. It's Killer WiFi. I have it on a couple of my machines. I hate it. It's just a terrible name.

Steve: Yeah, it is a bad name. Dell calls their similar offering "SmartByte," which seems a lot better. Both of those are attempts to optimize network throughput by prioritizing network packet flow. So they're a sort of automatic Quality of Service kind of a shoehorn on the system. The trouble is, for some reason, when they are prioritizing UDP packets under Windows 11, those packets get dropped, lost and forgotten. It's on Microsoft's list of known problems with Windows 11, and it's expected to be fixed, oh, in today's Patch Tuesday. So maybe if we have some listeners who previously broke the rules and installed 11 on a machine where Microsoft said "No no no," and then later said, "Well, maybe," it would be interesting to know whether this is among the things that are known to be fixed, and if those machines get them.

Just, again, I can't imagine that Microsoft is going to produce what is by any definition a half-baked, you know, soggy Windows 11, and then make it possible for, I mean, they've documented this value, this registry value, by the way. I don't know if I made that clear.

It's on Microsoft's official pages that, okay, if you want to do it anyway, fine. And then not fix it? It's just not conceivable. I mean, you know, could happen, but really.

So today's Patch Tuesday. We've entered into an era where we're going to be watching now two versions of Windows being fixed on the fly. Apparently today there will already be patches for the Windows 11 released last week. Since nothing really changes, as we know, under the covers from one Windows release to the next and of course remember that's why most of the troubles historically that once affected Windows 7 and 8.1 and 10, well, okay. The only reason that's true is it was all the same code; right? So now with Windows 7 only being still supported for organizations who pay, we'll have updates which broadly affect 8.1 and 10 and 11, with a bunch of extras for 11 only, due to the fact that Microsoft, as I keep saying, clearly shipped Windows 11 well before it was ready for release. Next week we'll take a look at what happened.

Okay. So rather than the tyranny of the default, we have the joy of the new default: Excel 4.0 macros finally to be disabled. Kinda. For years, ever since Excel 4 introduced macros known as XLMs, Excel macros, they have been one of the most abused features of Microsoft's Office suite. And, you know, that's saying something. Excel macros have been a favorite of malicious campaigns including recently TrickBot, Qbot, Dridex, Zloader, and many more. They were introduced nearly three decades ago, back in 1992, so 29 years ago, with the release of Excel 4.0. These macros provide users with a means of executing commands from within Excel cells. And yeah, you know, what could possibly go wrong? And although these XLM-style macros were superseded long ago with the release of Excel 5 which introduced VBA, right, Visual Basic for Applications, naturally, to be backward compatible, support for XLM macros has remained in place to this day.

Now, talking out of both sides of their mouth, due to the continued known abuse of these XLM macros, while they have deliberately left XLM macros enabled, Microsoft has been recommending that users switch away from and disable this style of macro for years in favor of their newer and more secure VBA macros. And VBA macros actually do have the opportunity, at least, to be more secure since VBA macros have supported for quite a while the so-called AMSI, the Antimalware Scan Interface, which can be used by its formally supported API to allow security software to scan macros for malicious behavior. After many, many years of dragging their feet, Microsoft just added support for AMSI scanning to XLM macros this last March. But this was seen as much too little and much too late.

Microsoft may have finally been spurred to some action because of a huge relatively recent spike in XLM abuse which was first noted in early 2020. A number of security researchers noted the sudden and unexplainable increased attention that XLM macros had been getting from numerous top-tier bad guys, among those pieces of malware that I just referred to. Reports from VMware, ReversingLabs, Lastline, MadLabs, Expel, Deep Instinct, and others observed a sharp spike in malware strains and threat actors abusing XLM macros for any purpose from cyberespionage to banking trojans, ransomware, and cryptocurrency theft.

Finally, this past summer, security researchers began loudly and publicly criticizing Microsoft for leaving users exposed to attacks, asking for more action from the Gods of Redmond, namely that XLM macros, again, long having been only of legacy value, should be disabled by default within Office applications. In this way, the researchers have argued, the companies which actually still rely upon legacy XLM could selectively reenable it for their employees, while everyone else who is being actively abused by having XLM always enabled would then be, and would remain, protected from Excel documents containing malicious XLMs.

The logic, of course, of that is flawless. But believe it or not, Microsoft will still not be disabling this massively abused 30-year-old technology by default for everyone. They will, however, be disabling it now for their paying subscribers as part of the Microsoft 365 service. So any enterprise admins listening to this podcast should know that Windows group policies can be used to disable this unneeded and unused macro capability to protect their users globally.

And non-enterprise users, end users, I use Office stuff offline, can do this for themselves by opening Excel and going to Excel Options, Trust Center, then click on the Trust Center Settings button and select Macro Settings on the left. You'll see then a set of options which are very clear, and absolutely disable XLM macros. Well, in fact, you're able to disable it and request a notification, which is probably what you want if you think there's any reason for macros to be tucked into cells of Excel objects. And remember that it doesn't have to be an Excel spreadsheet.

Thanks to Microsoft's cross-embedding, it's entirely possible, and it often happens, that a Word doc will have an embedded Excel invocation in the form of an ActiveX object, also known or previously known as OLE, which allows the Excel functionality to be imported into a .doc file where an XLM macro will happily run and do whatever the bad guys have set it up to do. So again, Microsoft really doesn't want to break anything by disabling this, worried of the consequences, which I understand. But people are being hurt by it.

Google recently sent warning notices to more than 14,000 users of Gmail, warning that: "Government-backed attackers may be trying to steal your password." In a longer note, I have a screenshot of what was sent to everybody in the show notes. Basically it says, you know, they wrote: "There's a chance that this is a false alarm, but we believe we detected government-backed attackers trying to steal your password. This happens to less than 0.1% of all Gmail users. We can't reveal what tipped us off because the attackers will take note and change their tactics. But if they are successful, at some point they could access your data or take other actions using your account. To further improve your security, based on your current settings, we recommend...." And then they have, I guess this is sort of a depending upon who you are. And in this instance it said: "You're a security pro. Just keep Microsoft Word up to date, or open Microsoft Word documents with Google Docs."

So as I said, these notices were sent to notify Gmail users that they've been the target of a spear-phishing attack orchestrated by a state-sponsored hacking group. Now, I'm familiar with the term "apex predator," and the idea sort of gives me chills. Wikipedia says: "An apex predator, also known as an alpha predator or top predator, is a predator at the top of a food chain, without natural predators." But until now I've never encountered the term in the context of the security industry. But we now have what are being called "Apex Threat Actors," and their tracking and identification is the mission of Google's TAG team. TAG of course is Google's Threat Analysis Group being led by Shane Huntley. We've been mentioning the TAG team often recently because they've been locating and reporting many very valuable vulnerabilities in everyone's software.

Shane told a reporter for The Record that: "In late September, we detected an APT28" - and of course that should be a familiar number to our listeners - "an APT28 phishing campaign targeting a large volume of Gmail users, approximately 14,000 across a wide variety of industries. This particular campaign," he said, "comprised 86% of the batch of warnings we sent this month. These warnings indicate targeting, not compromise. If we are warning you," he said, "there's a very high chance that we blocked the attempted hack." And indeed, in this case all attempts were blocked. Huntley added that: "If you are an activist/journalist/government official or work in national security, this warning shouldn't be a surprise."

Leo: I'm feeling kind of left out, to be honest.

Steve: Awww, yes. Leo, we could send each other...

Leo: I get Apple invitations now, but I still don't hear from Google ever. I don't know.

Steve: Well, hey, that's progress. You're getting Apple invitations?

Leo: Yeah, to their streams. Big deal; right?

Steve: Oh. Not to attend...

Leo: Not to go anywhere.

Steve: Not the Steve Jobs Auditorium. Anyway, he said at some point some government-backed entity will probably try to send you something.

Leo: A little gift horse from all of us here at Kremlin. Enjoy.

Steve: Frankly, Leo, I count myself lucky that I'm not receiving those notices. I'm much happier to be under the radar.

Leo: Well, now we know what APT28 stands for, Apex Predator Team or something like that; right?

Steve: Oh, that's interesting.

Leo: No, no, no, Advanced Persistent Threat.

Steve: It's Advanced Persistent Threat, yes. Anyway, the APT28 group is known by many names, including most popularly, or at least most fun, Fancy Bear, which both the FBI and the NSA directly link to Russian military intelligence apparatus, and in particular to the Russian General Staff Maintenance Intelligence Directorate, also known as the GRU, the 85th Main Special Service Center, GTsSS, Military Unit 26165. Whoa.

So the APT28 / Fancy Bear name comes up often because they've been one of the most active threat actors over the past decade, and the group has often relied on spear-phishing emails in pursuit of targets of interest. Their aim is to breach inboxes, get access to sensitive documents and communications, then pivot to other individuals or internal networks. Anyone receiving one of these email warnings, or anyone who might be a high-value target, a journalist, politician, celebrity, or CEO, is strongly advised to consider enrolling in Google's Advanced Threat Protection for work and personal emails. The Advanced Threat Protection adds and activates additional security protections for high-risk accounts.

And although we're talking about this particular instance, such warnings, for what it's worth, are not a new Gmail feature. Google's been sending alerts of this sort about attacks carried out by state-sponsored entities for about a decade, almost, since 2012. So anyway, just cool that they're being proactive. I think that's good. And clearly if something trips their alarm, they're recognizing behavior. And, wow, you know, 14,000 specific individuals that were being in this instance specifically targeted by a concerted attack by a single threat actor. And I'm being attacked by my dry throat, Leo.

Leo: Thirst. Thirst.

Steve: So I'm going to deal with that.

Leo: I'll have to show you this at some other point, but whitehoodhacker.net is a blog by a high school kid who rickrolled his entire high school district as a senior prank by hacking all the IoT devices, and played Rick Astley's "Never Gonna Give You Up" in talent shows, in Zoom calls, in presentations. He says: "I did it by hijacking every networked display in every school in the Township High School District 214 - that's six different schools, 11,000 students - to broadcast 'Never Gonna Give You Up' in perfect synchronization, whether it was a TV in the hall, a projector in a classroom, a jumbotron displaying the lunch menu. If it was networked, I hacked it." Yeah, I'm sure this is his resume, whitehoodhacker.net. It's hysterical.

Steve: Oh, my goodness. Absolutely. Absolutely.

Leo: I'm sure, I imagine he got in a little bit of trouble.

Steve: Yeah.

Leo: But he has pictures from all over the school of Rick Astley getting...

Steve: Wow. Very cool. So as I mentioned at the top, back on May 6th, Google posted a blog posting titled "A simpler and safer future without passwords," which stated that they were embarking on a campaign to auto-enroll all of their users in two-step verification, which they called 2SV. Note that they call this 2SV as opposed to the industry's 2FA, for two-factor authentication. It always sort of trips me up. I have to like, okay, wait, two-step verification. Doesn't seem natural to me yet. But I guess in time. And at the time...

Leo: Oh, they want to say that because I think it may not exactly be two-factor. You know what I'm saying?

Steve: Right.

Leo: It takes you two steps, but it could be things you know both times. So I think that's why they call it two-step.

Steve: Yeah. Or maybe just a little bit of NIH. I don't know. Anyway, at the time I noted that I wished them luck, since this whole auto-enroll all of their users was sure to be a heavy lift. Well, a week ago, last Tuesday the 5th, they explained how the first step of this campaign was going to happen, and that it was underway, with their posting titled "Making sign-in safer and more convenient." Okay, well, at least safer I would agree with. But there's nothing more convenient about it, if they're only adding steps, as they are, without removing any steps. What makes SQRL, for example, truly convenient by comparison is that it completely replaces both identification and all authentication with its single step. But in any event.

Google has announced their plans to auto-enroll 150 million user accounts by the end of this year into their two-step verification system. It turns out these are accounts where Google's - I just hate that 2SV. I don't know, I just stumble on that every time I hit it - where Google's 2SV login can be enabled, but where users have not done so on their own. Google wrote: "Right now, we are auto-enrolling Google accounts that have the proper backup mechanisms in place to make a seamless transition to 2SV."

Okay. Now, by "proper backup mechanisms in place" I assume Google means that when they break something by doing this unilaterally there will be some reasonable recovery path. And I really do appreciate Google's position on this. What we know is that users won't budge. They just won't. You know, yeah, yeah, yeah, everything's fine. Don't bother me with whatever you're selling. That's users. But they'll surely squawk loudly if someone sneaks into their Google account and starts mucking around with their life. So to make this happen, Google is going to need to be proactive. I get that. And it's for their users' own good, even if Google needs to get all up in their face with this.

So Google's posting explains that this will apply to users with modern smartphones that run recent versions of Android. Once Google proactively and unilaterally enables the 2SV feature, users will be asked to confirm a prompt that appears on their Android smartphone every time they log into their Google account on a new device, app, or browser. And that certainly seems reasonable and not that burdensome. And it would certainly go a long way toward preventing a large class of current remote abuse. And if Google has end-around access to the registered smartphones running their Android OS, their own OS, which enables them to send and receive a real-time push notification, this shouldn't be a big problem. Mostly it's just going to be like a surprise to users, like wait, what? Now I have to do what? On the other hand, only I guess if a user tries to log in on a device where they haven't previously, you know, it hasn't been tagged already by Google as approved.

So today's announcement is the first step in Google's ambitious plan to enable 2SV login support for all of its users by default. This is just the start. As part of Google's long-term goal, more users will have 2SV enabled on their accounts going forward as part of what they say is going to be a carefully executed, staggered rollout plan to avoid large breakage. I still think this is going to be an ambitious and heavy lift. It'll be interesting to see what comes next, since this one, you know, basically using a recent version of their own OS on a device that the user has registered with their account where enough backup information is present in order to keep it from being a problem. And I guess they've identified that as 150 million of their users. That was the easy one to do. What are you going to do for iOS users? Or desktop users? It'll be interesting to see what they have in store for that.

The U.S. Senate has approved some hacking and ransomware legislation, not yet signed into law. But because both houses of Congress appear to be in sync on this, it shouldn't be a big problem. Last Wednesday the U.S. Senate's Homeland Security Committee advanced two bills which are aimed at boosting the U.S. government's insight, meaning their reporting requirements, into cyberattacks on critical infrastructure operators and the private sector, as well as federal agencies. By a voice vote, the Committee approved the Cyber Incident Reporting Act, which would give critical infrastructure owners and operators up to three days, 72 hours, to report hacks, and 24 hours to disclose ransom payments.

The Senate Homeland legislation mirrors a bipartisan measure from the House's Homeland Security Committee that was attached to the House's annual defense policy bill as an amendment. The fact that the bills in each chamber of Congress are aligned suggests that we're going to get that agreed to and signed into law. The Senate bill took on ransomware by requiring organizations, including businesses with more than 50 employees, nonprofits, and state and local governments, to notify the CISA if they make a ransom payment.

Now, that was the original bill. The Committee rejected an amendment that would limit the scope of ransom payment reporting to critical infrastructure operators. So it ends up being broader. Many members voiced concern that the mandate would prove burdensome to small businesses. However, the lawmakers adopted by voice vote an amendment that would, among other things, exempt religious organizations from having to report ransom payments. And the Committee later adopted an amendment which would use the Small Business Act's definition for "small business concerns" to exempt small businesses that meet that definition from having to comply with the ransom payment reporting requirement in the bill. I don't really know why that's burdensome for a small business. But at least, Leo, you and I ever get attacked, I'm sure we qualify as small. You know, it's like 50 members, but it's not strictly 50 employees. It provides a little more leeway for fudging. The definition does not set a fixed threshold, as I mentioned, for the number of employees for the business.

So before long enterprises which do not meet the Small Business Act's definition for small business will be required, by law, to report any ransom payments made to ransomware operators or their affiliates. And all operators of critical infrastructure will also be required to report any and all hacks of their facilities within three days. So I guess it's not surprising that these are not controversial, and as a consequence they're moving through Congress without much problem.

As our listeners may know, since it's certainly been in the news, Amazon's Twitch service was hacked big-time. Last Wednesday we learned that Twitch suffered a major breach, and that they first learned of it when 120GB of their internal proprietary data appeared in a massive online Torrent anonymously released on 4Chan. Twitch said that no user passwords or credit card numbers were exposed. But if that's true, it was about the only thing that wasn't.

They said: "At this time we have no indication that login credentials have been exposed. Additionally, since full credit card numbers are not stored by Twitch, full credit card numbers were not exposed." Now, the wording of that does sort of sound as though perhaps they have been keeping the last four digits, which is a common method of allowing a user to select a blinded card number. In any event, they said that they had reset all stream keys as a result of the incident, so those were lost to this breach. So users who stream to Twitch would need to obtain a new stream key from their Twitch profile backends. And Twitch is owned by Amazon. So they said that while it's still investigating the breach, it believes the breach occurred due to "an error in a Twitch server configuration change" - yeah, when have we heard about a configuration change being a problem?

Leo: It's always, isn't it, an error in the configuration. Seems like that's always the case.

Steve: Yeah, "that was subsequently accessed by a malicious third party." Yeah, no kidding. So the massive data repository which was breached contained the entirety of Twitch.tv, with commit history going back to its early beginnings. Meaning all of its source code. Mobile, desktop, and video game console Twitch client source. Various proprietary SDKs and internal AWS services used by Twitch. Every other property that Twitch owns, including IGDB and CurseForge. An unreleased Steam competitor from Amazon Game Studios. Whoops. Twitch SOC internal red teaming tools. And creator payout reports from 2019 through today.

Leo: That's what got the most attention, of course.

Steve: Yes. And we're talking millions of dollars at the high end.

Leo: There's good money in that, yeah.

Steve: Yeah, of those payouts. So among the treasure trove, the most sensitive folders are the ones containing information about Twitch's user identity and authentication mechanisms, admin management tools, and data from Twitch's internal security team, including white-boarded threat models describing various parts of Twitch's backend infrastructure. There were actually photos that were taken of a whiteboard showing all of the interconnected block diagrams of the threat models, basically the way they see that people could get in. And I saw one that was redacted. It wasn't redacted in the breach.

The unknown leaker promised to release more data, claiming that this was only the first batch, but they didn't provide a timeline, and there isn't any sense for what more is available. It's like, well, wait a minute. There's anything that wasn't released? The threat actor said they leaked the data in response to Twitch's poor handling of "hate raids," which are bot attacks that have flooded the chats of top streamers with abusive content. Although part of what was leaked shows that Twitch was getting ready to deal with that trouble.

The source of the leak appears to be an internal Git server whose domain name is git-aws.internal.justin.tv.

Leo: Hi, Justin.

Steve: Uh-huh.

Leo: Actually, Justin.tv was the original name of Twitch, so it might just be that.

Steve: Exactly. Justin.tv was the name, as you said, of the original company prior to its rebranding as Twitch. Since this occurred 10 years ago, back in 2011, that suggests that that Git server may have been part of some very old infrastructure that hadn't had much attention for the last decade. The leaker labeled this, as I said, "part one," suggesting that more data might be forthcoming in the future. And the biggest question which many security researchers pointed to is why no alarms were triggered, not only as a result of the deep internal compromise this represents, but also during the exfiltration of 125GB of the organization's highly proprietary data. That's a chunk of data, and they had no idea that had been exfiltrated until it was found in a torrent posted online. So, ouch.

Leo: Yeah. I don't know if you met Justin at Gnomedex when we were there. But Justin Kan, who was the founder of Justin.tv, was a friend of the network.

Steve: Oh, cool.

Leo: And iJustine was one of their - he was a life streamer. And then iJustine went on. Then they called it iJustine.tv, I guess, I don't know. But, yeah, he did all right.

Steve: Cool.

Leo: Sold it to Amazon, yeah.

Steve: Good for him. A major Apache web server update introduced a new critical zero-day error. The newly introduced vulnerability was discovered and reported to the Apache team by security researcher Ash Daulton and the cPanel Security Team on Wednesday, September 29th. It was being actively exploited in the wild, so it was a true zero-day. And consequently the fix for it was pushed out very quickly.

It's unclear how long the vulnerability was being exploited. But the Apache group was asked, and they sort of sidestepped the question in a written reply by saying: "As Apache HTTP Server 2.4.49" - that's the bad one - "was only released a few weeks ago, it's likely many users will not have upgraded yet." Okay, well, we have a count. But hold on. They continue: "If and how this issue can be exploited is highly dependent on how users will have configured the server. If you're using 2.4.49, it is recommended that you upgrade to the latest version instead of using access control configuration as a mitigation. On a default installation, an attacker could still use the flaw to obtain the source code of interpreted files like CGI scripts."

Okay, now, what happened was that the release of Apache HTTP Server version 2.4.49 fixed a slew of security flaws including a validation bypass bug, whoops; a null pointer dereference, that's good for a crash; a denial-of-service issue; and a severe server-side request forgery vulnerability. But the major update also inadvertently introduced a separate, new, critical issue: a path traversal vulnerability that can be exploited to map and leak files. The developers wrote: "An attacker could use a path traversal attack to map URLs to files outside the expected document root. If files outside of a document root are not protected by 'require all denied' access control, these requests can succeed. Additionally, this flaw could leak the source of interpreted files like CGI scripts."

So Positive Technologies has reproduced the bug. And Will Dormann, the vulnerability analyst at CERT Coordination Center, says that: "If the mod-cgi function is enabled" - and we'll note it typically is - "and the default 'require all denied' function is missing, then the vulnerability is as RCE" - meaning remote code execution - "as it gets."

So the new trouble only impacts this Apache 2.4.49, which was, as I noted, only a few weeks old. Even so, as of last Wednesday, approximately 112,755 Apache servers were running the vulnerable version, with roughly 40% of those residing in the United States. So first off, props for those running Apache for, like, jumping on a new version quickly. Unfortunately in this case it kind of may have bitten some people because this thing was being exploited, apparently immediately, in the wild. On the other hand, that also suggests that now that 2.4.50, the fix, which only took five days to be released, it came out on October 4th, that suggests that they will all as quickly be moving themselves up to 2.4.50 and then be safe. So, you know, this is the problem with our software, right, is that sometimes regression errors sneak in. We fix a bunch of things, but break some other things. And if they're found, that can create a problem.

Okay. During last week's six-hour Facebook services outage, the alternative Signal and Telegram secure messaging platforms struggled to keep pace with the deluge of new users jumping ship from WhatsApp as they looked for an alternative. Unfortunately, some of those services' new users experienced some lagging service and trouble, since as we've seen when this happened before, both the Signal and Telegram services struggled to keep their own heads above water amid the roaring new demand. This isn't the first time we've noted that new-user sign-up processes might not be scaling as well as they should.

Signal tweeted: "Signups are way up on Signal. Welcome, everyone. Millions of new people have joined Signal today, and our messaging and calling have been up and running, but some people aren't seeing all of their contacts appear on Signal. We're working hard to fix this up." Since the WhatsApp outage was providing a hard "no" for its use, as opposed to being a little flaky, right, I mean, it was gone, even services that may have been limping along at times were better than nothing.

Pavel Durov, who's Telegram's CEO and founder, noted that more than 70 - seven zero - million new users joined Telegram in a single day following Facebook's outage. He added that this massive deluge of millions of new users led to performance issues as they were all trying to sign up on the messaging platform at the same time. Pavel said: "The daily growth rate of Telegram exceeded the norm by an order of magnitude, as we welcomed over 70 million refugees from other platforms in one day. I'm proud of how our team handled the unprecedented growth because Telegram continued to work flawlessly for the vast majority of users."

So, yup, not much loyalty there. I suppose, I mean, if you live in WhatsApp, if you depend upon it, and an hour goes by, ouch. And then two, ouch. And then three. At some point you're just going to say, okay, screw this. Let's all go somewhere else. A bunch of people chose Signal. A bunch of people chose Telegram. Well, you know, 70 million plus people chose Telegram. And you've got to wonder, I mean, I'm sure there are people who are still over on WhatsApp, so it probably created a fragmentation of messaging once WhatsApp was back up and on the air after a total of six hours. Still, there were probably some losses that won't be wandering back anytime soon.

I wanted to close the loop, sharing some of what our listeners have sent to me. I received a Twitter DM from a listener who asked, he said: "Steve, I listen every week, but a lot is over my head. You would help people like me if you did a short segment on Win versus Mac. That is to say, since my Windows computer is old and cannot get Win11, instead of buying a new Win computer, what about a Mac? Trade-offs? Your thoughts. Thanks."

Okay. So first of all, as we know, all past evidence suggests and all new evidence confirms that Microsoft appears to have set the Windows 11 CPU requirements bar quite high. I've heard from many people with recently purchased machines, I mean, like heavy-duty gaming machines, they've got late-model chips, and Win11 says, uh, no. And as I've mentioned, I have a lovely, recently purchased Intel NUC which is by no means old. It runs Windows 10 like greased lightning with a fast 2.6 GHz quad core i7-6670HQ processor with 32GB of RAM and TPM 2.0. There's absolutely no reason for that machine not to gleefully run Windows 11. But so far Microsoft has said no, which is ridiculous. I expect that this might change once they have pushed as many people as they can up to newer hardware. Once that's done, maybe they'll relax the requirements in the interest of resynchronizing everyone under Windows 11.

They'll say something like: "We've finally finished performing further testing on older hardware, and we've confirmed additional compatibility." Ah. What do you know? "So we're further relaxing the requirements. We can now state confidently that, if a machine runs Windows 10, it'll be able to run Windows 11 without problems." So, you know, that appears to be true today. But Microsoft wants us to play along for now. We still have four years of Windows 10 support, and four years feels like longer than maybe they'll be willing to wait before they move to reunite everyone under Windows 11, especially in light of things like the presence of the "AllowUpgradesWithUnsupportedTPMOrCPU" registry key.

In my opinion, under no circumstances should you, the person who tweeted me, purchase a new computer for the sake of running Windows 11. That lovely Intel NUC I mentioned has a wide screen where it's much more fitting to place the taskbar against the screen's left-hand edge. But at the moment, Windows 11 says no to that.

If a lot of this podcast's content feels like it's a bit over your head, I suggest that you might not be ready for a move to one of the Linux desktop environments which, while certainly discoverable, are still a bit less handholding than either Windows or macOS. So, I would say that remaining right where you are, presumably with Windows 10, for the next four years of Windows 10 remaining service life, to see whether Microsoft discovers that, what do you know, Windows 11 is so good that it works everywhere after all. I wouldn't be a bit surprised.

Mark James Wilcox said: "Just remember the timeline of Windows." I got a kick out of this because of course all long-term Windows followers are aware of this. "3.1 good, 95 bad. 98 good, Millennium bad. XP good, Vista bad. 7 good, 8 and 8.1 bad. 10 good, 11... Who knows if this is going to follow a pattern?"

Oh, and I got a big kick out of several people who tweeted. I only grabbed one of them, Philip Le Riche. He said: "@SGgrc (SN-839)." He tweeted: "I don't believe it! I thought I was the only person on the planet still using PSP6." Of course he's referring to my reference last week to Paint Shop Pro, which remains my go-to bitmap editing software. He said: "Simple, does 95% of common tasks." And he says: "I'd use the Gimp, but learning curve too steep for occasional use."

And so I replied to Philip with a shorter version of "Yep, Paint Shop Pro v6, the best, cleanest, and most straightforward bitmapped graphics editor for Windows ever. I also own PSP7, which was the last one before Corel bought it and ruined it. But I didn't like what JASC" - JASC was the original publisher of Paint Shop Pro - "did to PSP7, so I returned to v6. I use a couple of plugins, Eye Candy, with another which does more highly optimized image saving. Any images that appear anywhere on GRC, any that you ever see me post, were tailored, trimmed, and produced by Paint Shop Pro v6. Nothing beats it. And I'm glad to know that I'm in good company."

A bit of errata from last week's statement about Apple's new "Invasion" series, which I said would be starting last Friday. Whoops. Make that Friday after next, the 22nd, October 22nd, as I originally said the first time I mentioned it. I just made a mistake last week. And I haven't yet...

Leo: You got my hopes up, too, by the way. I thought, oh, I can't wait to watch it tonight. And no.

Steve: Yup. Yup. And I've not watched the fourth installment of "Foundation." I assume you have, Leo?

Leo: Not yet. But I like it. I'm liking it.

Steve: Well, there we go.

Leo: I think I started Episode 4, but I haven't finished it yet.

Steve: I rest my case.

Leo: They never did explain the weird ending on Episode 2.

Steve: I know.

Leo: Ever. Right? I don't think they ever did.

Steve: No, they didn't.

Leo: Apparently the book doesn't either. So we just have to figure that one out.

Steve: No, the book doesn't have that happening.

Leo: It doesn't happen in the book?

Steve: It has like a whole different storyline.

Leo: Okay, all right.

Steve: Okay. So last week I shared my decision to rework and rewrite SpinRite's benchmarking technology. Today I very nearly have all that work done. Actually, I was pushing over the weekend, hoping to get it done before the podcast. I got close, but not quite there. Anyway, I am so glad that I decided to go this route. The new UI display for the new benchmark is designed, and I'm rolling along very nicely with the implementation of the new system. Rooting out all of the old code was very gratifying, since it embodied a number of kludges which had long since worn out their welcome. But they were required, as I explained last week, until the use of that flaky old counter/timer could be reliably retired and replaced by using the clean and solid up-counter clock which we've now had for some time.

I'm working on the code to dynamically display the ratio of the accumulated bytes transferred to the accumulated total benchmark time. Thus, as the benchmark runs, both of those baselines extend, and their ratio will settle down into the result. Since the drive speeds it will be encountering will run from the very old to the solid-state, that code needs to dynamically scale to handle bytes, kilobytes, megabytes, gigabytes, and terabytes per second, while being careful to do the math in such a sequence that preserves the greatest number of significant digits. So I expect to have it all nailed down and tested in another day or two, after which I'll release it to the gang in the GRC newsgroup to pound on.

And I'm excited about that since this will be the first widespread testing of SpinRite's new IO abstraction system, which is now fully implemented. And, you know, I've had it running here for quite a while, but it hasn't had wider testing. And at that point every single piece of code that I've written so far, although I actually had written more of the data recovery portion before, remember, I sort of got to feel like, okay, it's been too long since this thing's had any testing. So I stopped, went back, and I brought that all current. So anyway, that's where we are, making great progress. And Leo.

Leo: Nice.

Steve: Once again, progress on rehydration, and then we're going to talk about zero-day angst.

Leo: Zero-days, yeah. All right. Zero-days and Apple.

Steve: So as I said at the top, I was originally planning to lead with this topic, under the podcast's "Zero-Day Watch" heading. But it sort of grew into more than that. Okay. So first of all, it's Apple's turn. Apple has just patched iPhone zero-day in iOS 15, and it's an authentic zero-day because it is being exploited in the wild. It's tracked as CVE-2021-30883. The zero-day resides in the IOMobileFramebuffer, which is a kernel extension that allows app developers to manage a device's screen framebuffer, as it's called, which is memory. And as a consequence of this flaw, malicious applications were able to execute arbitrary code with kernel privileges thanks to using this vulnerability. And as we know, running one's own malicious code with kernel privileges gives the attacker full control over the device.

As always, Apple is mum about the technical details of the vulnerability, or about how the vulnerability was being leveraged. However, an unrelated security researcher immediately posted a detailed technical teardown on GitHub based upon his discovery from comparing the pre- and post-patched code. As we've noted before, "bindiffing," as it's called, is the practice of comparing a compiled binary file containing a now-known vulnerability which has been fixed to the same compiled binary after it has been patched and repaired. In other words, the repair changes the file. So BinDiff stands for binary difference. So by comparing, by like scrutinizing where the file is different, and then reverse-engineering that region of the code, it's often the case that the problem that was fixed is revealed.

In this case the researcher wrote: "In the last iOS security update 15.0.2, Apple fixed a vulnerability in" - and by the way, this was just yesterday - "a vulnerability in IOMobileFramebuffer/AppleCLCD, which they specified was exploited in the wild. This attack surface is highly interesting because it's accessible from the app sandbox, so it's great for jailbreaks," he wrote, "and many other processes, making it a good candidate for LPE exploits in chains," he says,

"WebContent, et cetera." And of course LPEs are Local Privilege Elevations.

He says: "Therefore, I decided to take a quick look, bindiff the patch, and identify the root cause of the bug. After bindiffing and reversing, I saw that the bug is great, and I decided to write this short blog post, which I hope you'll find helpful. I really want to publish my bindiff findings as close to the patch release as possible, so there will be no full exploit here. However, I did manage to build a really nice and stable proof of concept that results in a great panic at the end." He said: "Sorry in advance for any English mistakes. I prioritized time over grammar." And he says: "Good thing we have automatic spell checkers."

So anyway, he then proceeds with a very long - and I forgot to mention I have the link in the show notes for anyone who wants to see the whole thing - a very long and satisfyingly detailed breakdown of the now fixed, or for those not yet patched, soon to be fixed in your devices update. When I checked my iPhone, it was still back - this is an iPhone, what do I have, an 11. It was still back on the last - wait. I have an iPhone. I forgot, Leo. Where are we with iPhones? I've got, like, one back from what they most recently did. I can't remember. I don't even know what iPhone number I have. Anyway, when I checked my iPhone, it was still back on the last version 14 release, that is, of iOS. So I updated it to v15.0.2, which has this problem solved.

Okay. Since we now have a "Zero-Day Watch" section of this podcast, and since we've been counting the Chrome and Chromium zero-days year-to-date, we just talked about one last week, I think it's only fair to note that this latest zero-day in iOS brings Apple's year-to-date count to 17. So more than Chrome/Chromium. I've got a list of all of them in the show notes. Given the big target that every web browser presents, and the fact that the other zero-day march we've been following is the world's leading web browser, Chrome, it should not come as any surprise that 10 of those 17 zero-days in iOS were discovered and fixed in Apple's WebKit browser component. Again, this is the component which is out on the front lines, is receiving script from websites you visit, which are not necessarily trustworthy, and also receiving script from things those websites import, ads and other third-party content.

So one point of interest was brought out by this researcher, who noted that the July 26th zero-day, which I've got listed above, CVE-2021-30807, was also an IOMobileFramebuffer zero-day impacting iOS, iPadOS, and macOS. So one wonders whether, once Apple removed that earlier zero-day in the framebuffer module, those attackers may have simply switched to using another zero-day they already knew of and had at the ready in the same module. In any event, stepping back from this a bit, I think it's clear that the engineering practices surrounding the creation and maintenance of the best-designed software, which is what I would argue Apple and Google are both capable of producing and do produce where it matters most, has become so secure that the world's users should feel completely safe using it. But at the same time, that software has become so complex that our current development methodologies, languages, and toolsets clearly fall short of creating perfect software.

So does software have to be perfect? Perfect software doesn't support a running total of zero-days so far seen this year. And those tiny imperfections give the likes of Israel's NSO Group, with their Pegasus smartphone spyware, just enough of a toehold to enable highly targeted attacks against the world's highest value targets.

At this stage in the development of the Internet and the use of personal connected devices, predominantly smartphones, what Google has created and accomplished with Android has been phenomenally valuable to all mankind. And placing sharing first, doing it all in plain sight for the world to see and benefit from, to examine and unfortunately to also attack, whether it's Android or Chromium, is significantly more difficult than working to keep everything a tightly locked-down secret. This sort of development in plain sight, you know, I mean, it's tremendously beneficial. But it allows the bad guys to see what's going on, too. At the same time, if one's goal, rather than altruism, is profit, secrecy has its place.

Apple, whose products are as proprietary as they're capable of being, made a serious strategic security blunder when they chose to share code between iOS and macOS. Sure, it made huge economies of development. Why continually reinvent the wheel on separate platforms? But the security of iOS is arguably far more critical than that of macOS. Yet by merging and sharing their codebases, iOS's security has been reduced to the lowest common denominator.

In that list of Apple zero-days so far this year, I highlighted in red those shared by macOS and iOS. Of the total 17, 10 again were zero-days present in both OS platforms. Want to place a bet where those zero-days were first discovered? After the merging of Apple's codebases, we've encountered many examples, we've talked about them on the podcast, where researchers and attackers were able to reverse engineer the code first on the unprotectable and far more accessible macOS platform - and, by the way, where they really couldn't care less about a vulnerability on macOS - then pivot with what they learned there, to the far more valuable iOS platform, knowing that the same vulnerabilities, though unseen so far on iOS, were likely to exist there, too.

The advantage Apple had, and unfortunately squandered, was that any code running on an iDevice is inherently far more easily protected, hidden, and kept secret. The favorite slogan "Security through obscurity is not security" makes for a catchy phrase, but it's not quite true. Some obscurity, any obscurity, is better than none. And at some point, sufficient obscurity becomes secrecy. "Security through secrecy" does provide true security. After all, we all keep passwords and private keys secret which, notwithstanding other errors in their management, keeps them secure. Apple's iDevices today are measurably less secure than those of the past because keeping their code secret had true security value. And that's been lost.

Although Google's device security does suffer from its total openness, the openness of its Google Play Store and its support for sideloading completely uncurated applications, you know, that's tough. Google's efforts are providing far more benefit to the world at large than Apple's.

I titled this podcast "0-day Angst" because I wanted to place the issue of zero-day vulnerabilities front and center and in a sober context. Yes, it's true that despite everyone's best efforts, zero-days occur, and there's no reason to believe that the near-term future will see any change in that. We're not going to reduce the features nor the complexity of our software. No new bulletproof languages, development, or environment solutions are apparent. And the truth is zero-days are more of an embarrassment to their publishers than a true global security threat. And they're also inherently self-limiting. The more they're used, the more likely they are to be discovered and eliminated.

Thanks to a great deal of effort being made by mainstream software publishers, aided by a massive, distributed, and growing community of security researchers, and even with some inadvertent help from the bad guys, today's devices, even though we absolutely positively know they still contain known and unknown vulnerabilities, are more than secure enough for nearly everyone to use without worry. Only those very few who are likely to be targeted by nation-state actors have any real reason for concern.

So, yeah. We'll talk about zero-day vulnerabilities. We'll point them out. We do need to keep them under control. But by no means should anybody inconvenience themselves, for example, updating their iPhone from 14 point whatever the last one was, or the very first 15 to 15.0.2 today, because there was a problem that was known being found exploited in the wild. It has to have been the case that that was being used by somebody in very tightly targeted attacks. And that's just not going to affect most of us. And Leo and I, you and I don't get email from Google telling us to duck.

Leo: I think "high-value target" is the right word because these exploits I suspect are difficult to both discover and achieve, especially in iOS. And as a result, they are pretty much reserved for the highest bidder. I mean, it's hard, you know, they're hard to find. If you find it, you sell it to NSO Group, and they're going to pay you millions because they're going to go on and sell it only to nation-states who will also pay millions to target a handful of individuals with zero-click exploits. So I think that whole supply chain means it's really mostly likely, it's not ransomware gangs that are buying these or finding them.

Steve: No.

Leo: It's researchers who are very good, very sophisticated who are finding them. And instead of unfortunately selling them to Zerodium or even directly to Apple...

Steve: And they're quietly making themselves a good living, yeah.

Leo: They're getting millions of dollars from people, unfortunately, like NSO Group, who probably outbid Zerodium. And the reason they do is because they have high-price customers who will pay a lot of money.

Steve: Yup.

Leo: But they use them in such narrow ways that, you're right, I doubt it's a threat. I actually - we talked about this on the radio show. Somebody was worried. And I said, unless you're working for a three-letter agency in McLean, Virginia, or you're a dissident in Turkey or Bahrain, you're probably okay. Still, I'd get 15.0.2 as soon as I could.

Steve: Yeah.

Leo: And I always tell Lisa, I said do 15.0.2. And Apple, you know, they don't say, oh, you've got to do it. They pop up things and stuff. They just slowly eventually get everybody updated to the release.

Steve: Yeah, I was still back on 14.

Leo: That surprises me.

Steve: Yeah.

Leo: They pushed 15 pretty hard. But you must never check your update section. So that's fine.

Steve: Yeah, I don't.

Leo: You're happy. You don't need it.

Steve: I am.

Leo: He is the happy-go-lucky, which is amazing given what he knows, Steve Gibson. He is our security guru here at Security Now!. You can get copies of this. I'm going to tell you when we do it so you can watch live if you want. It's fun to watch live because you can chat along with other people watching live. We stream it every - right after MacBreak Weekly, which is every Tuesday, usually between 1:30 and 2:00 p.m. Pacific. That's about 5:00 p.m. Eastern. That would be 21:00 UTC.

The live streams go on all day and all night at TWiT.tv/live. If there's not a live show in production, you'll see reruns of live shows and productions so you can, you know, like this whole thing will be repeated later today. So even if you're not around at that particular time, you can watch. If you're watching live, chat live. We have a great IRC chat room, irc.twit.tv, where you can converse with others watching at the same time.

If you want to get a copy of the show, there are a couple of unique versions on Steve's site. 16Kb audio, he's the only one that does that. And that's for the bandwidth-impaired. And there's beautiful transcriptions written by Elaine Farris. He hosts those, as well, at GRC.com. While you're there, pick up the world's best mass storage maintenance and recovery utility. That would be SpinRite v6. Currently 6.1 is on its way, as you heard. And you will get it automatically if you buy 6.0 now. You can also participate on the forums in the development of 6.1. Steve's got some great forums at GRC.com. Lots of other stuff, too.

We have copies of the show on our website at TWiT.tv/sn for Security Now!. There's a YouTube channel devoted to Security Now!. All the videos are up there, all 800 and some. Actually, we didn't start doing video right away. So I don't know how many videos.

Steve: No.

Leo: 600, something like that. And you can also subscribe. Probably that's the thing to do. If you're really a fan and you know you want every Security Now!, collect all 840. If you subscribe to the podcast, we'll keep you up to date automatically. They'll download the minute it's available.

There are lots of different podcast players. Certainly Apple and Google have theirs. Pocket Casts is very popular. If your podcast player supports reviews, do leave us a five-star review. Let the world know Security Now! exists. We need to get the word out. More people ought to listen to the show, obviously. Steve, we'll see you back here next week. Good job.

Steve: Yay. Thank you, buddy.