The New EU Surveillance State

Description: This week we look back at what no one wanted, an eventful Patch Tuesday. Apple has pushed a set of updates to close an actively exploited zero-day. Google announced the creation of their Open Source Maintenance Crew. A ransomware gang wants to overthrow a government. Google's Play Store faces an endlessly daunting task. The predicted disaster for F5's BIG-IP systems arrived. A piece of errata and some closing-the-loop feedback from our terrific listeners. Then we're going to look at just how far afield the European Union has wandered with their forthcoming breathtaking surveillance legislation.

High quality (64 kbps) mp3 audio file URL: http://media.GRC.com/sn/SN-871.mp3
Quarter size (16 kbps) mp3 audio file URL: http://media.GRC.com/sn/sn-871-lq.mp3

SHOW TEASE: It's time for Security Now!. Steve Gibson is here. Big show planned for you. Why the government is saying don't install the Microsoft patches from last Tuesday, a Patch Tuesday that didn't go as well as planned. We'll talk about the ransomware gang that wants the citizens of Costa Rica to rebel. And a new proposed regulation in the EU that could mean disaster for privacy. It's all coming up next on Security Now!.

Leo Laporte: This is Security Now! with Steve Gibson, Episode 871, recorded May 17th, 2022: The New EU Surveillance State.

It's time for Security Now!, the show where we cover your security, privacy, safety online with this guy right here, Steve Gibson of GRC.com. Hi, Steve.

Steve Gibson: Yo, Leo. Great to be with you.

Leo: Careful. The drone is right in front of you. It could wake up at any time.

Steve: Well, it's tethered right now, so that's good. It can't escape because it's got a tail on it. For those who don't know, Leo's been playing with a little inexpensive sort of a selfie cam drone that doesn't really cut the mustard.

Leo: But as you point out, I mean, given this thing is about four square inches, it's pretty impressive what it can do.

Steve: Yeah. It's basically flying propellers that suspend a camera.

Leo: Yeah, and it has sensors, not only a camera but has sensors and things. It doesn't run into things.

Steve: It's got to have position sensors, inertial sensors in order to be stabilized and all that. Yeah.

Leo: Pretty impressive.

Steve: But anyway, so we're at Episode 871 for mid-May. And I titled this one "The New EU Surveillance State." That was about the fourth title the podcast got. And I kept changing the title as I read more deeply into the details of some proposed legislation in the EU which first leaked last Tuesday and then was - it was funny, too, because the leaked copy actually had the word "sensitive" on the front page. Yes.

Leo: Yeah. Yes, it is.

Steve: And that was removed from the actual formal official legislation that came out the next day on the 11th. Anyway, we've got to talk about that because it's been compared to the CSAM Apple stuff. No. This is way beyond what Apple was proposing. It's, well, it's breathtaking. Anyway, we'll get there. First we're going to take a look back at what no one wanted, which was an eventful Patch Tuesday. You don't want your Patch Tuesdays to be eventful.

Leo: No.

Steve: You want them to be uneventful.

Leo: Quiet, yes.

Steve: And we didn't get that. Apple has pushed a set of updates to close an actively exploited zero-day across a bunch of their products. We'll touch on that. Google has announced the creation of their Open Source Maintenance Crew. That's the formal name of it, the OSMC, the Open Source Maintenance Crew. A ransomware gang has the temerity to call for the overthrow of a government. Google's Play Store is facing an endlessly daunting task which we'll mention and talk about and look into. The predicted disaster for F5's BIG-IP systems, which we expected last week, that arrived, right on schedule.

We've got a piece of errata. I've got a bunch of closing-the-loop feedback from our terrific listeners. Then we're going to look at just how far afield the European Union has now wandered with their forthcoming breathtaking surveillance legislation. And last week I mentioned, because it was in the context that we were talking about, that I had already cued up this week's Picture of the Week, which talks about supply chain security, which actually we'll be talking about also in the middle of the podcast. So I think another great podcast for our listeners.

Leo: Yeah, boy, this Picture of the Week is not funny at all. Not a comic this time. Picture of the Week time.

Steve: So, yeah, this is a perfect snapshot to characterize the current state of the security of the open source software supply chain. This was a legitimate - I actually dug into it and went to the NPM listing for this. Someone named Lance R. Vick, he put up a note on mastodon.social. He wrote: "I just noticed 'foreach' on NPM is controlled by a single maintainer. I also noticed they let their personal email domain expire, so I bought it before someone else did."

Leo: Oh, my god.

Steve: He said: "I now control 'foreach' on NPM."

Leo: Oh, my god.

Steve: "And the 36,826 projects that depend on it." And when I went over there, sure enough, more than six million downloads of this little "foreach" module per week. So this thing is like deeply dependent. All it does is iterate across...

Leo: It's dopey, yeah.

Steve: ...an array like, I mean, it's like the dumbest thing. It's like, why would you just not - why would you go get that? But lots of people do.

Leo: Well, Python has it. A lot of languages have it. I guess for some reason JavaScript doesn't. So now you can.

Steve: Yeah, so it adds that to it because, you know, you wouldn't have to want to write an iterator on your own. Oh, goodness, no.

Leo: Heaven forfend.

Steve: Let's instead depend on somebody else's who apparently has wandered off and allowed his personal domain to expire. But here's the point. So this is the system that has evolved. And we're going to be talking about this today, talking about the Open Source Software Security Foundation because there's news there. And I'm really glad there's news there because open source software just sort of started off as being kind of a curio, and everybody would agree now it's become a real force. The problem is it's much less fun to maintain and secure things than it is to just write them. And so a lot of stuff is written and then, like, here you go, I'm going to go back to my regularly scheduled job. And things kind of just get left. So we have to fix this over time. We'll get there in a minute.

First of all, I've observed in the past that what one looks for in a Patch Tuesday is a seamless and uneventful experience. Either you check for updates and then decide to install them, if they're ready for you; or you receive a notice that updates have arrived and have already been installed and are just waiting for you to step away from your computer. Again, uneventful. What you don't want is to see a headline such as BleepingComputer ran yesterday, which reads: "CISA warns not to install May Windows updates on domain controllers."

Leo: Oh, boy.

Steve: In fact, CISA so much doesn't want May's updates installed that they went so far as to temporarily remove the listing of one of their "must patch" mandate's security flaws from their own catalog of known exploited vulnerabilities because they really can't have it listed there while they're also warning all users of Active Directory not to install those updates to fix the problem which is in their catalog because it's known to be exploited. It's a mess.

The headline on CISA's published notice reads: "CISA Temporarily Removes CVE-2022-26925 from Known Exploited Vulnerability Catalog." And they wrote: "CISA is temporarily removing CVE-2022-26925 from its Known Exploited Vulnerability Catalog due to a risk of authentication failures when the May 10th, 2022 Microsoft rollup update is applied to domain controllers. After installing May 10, 2022 rollup update on domain controllers, organizations might experience authentication failures on the server or client for services, such as Network Policy Server, Routing and Remote Access Service, Radius, Extensible Authentication Protocol, and Protected Extensible Authentication Protocol. Microsoft notified CISA of this issue, which is related to how the mapping of credentials to machine accounts is being handled by the domain controller."

So as we know, once upon a time it was possible to individually install security patches so that a troublesome patch could be manually avoided. No longer. Now everything is rolled up into a take-them-all-or-none solution. And I don't blame Microsoft for that, frankly. My mind was always boggled that Microsoft was even able to somehow consider offering a la carte patching of such an incredibly complicated codebase as Windows has become. The complexity of offering that option to me was just astonishing. And I don't know how they even did it. And, as we know, it often didn't quite work as hoped. So now it's all or nothing. And in this case of Windows domain controllers, for the time being, "nothing" is what you want.

As enterprise admins last week began installing the May updates, problems quickly started surfacing, with admins sharing reports online of some Active Directory policies failing with the error message: "Authentication failed due to a user credentials mismatch." And this continues: "Either the user name provided does not map to an existing account, or the password was incorrect."

So Microsoft explained that the issue is only triggered after installing the updates on servers used as domain controllers. Okay, well, that doesn't help those domain controllers. But the updates, they made a point of saying, will not negatively impact when deployed on client Windows devices and non-domain controller Windows Servers. And this is an example of an instance where it's going to be really interesting to eventually watch and learn whether Microsoft's announced and forthcoming Autopatch system turns out to be a good thing or more trouble than it's worth. Presumably, Autopatch is somehow going to handle unforeseen problems like this. Right? I mean, that's the whole point. And at the moment it's unclear where this omniscience is going to come from, since it's apparently not coming from Microsoft. So we're going to be writing that eventually.

The problem all of this is trying to fix, and in this case of this domain controller patch, it's a flaw that's being actively exploited, as I mentioned before, thus the reason it was on CISA's thou-shalt-patch-now list, actively exploited in the wild. It's a Windows LSA, that's the Local Security Authority, spoofing zero-day which has been confirmed as a new PetitPotam Windows NT LAN Manager Relay attack vector. The PetitPotam problem was discovered and named last July by the French security researcher Gilles Lionel. We talked about it at the time, last July, which is probably why PetitPotam sounds familiar. It's just fun to say.

And an NTLM relay attack, which is what this specific PetitPotam exploit allows, it allows bad guys to force devices, in this case domain controllers, to authenticate against malicious servers under their control, essentially joining the malicious server to the domain. Once a device authenticates, the malicious server is able to impersonate the device and gain all of its privileges. This in turn gives attackers complete control over the domain. In other words, not good. Not what you want.

Okay. Now, as for the actual patch itself, we're back to that disheartening story where Microsoft patches to stop the proof of concept from functioning, while leaving the underlying problem unresolved. Gilles has confirmed to the tech press that May's security update, the one that you're not supposed to install even, leaves the underlying problem unresolved. It did finally fix that particular specific problem, and even that was Microsoft's second attempt. They first tried to fix it last August. He worked around that because they didn't really fix it. And now he's saying it still isn't fixed. They closed one specific attack vector in the encrypted file system, which is where this whole thing surrounds this. But he said: "Attack vectors still exist which will allow a slightly modified attack to continue to work." He said: "All functions of PetitPotam, as other vectors, still work except the EfsOpenFileRaw," which was the one thing they fixed.

So to me this feels like there's a bigger problem, that is, they posted workarounds for this and ways of locking things down so that this isn't a problem. I mean, so it feels like it's kind of big, the way all of these printer server problems were that we faced at the beginning of last year, where it just took them basically all year to fix this problem because it was fundamental to some early assumptions that Windows had made back when security wasn't such a focus, and they're kind of stuck with the protocol. Feels to me like they're kind of stuck with not breaking things, which is what would happen if they really locked this down as it needs to be, so they're just kind of trying not to.

So all of this is current as of yesterday. Assuming that Microsoft is eventually able to fix and reissue May's security bundle in a way that doesn't break Active Directory servers, and hopefully they'll test it first this time to be sure, then they will either announce an out-of-band update, or maybe they'll just wait till next month, till June.

Okay, but this all happened last Tuesday, right, on Patch Tuesday. What else happened? We received fixes for three new zero-day vulnerabilities, one of which was that one, and patches for a total of 75 flaws across Microsoft's entire software suite. Eight of those 75 flaws were rated "critical." Twenty-six, which would have been more than one third of the 75, were remote code execution vulnerabilities, now closed. Twenty-one were elevation of privilege vulnerabilities. Seventeen were information disclosure. Six were denial of service. Four were security feature bypass, and I still love that, it's so generic. We have one spoofing vulnerability. Nothing was fixed in Chrome this month by Microsoft.

And remember that Microsoft classifies a flaw as a zero-day different than the rest of the world, if it's been disclosed publicly, even if it's not known to be actively exploited. So somebody talked about two other problems, aside from this PetitPotam problem. But as far as they know, it hasn't been exploited at the time that they closed it. So, okay, fine.

So as we know, given that Exploit Wednesday now follows Patch Tuesday, the urgency to install updates in a timely manner has increased because we just see one example after the other, month after month, that the bad guys are jumping on patches, reverse engineering them, and working to exploit them as quickly as they can.

And speaking of zero-days, yesterday Apple updated watchOS to v8.6, tvOS to 15.5, and macOS Big Sur to 11.6. So that would cover the Apple Watch Series 3 or later, Apple TV 4K 1st and 2nd gen, Apple TV HD, and Macs, which are running Big Sur. In each of those cases the updates fixed an out-of-bands write which could be made to occur in the AppleAVD module. That's a kernel extension for handling, as AVD sounds, audio and video decoding.

It will come as no surprise to our longtime listeners who have all learned to expect trouble to arise in complex media decoders which are inherently complex interpreters of encoded bitstreams. In this case, remote attackers could have, and were known to be, executing their arbitrary code with kernel privileges, which is never good. Apple was as closed-mouthed as usual about this, only saying that they had added improved bounds checking. Which, yes, you would want to add to an unbounded write in one of these modules. So they never share very much. They just update and say this is important, everybody please fix this.

Okay. Google's Open Source Maintenance Crew. Recall that two weeks ago we first talked about the OpenSSF, the tongue-twister, Open Source Security Foundation. At that time I enumerated the gratifyingly large number of participating and supporting companies, pretty much a who's who, and even some you wouldn't expect. And the occasion two weeks ago was their announcement of that Package Analysis Project which, in just one month, they said, had identified more than 200 malicious packages which were present in the Python and JavaScript repositories.

And recall that I was a bit wary at this point of getting too excited about this particular effort, although I applaud the whole concept of an Open Source Security Foundation, as we'll see. But in this case it appeared that they were mostly just scanning, doing static code scanning for references to previously known malicious domains and IPs. Okay. Which would all be trivial to change once it became clear to the bad guys that this was the way to avoid being picked up by this particular detector. So that's not going to take long to fix. That malware will be back under a different name, using an unknown domain, and apparently not go detected. Anyway, we'll see.

But as for the OpenSSF effort overall, I'm very bullish about the prospect of this. It's what has been needed for now quite some time. Those of us who are old enough to have our hair thinning will remember once upon a time when open source software was sort of a counterculture phenomenon, back in the days when source code was not commonly shared for any purpose, and the idea of doing that was kind of bizarre. I mean, even shareware was still closed source. It was just please pay me if you find this useful, but it's still mine. And back then the idea of software being free represented a clear threat to the interests of commercial proprietary software vendors. In fact, in February of 2001, Microsoft's Jim Allchin publicly stated that "Open source is an intellectual property destroyer." He said: "I can't imagine something that could be worse than this for the software business and the intellectual property business." Well, yeah, I guess that's sort of obviously true.

And then early the following year, in January of 2002, one of Microsoft's chief strategists, Craig Mundie, addressing New York University's School of Business, said that releasing source code into the public domain is "unhealthy," causes security risks - yeah, you wouldn't want anybody else to look at your code and find all those bugs - and, he said, "as history has shown, while this type of model may have a place, it isn't successful in

building a mass market and making powerful, easy-to-use software broadly accessible to consumers."

Okay. Well, now that was then. And no one is holding Microsoft responsible for anything that was said 20-plus years ago. The world today is an entirely different place. But it does remind us just how much things have changed in 20 years. And we know that change is slow. We also know that the Open Source model has produced tremendous wealth, both intellectual and economic. I saw somewhere, and it was an old stat, so I didn't add it to the show notes. But years ago it was estimated that $60 billion of wealth had been created and just sort of dumped into the public community by open source software. And as we know, it's become a crucial component of today's software technology landscape, which even Microsoft has now begun to embrace. Today it is entirely possible to operate a major enterprise using nothing but open source software. Which says a lot.

But its problems are also many. The trouble is, as I mentioned at the top of the show, that volunteer effort is much more interested in creating new stuff than in maintaining and securing it. It's not that maintenance and security focuses are absent; but as we've seen, so much maintenance and security focus is needed beyond just getting something to work that it's a big ask. And truly securing software, understanding the many ways in which code which works can still be made not to work, requires an entirely different mindset and a very different type of specific education and training.

So many major organizations are now benefiting from the work that has been done for them that having them join a Foundation so that they have an organized platform for giving something back, especially when it's about improving the crucial security of the software they are now all using within their own enterprises and on their network borders, it's the right thing to do. And the OpenSSF looks like it's the foundation that's going to succeed.

We're talking about this today because last Thursday Google made a major announcement about specific new support for this effort. Google wrote: "Today we joined the Open Source Security Foundation (OpenSSF), Linux Foundation, and industry leaders for a meeting to continue progressing the open source security initiatives discussed during January's White House Summit on Open Source Security. During this meeting, Google announced the creation of its new Open Source Maintenance Crew," that is, the meeting on Thursday of last week. Google announced the Open Source Maintenance Crew, they wrote, "a dedicated staff of Google engineers who will work closely with upstream maintainers on improving the security of critical open source projects. In addition to this initiative, we contributed ideas and participated in discussions on improving the security and trustworthiness of open source software." Which is why our Picture of the Week was so apropos.

"Amid all this momentum and progress," they wrote, "it is important to take stock on how far we've come as a community over the past year and a half. In this post we will provide an update on some major milestones and projects that have launched, and look towards the future and the work that still needs to be done." And I'm not going to share all of it, but just a little, it's sort of the preamble of that.

They wrote: "A little over a year ago we published Know, Prevent, Fix, which laid out a framework for how the software industry could address vulnerabilities in open source software. At the time, there was a growing interest in the topic, and the hope was to generate momentum in the cause of advancing and improving software supply-chain security." And amen to that. So they said: "The landscape has changed greatly since then," a year and a half. They highlighted three points. They said: "Prominent attacks and vulnerabilities in critical open source libraries such as Log4j and Codecov made

headline news, bringing a new level of awareness to the issue and unifying the industry to address the problem."

Second: "The U.S. government formalized the push for higher security standards in May of last year, 2021, with the Executive Order on Cybersecurity. The release of the Secure Software Development Framework, a set of guidelines for national security standards on software development, sparked an industry-wide discussion about how to implement them."

And finally: "Last August, technology leaders including Google, Apple, IBM, Microsoft, and Amazon invested in improving cybersecurity; and Google alone pledged $10 billion over the next five years to strengthen cybersecurity, including $100 million to support third-party foundations like OpenSSF that manage open source security priorities and help fix vulnerabilities."

So I'm finishing their quote, saying: "In light of these changes, the Know, Prevent, Fix framework proved prescient. Beyond just the increased discussion about open source security, we're witnessing real progress in the industry to act on those discussions. In particular, the OpenSSF has become a community town hall for driving security engineering efforts, discussions, and industry-wide collaboration." And again, I will say what I said two weeks ago. I will encourage any of our listeners who are so inclined to go over to OpenSSF.org and poke around. Consider perhaps getting yourself involved.

Google's post goes into greater details about their plans for participation. But I wanted to just follow-up on our introduction of the OpenSSF two weeks ago to note that this is looking like the organization that's going to succeed. Previous efforts were well-meaning but premature; and as history shows, visionaries are often too far ahead of the pack. As the saying goes, they're the ones who get the arrows in their backs just because they're out in front, and they get the problem, but there's just not enough yet mass behind them.

To me it feels like the open source movement is finally being recognized and is earning the respect it deserves. And it may have taken something like the scare of the Log4j vulnerability at the beginning of this year to give major organizations a bit of a wakeup call, to realize just how dependent they had slowly grown on open source solutions through the years. But either way, it appears that it's finally happening now. And, you know, bravo. We really need someone to take a look at the things that have sort of just been created with no oversight and no real focus on security in mind and get them strengthened.

As I have promised, I won't spend lots of our listeners' valuable time discussing boring details of endless ransomware attacks. But when a ransomware gang gets so big for their britches that they suggest that perhaps a government which is refusing to pay their ransom should be overthrown by its citizenry...

Leo: This was so ridiculous.

Steve: I know. I think that rises to a new level of interest.

Leo: Yes.

Steve: I referred to this drama for the first time, mostly in passing, last week. Russia's Conti ransomware gang is behind the attacks on several Costa Rican government

ministries. Over the weekend they doubled their ransom demand from $10 million to $20 million. The Costa Rican operations which have been affected are the Finance Ministry; the Ministry of Science, Innovation, Technology, and Telecommunications; the Labor and Social Security Ministry; the Social Development and Family Allowances Fund; the National Meteorological Institute; the Costa Rican Social Security Fund; and the Interuniversity Headquarters.

In two messages posted to Conti's leak site on Saturday, the gang, which has already leaked 97% of the 670GB stolen during their attacks, claimed the U.S. government was "sacrificing" Costa Rica, and that the country's government should pay for the decryption keys to unlock their systems. As I mentioned last week, Costa Rica's new government had just taken office and immediately declared a state of emergency after refusing to pay the initial $10 million ransom demand issued by Conti. Costa Rica has received assistance from officials in the U.S., Israel, and other countries. And the context for me mentioning all this last week was the U.S. State Department's announcement of a $10 million bounty for information about anyone connected to Conti, with an additional $5 million payable for information leading to an arrest and conviction. So Conti posted: "Why not just buy a key?"

Leo: Yeah, why not? Yeah.

Steve: Yeah. It's only $10 million. Yeah, come on.

Leo: Only $10 million, yeah, no big. Come on.

Steve: And the person wrote in the first person, saying: "I do not know if there have been cases of entering an emergency situation in the country due to a cyberattack." I think they're like, again, getting a little ahead of themselves.

Leo: Is this a ransomware for hire kind of ransomware, Conti? Like it's a service? So this could be just some guy who's being a jerk?

Steve: Good question. I don't think...

Leo: Or is this the Conti group themselves?

Steve: I think this is the Conti group themselves. And actually they sort of mention...

Leo: Probably Putin's henchmen, honestly.

Steve: They mention this. So they said: "In a week we will delete the decryption keys for Costa Rica." And then followed up with a posting: "I appeal to every resident of Costa Rica."

Leo: So evil.

Steve: I know. "Go to your government" - wherever that is - "and organize rallies so that they would pay us as soon as possible." But apparently you only have a week because the decryption keys are going to get deleted. Then it's too late. They said: "If your current government cannot stabilize the situation, maybe it's worth changing it."

Leo: What jerky jerks.

Steve: Like I said, too big for their britches.

Leo: Oh, my god.

Steve: In another message, the group called President Joe Biden a "terrorist," probably as a result of the State Department's new bounty declaration, and said it was raising the ransom to $20 million. The group also implied that it would begin calling government officials to demand the ransom. Yeah, like they've got a spare $20 million in their pockets.

And then, finally, the last message I'm quoting, they said: "Just pay before it's too late. Your country was destroyed by two people. We are determined to overthrow the government by means of a cyberattack. We have already shown you all the strength and power. You have introduced an emergency." Wow.

Leo: So depressing. They're such jerks.

Steve: Yeah. It's true that Costa Rica is limping along at the moment. The attack crippled the country's customs and taxes platforms alongside several other government agencies, even bringing down one Costa Rican town's energy supplier. The country's treasury department has been unable to operate any of its digital services since the attack, making it nearly impossible for paperwork, signatures, and stamps, all required by law, to be processed.

More than three weeks after the attack began, the country is still facing significant struggles, particularly because of the damage done to the Finance Ministry. Last week the country told its residents that taxes need to be calculated by hand and paid in person at local banks, as opposed to the digital system the country had previously used. So back to the Stone Age. Or pre-Internet, at least. Wow. And yes, Leo, as you said, they're not going to get any money from this. Costa Rica will get outside assistance and limp themselves back into existence. And nothing will come of it.

Okay. Policing the Google Play Store. There's a probably intractable problem with the model we currently have - and I can't think of a better one, so I'm not criticizing the model, I'm just elucidating - for freely downloadable mobile device apps created by individuals lacking a reputation. After all, everyone starts off with no reputation. Android handsets are available for a fraction of the price of Apple's devices, and Android users typically cite the expansive freedom provided by the Android platform as their primary reason for preferring that much more open mobile environment. But those listening to this podcast realize that with that freedom comes significantly increased danger.

I think it's clear that Google is doing the best they can to minimize this danger. But a continuous daily incoming torrential flood of apps is arriving at the Google Play Store, and there is just no way for Google to deeply research the behavior in each and every

one of these apps. And to provide the useful and powerful freedom that Android users demand, apps must be given powerful enough access to the underlying hosting platform that a malicious app could be quite abusive. So Google is always stuck playing catch-up. And in addition to their efforts, thank goodness, they're able to rely upon the motivations and scrutiny that is also being offered by third-party security companies.

Yesterday, Trend Micro posted their piece titled "Fake Mobile Apps Steal Facebook Credentials & Cryptocurrency-Related Keys." In their article, Trend Micro explained that malware that's expressly designed and intended to steal the Facebook logon credentials of Android phone users continues to pop up on the Play Store. Such malware has become so commonplace, in fact, that it's now being called "Facestealer" malware. But it doesn't say that on the cover, of course. It's hidden in apps that otherwise look harmless, compelling, and are of course completely free.

Trend Micro recently identified more than 200 Facestealer variants in the Google Play Store, notified Google, and Google took them down. But how long will it be before they're replaced by another 200? Some of the apps that were just taken down had been installed more than 100,000 times. The apps take the form of tools for editing, manipulating, or sharing photos, but they can take many other forms. An example was "Daily Fitness OL" which appears to be a fitness app complete with exercises and video demonstrations. I looked at the screenshots of this thing. It's gone now. But it looked, like, completely convincing. But it was entirely designed to steal the Facebook credentials of anyone who used it.

The so-called Facestealer apps were first identified in July of last year and have been linked to Russian servers by researchers with the mobile security company Pradeo. Attackers typically use the compromised Facebook accounts they acquire for various malicious purposes such as phishing scams, fake posts, and ad bots. In the case of Daily Fitness OL, users are prompted to log into Facebook through an embedded browser. Okay, what could possibly go wrong with that; right? Then, not surprisingly, a piece of JavaScript is injected into the loaded webpage which, of course, steals the logon credentials entered by the user. Easy peasy.

Trend Micro identified many other Facestealer apps with names like Enjoy Photo Editor, Panorama Camera, Photo Gaming Puzzle, Swarm Photo, and Business Meta Manager. And in addition to these 200-plus Facestealer apps, Trend Micro noted they had also found 40 fake cryptocurrency mining apps that are designed to steal their users' cryptocurrency, not only what it mines, but what was there before. Last month Google reported that last year they had removed more than one million malicious apps from the Play Store. Think about that. One million malicious apps in 2021 alone. An intractable problem indeed.

And the trouble is, there is next to zero general awareness of this problem among the Android-using population. There are presently more than three billion - with a B - active Android devices being used worldwide. There's no question that the majority of Google Play Store apps are legitimate and well meaning. But when a malicious app is only removed after having been downloaded and installed more than 100,000 times, it's also clear that downloading Android apps carries a non-zero risk, and it's not clear that there's anything that can be done. Again, it's a nice open model, but it is under constant relentless attack.

Speaking of relentless attack, the situation has indeed grown more dire for F5 Systems' BIG-IP boxes. The day after we talked about this last week, CISA added that recently disclosed F5 BIG-IP flaw to its Known Exploited Vulnerabilities Catalog following reports of its active abuse in the wild.

The problem is CVE-2022-1388, bearing a well-deserved CVSS of 9.8, due to a critical bug in BIG-IP's iControl REST endpoint, which provides an unauthenticated attacker with a method to execute arbitrary system commands. And I actually saw one that was posted in some of the write-ups. They posted that one of the commands that was being issued was "rm -rf" - that "-r" as in recurse - *, which of course will remove all the files on the system, in the file system, starting from the device's root. Wow. Anyway, the firm Horizon3.ai wrote: "An attacker can use this vulnerability to do just about anything they want to the vulnerable server. This includes making configuration changes, stealing sensitive information, and moving laterally within the target network."

Although patches and mitigations for the flaw were introduced by F5 on May 4th, the Wednesday before last, we know how that tends to go. And in fact the F5 boxes have been subjected to in-the-wild exploitation ever since F5's announcement was followed by reverse engineering the fix and then going to town. Some attackers attempted to install web shells that would grant backdoor access to the targeted systems, and others simply destroyed the device's usability by executing that recursive "rm," the remove files command, across the entire file system from the root outward.

Rapid7 wrote: "Due to the ease of exploiting this vulnerability, the public exploit code, and the fact that it provides root access, exploitation attempts are likely to increase." But their security researcher Ron Bowes added that: "Widespread exploitation is somewhat mitigated by the relatively small number of Internet-facing F5 BIG-IP devices." And, yeah, those that are still surviving on the Internet because, once wiped, they're no longer a BIG-IP device. They're just a machine with no mission. The SANS Internet Storm Center (ISC) wrote on Twitter that: "Given that the web server runs as root, this should take care of any vulnerable server out there and destroy any vulnerable BIG-IP appliance." And indeed that's what's happening.

Pursuant to CISA's addition of this vulnerability to their catalog, all Federal Civilian Executive Branch agencies have been mandated to patch all systems against this issue by May 31st. But that's two weeks from today. And of course by that time there will be nothing left standing to patch. I did see GossiTheDog also tweet that he had used - I'm blanking on this service, the scanning service. Oh, Shodan. He had used Shodan to find the machines. They were discoverable using Shodan.

Leo: Oh, boy.

Steve: So, you know, it's just a matter of...

Leo: They're out there.

Steve: Yeah, no time before they'll all be gone. Okay. I have a piece of errata. Not often, but important we do this. Since it's an interesting and important topic that's perfect for this podcast, I want to take a moment to talk about classical computing, quantum computing, and symmetric versus asymmetric cryptography.

Back in 1994, an American mathematician by the name of Peter Shor conceived of an algorithm for quantum computers which would be able to determine the prime factors of integers. That algorithm worked, and it bears the name "Shor's Algorithm." Wikipedia explains that: "The efficiency of Shor's algorithm is due to the efficiency of the quantum Fourier transform and modular exponentiation by repeated squarings." They write: "If a quantum computer with a sufficient number of qubits [quantum bits] could operate without succumbing to quantum noise and other quantum-decoherence phenomena, then

Shor's algorithm could be used to break public key cryptography schemes, such as The RSA scheme, the Finite Field Diffie-Hellman key exchange, and the Elliptic Curve Diffie-Hellman key exchange."

In other words, I was incorrect to state last week that the use of elliptic curve crypto was "post-quantum" safe. It's generically any asymmetric public-key crypto that we're currently using that is not safe. And I know better, so I wanted to correct the record. It's symmetric crypto, interestingly enough, that remains safe in a post-quantum crypto world.

Leo: What uses symmetric?

Steve: Well, everything does. We start with asymmetric in order to share the key. Then that key, because asymmetric encryption is so slow, we don't actually do the bulk encryption and decryption asymmetrically. We only use the asymmetric encryption for the key. So that decrypts the key. Then we use symmetric encryption, like Rijndael AES, to perform the actual bulk decryption or encryption.

Leo: Because with symmetric the real issue is the transfer of that key.

Steve: Correct. Correct.

Leo: So you use public key to kind of get the symmetric key across.

Steve: Precisely.

Leo: And then you can continue with the symmetric key.

Steve: Precisely. So Wikipedia explains: "RSA is based on the assumption that factoring large integers is computationally intractable. As far as is known, this assumption is valid for classical non-quantum computers. No classical algorithm is known that can factor integers in polynomial time. However, Shor's algorithm shows that factoring integers is efficient on an ideal quantum computer, so it may be feasible to defeat RSA by constructing a large quantum computer. It was also a powerful motivator," writes Wikipedia, "for the design and construction of quantum computers" - yeah, let's crack crypto, that would be good - "and for the study of new quantum computer algorithms. It has also facilitated research on new cryptosystems that are secure from quantum computers, collectively called 'post-quantum cryptography.'"

Okay, now, Leo. The good news is - you're going to get a kick out of this - from a practical standpoint it still looks like we're well away from the quantum crypto apocalypse, since Wikipedia also reports on the recent progress being made in quantum prime factorization. They write: "In 2001, Shor's algorithm was demonstrated by a group at IBM."

Leo: Wait'll you hear this. Go ahead. Go ahead. What did they factor, Steve?

Steve: "Who factored 15..."

Leo: Fifteen, the number 15.

Steve: "...into 3 times 5."

Leo: Wow.

Steve: "That used nuclear magnetic resonance implementation of a quantum computer with 7 qubits." Now, progress; right? "After IBM's implementation, two independent groups implemented Shor's algorithm using photonic qubits..."

Leo: Ooh.

Steve: Yeah, that's some fancy photonics out there.

Leo: They could get a big number there, huh?

Steve: Well, "...emphasizing that multi-qubit entanglement was observed when running the Shor's algorithm circuits."

Leo: Is that good or bad?

Steve: Who knows?

Leo: Seems like entanglement would be bad, but okay.

Steve: I mean, this is all deep voodoo.

Leo: I know.

Steve: Okay. Now, so 11 years later, in 2012, the factorization of 15 was performed with solid-state qubits. So, okay.

Leo: Oh ho.

Steve: Yeah. Also in 2012 - wait for it - the factorization of 21 was achieved.

Leo: Oh, my god.

Steve: Setting the record, Leo.

Leo: Wait a minute, was it 7 and 3?

Steve: Uh, that's good. Those are both prime.

Leo: I did that in my head. I'm faster.

Steve: That's, you know, you are faster than a quantum qubit. That set the record for the largest integer factored with Shor's algorithm.

Leo: What is the largest integer ever factored?

Steve: Well, now, here's the problem.

Leo: Yeah.

Steve: Three years ago - now we're up to three years ago, just in 2019 - an attempt was made to factor the number 35.

Leo: Thirty-five, wow.

Steve: Yeah.

Leo: "Attempt" sounds like they couldn't do it.

Steve: Yeah, unfortunately it was using Shor's algorithm on an IBM Q System One. I hated Q, I told you.

Leo: Yeah, yeah.

Steve: But the algorithm failed. We were unable to factor, could not factor 35.

Leo: Wait a minute. Wait a minute. Shor's algorithm doesn't work?

Steve: And Leo, it's not fair if you give IBM Q System One a hint on the prime factorization of 35.

Leo: They're two prime numbers.

Steve: That would be cheating. Don't give it a hint. The algorithm failed because of accumulating errors before we got to the answer.

Leo: So it's not the algorithm's fault. It's those damn quantum bits.

Steve: You know, they're a little fuzzy on the edges. And so they're not, you know, you want a zero, or you want a one. You don't want, like, something in between there.

Leo: No, yeah.

Steve: So these algorithms are similar to classical brute force checking for factors. So unlike Shor's algorithm, they are not expected to ever perform better than classical factoring algorithms. Okay. So.

Leo: The whole point of Shor's is down the road, using many, many qubits - we're going to find out how many in a moment, and you will love the number - this is going to be better than just brute force.

Steve: Well, better than prime factorization the old-school way, right.

Leo: Right.

Steve: So quantum computers...

Leo: Using Newton's method or whatever it is.

Steve: ...have successfully factored the four-bit value of 15. Now they've done it several times. And Leo, you did it right here in front of us, which was astonishing.

Leo: Whew. I thought that was impressive, wasn't it, yeah.

Steve: Wow. And broke the record by factoring the five-bit value of 21 using Shor's algorithm.

Leo: Oh, nice.

Steve: However, they thought, hey, we're on a roll here. We did four bits. Then we got five bits. In an effort three years ago in 2019, couldn't quite make it to six bits to factor 35.

Leo: Darn it.

Steve: Which is binary 100011.

Leo: Is that supposed to help me? Because I think it's 7 and 5. But I might - I could be wrong.

Steve: Oh, Leo, you spoiled it. We had to wait. We had to wait for the answer.

Leo: Okay.

Steve: To come, you know.

Leo: Okay, fine.

Steve: Now, one of our listeners is a crypto-aware physicist who wrote after last week's podcast.

Leo: Oh, good, good.

Steve: And he raised a couple of very good points which I want to share. We'll get to him in a minute. But I want to finish up on the asymmetric versus symmetric crypto question. Elsewhere, Wikipedia notes that: "In contrast to the threat quantum computing poses to current public key algorithms" - and again, we should not be worried, we don't need to change our passwords, everybody's okay - "most current symmetric cryptographic algorithms and hash functions are considered to be relatively secure against attacks by quantum computers. While Grover's algorithm" - I think he was that green creature, wasn't he, on 'Sesame Street'? "While Grover's algorithm does speed up attacks against symmetric ciphers, doubling the key size can effectively block these attacks. Thus," Wikipedia concludes, "post-quantum symmetric cryptography does not need to differ significantly from current symmetric cryptography."

And as we know, we already periodically double the lengths of our symmetric crypto keys and hashes as the speed of traditional computation begins to narrow their practical security margins. So anyway, we can all collectively, Leo, breathe a sigh of relief. I absolutely want to correct the record. It's not that ECC is better.

Leo: It's just as vulnerable.

Steve: Yes.

Leo: Basically any asymmetric scheme is just as vulnerable. Is that what you're saying?

Steve: Well, there is, for example, lattice-based crypto. There are asymmetric post-crypto. So it's not...

Leo: Yeah, because we kind of need asymmetric.

Steve: Yes. Yes. We need it, and we're going to have it. And already the mathematicians are loving the fact that we're now at, well, not quite to six bits. But that lets them dangle the threat in front of the budget committee in order to get more funding because maybe we'll get to six, and maybe to seven bits.

Leo: The problem right now is just expanding the number of bits in the key, right, to 512, 1024, 2048.

Steve: And Leo, at some point, we will get to a bit number that you will not be able to factor in your head. And that will be a day to celebrate.

Leo: I think we've gotten past that point, but okay. What did your physicist friend say? Because I'm very curious. He's a crypto expert as well as a physicist, so he would have a good handle on all of this.

Steve: His name is Alim, and he's @dutchphysicist on Twitter.

Leo: Oh, okay.

Steve: He said: "Hi, Steve. Hope that all is well with you. As my weekly routine, I listened to your last podcast episode where you discussed Biden's memorandum on Quantum Computer threats on classical cryptography. Being a physicist by education (Ph.D.) and having practiced PKI-related IT work" - public key infrastructure - "for the past five to six years, I wanted to make a few remarks on your comments." And I'll just preface this by saying they're good ones.

"It is not only," he wrote, "RSA (based on difficulty of factorization problem), but also ECC (based on the discrete logarithm problem) which is vulnerable to Shor's algorithm, but a powerful enough quantum computer." He said: "Recently, Bruce Schneier also referred to an academic article where the authors discussed how much qubit capacity is required to achieve a reasonable attack on the Bitcoin blockchain."

Leo: Oh. There you go. That's an interesting problem.

Steve: Ah, Leo. You know, is that seven bits? Is that eight? You need 10? How many you need?

Leo: Is that also - is blockchain based on prime factorization? It's the same idea? I don't know.

Steve: That's a good point. He says: "Indeed, the required number of physical qubits is tremendous, on the order of 10^6." So I think the blockchain also is safe because we don't yet have six, let alone 10^6.

Leo: Because that's a lot. Okay, yeah.

Steve: Yeah. That would a million qubits. And they can't be fuzzy little bits. They've got to be, you know, they have to be sure of themselves.

Leo: Well, stability is a big problem with these.

Steve: You've got to be sure of your bitness, yeah.

Leo: Okay, all right.

Steve: "On the other hand, we know how fast it went," he says, "with the traditional silicon technology. Remember Moore's law." I'm not sure that applies here.

Leo: I don't know if this applies, yeah.

Steve: It feels too - I don't think it does. But the point here is now he really raises some good ones. He says: "I find Biden's statement correct from some aspects. First, there is an attack called 'store now, decrypt later.'" Of course we talked about that years ago.

Leo: And that's why you want perfect forward secrecy; right? Because yes.

Steve: No, actually.

Leo: No, that doesn't solve it?

Steve: That doesn't work for that.

Leo: Oh, okay.

Steve: Yeah. He says: "Any stored information today can be broken by a powerful quantum computer in the future. Therefore, any confidential information that should stay confidential for a long period of time should today be protected against quantum computers of tomorrow." And again, I mean, that's a good point. Also, and this is really good, everyone will get this, he wrote: "Achieving crypto agility is very difficult. Most cryptographic algorithms are embedded deep in the protocols and products. There are even cases that DES algorithm is still used in SIM cards and payment cards today."

Leo: Right, right. And that's been broken for years, yeah.

Steve: Yeah. "This makes the lifetime of such algorithms very long," he says, "i.e., 40 years or more. Therefore," he writes, "Biden's warning on the federal agencies, and thus

the industry, is an early call for a very hectic and difficult transition." He said: "Though I also see some issues with Biden's statement." He wrote: "Two, NIST's competition on post-quantum algorithms has not announced the winners yet, and standard finalization is still a few years down the road. From this perspective," he said, "any organization to attempt to implement a post-quantum solution is a premature action. Things may still change." Meaning let's not be too quick to force NIST to choose a winner here.

Leo: Right.

Steve: And he finally said: "I sincerely hope that Biden's administration will not implicitly try to pressure NIST to finalize the competition by this statement. Just a few moments ago a weakness was found and reported in one of the post-quantum digital signature algorithms," he said, "Rainbow. There should be no political pressure on such standardization activities, and researchers should be left free in making their decision and given enough/proper time."

Leo: So we don't really have a quantum-safe algorithm yet.

Steve: No. We haven't, like, NIST has not standardized on the way we have, like, Rijndael, which is now our chosen AES.

Leo: And his point is you don't want pressure on NIST to propose something prematurely. You really, I mean, you want to get this right.

Steve: Yes. And get people using it. The concern is that something will be standardized, and there will be a political mandate to start using it because, as he noted, there was just a problem found in one of the post-quantum digital signature algorithms.

Leo: Yeah, so much for that. Okay.

Steve: So, like, yeah, we don't want that to be in use when the problem is found. And again, it's like, you know, 35 we still can't factor. So just, you know, but he's also right about it taking - it being so difficult to change this stuff.

Leo: Yeah. Yeah.

Steve: So he finished, saying: "Needless to say, I definitely share your opinion that there are way more fundamental issues to be addressed urgently," which was the point I was making when I was kind of poking fun at this last week. He said: "However, I believe that quantum computer threats on cryptography is very serious and should be given enough attention due to slow adaptation or adoption of new cryptographic algorithms in billions of computers, protocols, et cetera."

Leo: But you can't rush it, either.

Steve: Exactly.

Leo: So we're just going to keep looking for a post-quantum technology, and then I'm not too worried about my SSH keys being cracked by a quantum computer anytime in the next - in my lifetime, probably.

Steve: Apparently it's got a ways to go.

Leo: Yeah. But important point, the fact that I switched to elliptical keys isn't really germane.

Steve: Yeah. Actually, the elliptical keys are nice, mostly because they're so much shorter.

Leo: They're small, yeah.

Steve: They're much shorter. It's much...

Leo: Yeah, I used to have to paste a paragraph in.

Steve: Right.

Leo: Now it's one line. So that's, I mean, that's a silly reason why. And that 512 bits is better than the 2048 bits I had in RSA, I presume. Or as good as. Okay. All right.

Steve: Yeah. Also I got, just for the record, I got a bunch of tweets from people saying, "Steve, you'd better look at ECC again and see if you still think it's quantum-safe."

Leo: But the thing I was worried about is somebody saying, oh, you're so wrong, quantum computing is just around the corner. I don't see that yet. But, you know.

Steve: And again, I think we framed it exactly right. Yes, we need to be thinking about it. Yes, it is advancing. Yes, it is advancing slowly. But yes, it does take a long time to take existing crypto out of circulation.

Leo: Yeah.

Steve: And, you know. So I would say as soon as we absolutely know that we have a bulletproof post-quantum crypto, where problems are not going to be found in it - and, see, and that's the other problem is we were, when we went through the Rijndael competition, or the AES competition which ended up choosing Rijndael, it was a whole bunch of ciphers. And they were able to say, well, we used a Feistel network here, and we ran that through a triple scrambler, and blah blah blah. I mean, like our

understanding of how to do symmetric crypto, which is what AES and Rijndael are, it's so mature that we were like taking building blocks and mixing and matching them and understanding in detail how they worked. We're in a whole new environment now with post-quantum crypto. And we're at risk of making a mistake. And that's almost worse than prematurely obsoleting something that isn't broken today.

Leo: Right.

Steve: So, yeah.

Leo: Yeah.

Steve: I got a nice note from someone whose name I thought I ought to redact, even though he didn't say I should. He said: "In Security Now! you've reported on a number of cybersecurity initiatives that the federal government has introduced this past year, including CISA's 'Known Exploited Vulnerabilities Catalog,' Congress's 'Strengthening American Cybersecurity Act,' and the White House's 'Executive Order on Cybersecurity.' What I haven't heard you mention are the TSA's two 'Security Directive Pipeline' memorandums. These are two successive directives, issued in response to the Colonial Pipeline compromise, that impose explicit cybersecurity requirements upon the midstream oil and gas pipeline industry."

And he said: "One of the lesser-known regulatory mandates of the TSA," he says, "yes, that TSA, is the safety of interstate pipelines." He says: "I work in the midstream pipeline industry, and these TSA directives have been the bane of my existence for the better part of a year. I'll reserve specific criticism, but will offer a recent Politico article which summarizes the situation nicely. Unfortunately, I'm not able to go into many particulars because the government, in its infinite wisdom, has marked the entire second directive (SD02) as Sensitive Security Information which prevents me from publicly divulging details. Suffice it to say that, yes, the government has instituted a cybersecurity standard that a segment of critical infrastructure must adhere to, but that can't be discussed except behind closed doors."

Leo: See, that seems like a bad idea.

Steve: I could not agree more. Yup.

Leo: I understand the notion of security through obscurity. But honestly, anything that you're doing for security should be tested.

Steve: Needs oversight, yes, yes.

Leo: Other experts have to look it over, I think.

Steve: He said: "One tidbit that I'm compelled to share is the role that CISA's Known Exploited Vulnerabilities Catalog plays. SD02 requires that pipeline operators patch vulnerabilities published in the Catalog within certain timeframes. Since you've

mentioned the Catalog in several Security Now! episodes, I wanted to call out the fact that this applies not just to government entities, but also to private pipeline companies. And yes, we are forced to review the list daily for new additions."

And he finished: "Thank you for all you do, and especially for a wonderful and informative weekly podcast." And I will say back to him, thank you. You've just contributed to making it more informative.

Leo: Nice.

Steve: So Liam Lynch tweeted from @L2actual, which I thought was a cool handle. He said: "Hi, Steve. I only listened to Episode 869 the other day and heard you and Leo again refer to the GDPR as being the cause of cookie notices. I kept meaning to contact you about this, as the only thing the GDPR did for cookie notices was to strengthen the consent requirement." On the other hand, I would argue that is what created the notices. He said: "Cookie notices have been around for a lot longer than the GDPR has been in force, as they came from the ePrivacy directive."

Leo: Yeah. But we ignored it until GDPR.

Steve: Exactly. And then he cited a thread. So I wanted to for the record put that out there. But again, it was the GDPR that gave it teeth and then forced it to be such an annoyance for all of us.

Leo: It is kind of the canonical example of overregulation or privacy regulation gone wrong, I think.

Steve: Boy.

Leo: Yeah, yeah.

Steve: And I wanted to officially note that the work on SpinRite's backend has officially finished.

Leo: What?

Steve: SpinRite's new hardware drivers are working without any exception that the group's extensive testing has revealed. In the case of two very old systems, they were like back when you could set the DMA speed in the BIOS. You remember, like you could set it to 0, 1, 2, or 3 or something. They were also the first VIA implementation chipset. In two very old systems it was necessary to turn off the Ultra DMA setting in the BIOS to obtain reliable transfers. SpinRite detected that they weren't reliable and refused to operate otherwise. But when that was done, everything worked perfectly.

Throughout this work, SpinRite's new and much-improved benchmarking was used to exercise those backend drivers through the IO abstraction that I've talked about before. So what that does is effectively isolate any backend devices from the front-end code.

When SpinRite 7 adds native hardware USB drivers and then NVMe drivers, nothing else about SpinRite's front-end needs to change because the IO abstraction provides a uniform interface to the front-end code. The benchmark was the first client of that IO abstraction. The actual SpinRite machine, with its multiple switchable screens, the grid display, the DynaStat data recovery, the detailed technical log and all the rest will be the second and final client. So that's where I now turn my focus. What we've just slogged and fought through has been by far the longest and toughest part, since it was where all of the hardware and machine dependency was. Now that's all behind us.

Since the BIOS was historically SpinRite's IO abstraction, which is now gone, I now have a lot of rewriting to do to support SpinRite's new abstraction. But it's not the sort of thing that will need constant interaction and iteration and tireless testing, the way the previous work on the backend did. I'm sure that the SpinRite testing gang will end up finding things I've missed and will have ideas for improvements when they begin to see something that is actually operating, which is what will happen next. But at this stage they're mostly going to be waiting for me, rather than me waiting to learn from them how the latest test release turned out. So we're getting close.

Leo: Getting very close. Very exciting. Yay. All right, Steve. I'm very curious.

Steve: Okay.

Leo: We've been of course reporting on this story for the last week. We talked about it Sunday on TWiT, on TWiG on Wednesday. But I'd like to get your take on the new EU rules here.

Steve: So the title of today's podcast, "The New EU Surveillance State," might seem hyperbolic. But just wait till you hear what the EU is proposing. The European Union's proposed new legislation will not only require scanning encrypted communications for child sexual abuse material content, but believe it or not, actually reading all text messages with the goal of detecting any textual content that might be regarded as "grooming" a minor.

Leo: Ugh. That's such a nebulous term.

Steve: Oh, Leo, it's impossibly nebulous. Okay. So those who haven't read far into the legislation quickly and correctly recognize that accomplishing any of this is inherently, necessarily, and unavoidably violating - it requires some agency or entity to scrutinize all communications capable of conveying any graphical or textual material. In other words, all of the social messaging platform used by European Union citizens. And such scrutiny necessarily contravenes the well-established goals, intents, and capabilities of end-to-end encryption. So, yeah, this would be the end of true meaningful privacy enabled and facilitated by end-to-end encryption.

But reading some of the proposed legislation, as I did, one discovers that it also requires that this surveillance goes beyond the matching of previously known content hashes to also include content that has not been previously seen. So this would require either humans to view everything that everyone sends to anyone, or to train up machine vision and learning models to automate the identification of previously unknown child sexual abuse material.

Okay. Listen to what Johns Hopkins cryptographer Matthew Green tweeted upon hearing of this last week. Mathew tweeted: "This document is the most terrifying thing I've ever seen. It is proposing a new mass" - I mean, Leo, as I'm saying this, it sounds like fiction. It sounds like I'm making this up. This is the actual legislation. He says: "It's proposing a new mass surveillance system that will read private text messages, not to detect CSAM, but to detect 'grooming.'"

Leo: That's the ridiculous part. CSAM you could pretty much know that's CSAM. But if I ask you, Steve, what's your age, sex, and location, am I grooming you?

Steve: Right. Or what are your pronouns?

Leo: Yeah. It's very contextually dependent. And I think it's a judgment call.

Steve: So it actually says, it actually says in the legislation...

Leo: And, by the way, that eliminates E2E encryption entirely because they say encryption's no excuse.

Steve: No. In fact, they say we still want the best encryption possible. Okay. But we also want this. So here's what Matthew highlighted from the legislation. This is the legislation: "As mentioned, detecting 'grooming' would have a positive impact on the fundamental rights of potential victims especially by contributing to the prevention of abuse; if swift action is taken, it may even prevent a child from suffering harm. At the same time, the detection process is generally speaking the most intrusive one for users (compared to the detection of the dissemination of known and new child sexual abuse material), since it requires automatically scanning through texts in interpersonal communications."

I mean, so the point is the legislators know this. And they continue: "It is important to bear in mind in this regard that such scanning is often the only possible way to detect it; and that the technology used does not understand the content of the communications, but rather looks for known pre-identified patterns that indicate potential grooming. Detection technologies have also already acquired a high degree of accuracy" - and they cite something, it's footnote 32 - "although human oversight and review remain necessary, and indicators of grooming are becoming ever more reliable with time, as the algorithms learn." Give me a frigging break.

Leo: It's pretty horrible.

Steve: Oh, Leo. Matthew then issued a series of tweets in a thread, which I'll read. He said: "Let me be clear what this means. To detect grooming is not simply searching for known CSAM. It isn't using AI to detect new CSAM, which is also on the table. It's running algorithms reading your actual text messages to figure out what you're saying, at scale. It is potentially going to do this on encrypted messages that should be private. It won't be good, and it won't be smart, and it will make mistakes. But what's terrifying is that once you open up 'machines reading your text messages' for any purpose, there are no limits. Here is the document. It is long but worth reading because it describes the

most sophisticated mass surveillance machinery ever deployed outside of China and the USSR. Not an exaggeration."

The link Matthew shared last week was from a leak of the official legislation, which then appeared the next day. They are the same, I looked at them both, 135-page document. The title is "Proposal for a REGULATION OF THE EUROPEAN PARLIAMENT AND OF THE COUNCIL laying down rules to prevent and combat child sexual abuse."

As I said, the legislation is 135 pages. But its first two paragraphs set the stage, and they're worth sharing. So here's the first two paragraphs: "The United Nations Convention on the Rights of the Child (UNCRC) and Article 24(2) of the Charter of Fundamental Rights of the European Union enshrine as rights the protection and care of children's best interests and well-being. In 2021, the United Nations Committee on the Rights of the Child underlined that these rights must be equally protected in the digital environment. The protection of children, both offline and online, is a Union priority.

"At least one in five children falls victim to sexual violence during childhood. A 2021 global study found that more than one in three respondents had been asked to do something sexually explicit online during their childhood, and over half had experienced a form of child sexual abuse online. Children with disabilities face an even higher risk of experiencing sexual violence. Up to 68% of girls and 30% of boys with intellectual or developmental disabilities will be sexually abused before their 18th birthday.

"Child sexual abuse material is a product of the physical sexual abuse of children. Its detection and reporting is necessary to prevent its production and dissemination, and a vital means to identify and assist its victims. The pandemic has exposed children to a significantly higher degree of unwanted approaches online, including solicitation into child sexual abuse. Despite the fact that the sexual abuse and sexual exploitation of children and child sexual abuse materials are criminalized across the EU by the Child Sexual Abuse Directive 6, adopted in 2011, it is clear that the EU is currently still failing to protect children from falling victim to child sexual abuse, and that the online dimension represents a particular challenge."

Well, okay. One of the most difficult lessons an ethical person learns and must come to terms with as they grow is that not all problems have workable solutions. And it doesn't matter at all how big the problem is nor how much we want there to be a good solution. The entire problem here can be broken down into two separate issues. First, in order for some overviewing agency to obtain the raw data to be scrutinized, there can be no true and meaningful privacy between digitally communicating endpoints or individuals. That must end. Period. Everything needs to be visible and visited. And it's not sufficient to only surveil the devices used by minors because the original intent of detecting child sexual abuse material was to discover and apprehend those non-minors who were actively trading in such illegal content, thus curtailing the demand for the creation of more material. This means that all of everyone's social media messaging content must pass through surveillance filters. Which brings us to the second issue, what to do with that content once it's obtained.

Matthew put it bluntly in one of his tweets. He said: "It is going to do this on encrypted messages that should be private. It won't be good, it won't be smart, and it will make mistakes." Even if we agreed to voluntarily relinquish all of our privacy rights, it's not at all clear that the world has the technology to do what the EU's governing legislators want. It's easy for them to write a law stating what they want. But wanting it doesn't will it into existence, no matter how fervently and sincerely they want it. So the technology is going to miss things that it should catch and flag things that it should not. Humans will be required to examine the previously private photos that some image classifier believes to be salacious, and previously private text messages shared by consenting adults will be open to others' scrutiny.

And then there's the devil's advocate side which is also absolutely true and well-established. Cryptography has already escaped. The algorithms which are able to unbreakably encipher plaintext are already public. So if the use of truly unbreakable end-to-end encryption is outlawed, then only the outlaws will be using it. And, yes, that too could be prevented. The next step in this escalation to doom would be for the communications carriers to refuse to transit any encrypted communications they cannot themselves decrypt. That's possible. In which case we might as well just turn back the clock to the 1970s and give up because the Internet would no longer be useful for commerce.

Matthew also noted the other elephant in the room by tweeting: "But what's terrifying is that once you open up 'machines reading your text messages' for any purpose, there are no limits." The distasteful issue of child sexual abuse is certainly real. But it has also been observed that it serves as a convenient stalking horse for governments' much broader interests in monitoring and controlling speech of many other kinds. Much of such speech could be criminal. But much that might be of interest to censors would not be.

The EU's governors are wrong to want this legislation which can only be characterized as dangerous, wholly impractical, and impossible to implement as they hope. So all we can do is pray that it dies.

Leo: And that's a possibility. It's not law now.

Steve: Correct.

Leo: It's just a proposal.

Steve: Correct. And, you know, the good news is it's now being aired. Everybody is talking about it. The EFF just, I mean, they just melted down, as you can imagine, over this. And, boy, I mean, as we've said, when we were crossing into this next decade, encryption is this massive challenge. I mean, it just is. The governments do not like the idea that they have no means for seeing what their citizenry is saying. And, I mean, and both sides have valid positions. It's a tough one.

Leo: Learn how to write your own encryption, kids. You might want to invest in your own telecommunications network while you're at it.

Steve: Well, and would somebody who wanted to abide by the law feel okay about doing it? I couldn't do that. I mean, first of all, I'm just texting Lorrie when I'm going to be home for dinner.

Leo: Right.

Steve: So it's not like I'm caring about that that much. There's nothing I'm doing. But we've been given this, and now it's looking like the EU wants to take it away. I mean, Leo, literally, you have to, in order to do this "grooming detection," everything someone types on their keyboard of their computer and their mobile device, it has to go through some censor's screening filter.

Leo: Yeah.

Steve: Or the system isn't providing what the law requires. And we know that users don't want them on their own phones. Apple tried that, and everyone went, you know, pinched their nose and said, "Eww. I don't want any form of kiddie porn on my phone in order to keep Apple from being involved." So that says you have to have a third party somewhere in a cloud that all this runs through.

Leo: Yeah.

Steve: So you have to have a man in the middle, a sanctioned man in the middle able to decrypt this in order for, you know...

Leo: Unbelievable. What a horrible idea. All right, my friend. We've wound people up enough. I think it's time to stop. That is Steve Gibson right there, man. That's the guy. If you like this show, go to his website, GRC.com. Pick up a copy of SpinRite, the world's best mass storage, I almost said surveillance, mass storage maintenance...

Steve: It's not.

Leo: No, no surveillance. And recovery utility available at GRC.com. Pick up 6.0. You'll get 6.1 free if you buy it today. And you get to participate in the development. We're just, obviously, just around the corner. While you're there you can also get this show. He has two unique versions of this show at his website, a 16Kb audio file for the bandwidth-impaired and really nicely done transcripts which you can read along as you listen, or use to search for parts of any of the 871 shows. That's all at GRC.com. You can leave him feedback at GRC.com/feedback, or slide into his DMs on Twitter. He's @SGgrc.

We also have copies of the show at our website, 64Kb audio, we have video, TWiT.tv/sn. There's a YouTube channel also devoted to Security Now!. If you want to share clips with friends, that's probably the easiest way to do it. You can also subscribe with your favorite podcast client, get it automatically the minute it's available. We do the show Tuesdays, 1:30 Pacific, 4:30 Eastern, 20:30 UTC. You can watch us do it live. We stream it live behind the scenes.

Steve: Right now.

Leo: Right now.

Steve: It's happening right now.

Leo: We're doing it as we speak. Unless you're watching later, and then in this case it's not live. You can watch that at live.twit.tv in those hours. If you're watching live, chat live at irc.twit.tv. Club TWiT members get their own special Discord.