
# National Public Data

**Description:** As we embark on our 20th year of this weekly Internet security and privacy-oriented technical news podcast, we're going to look at some more interesting certificate revocation news, and we have an experiment for our listeners. What six zero-days were patched during Microsoft's Patch Tuesday last week? Fifty-three episodes of the 1980s "Famous Computer Caf" radio show were recently discovered and are now online; hear Bill Gates before his voice changed. We have Release #3 of IsBootSecure, and a GRC email update, and some interesting listener feedback. Then, to no one's surprise, we're going to take a deep dive into the background, meaning, and impact of the largest personal data breach in history: How to look up your own breached records online, what to do, and what this means for the future.

High quality (64 kbps) mp3 audio file URL: http://media.GRC.com/sn/SN-988.mp3
Quarter size (16 kbps) mp3 audio file URL: http://media.GRC.com/sn/sn-988-lq.mp3

SHOW TEASE: It's time for Security Now!. Steve Gibson is here. We've got a very big show for you. Ninety, count 'em, 90 fixes on Patch Tuesday last week. Steve will count them all. No, he won't. Ninety-nine fixes on the wall, no. He's going to talk about a few of them. We've got a great Picture of the Week that explains how RAID works, sort of. An update on the certificate revocation issues. And then, finally, a look at the biggest data breach we think of all time. How to find out if you're part of the NPD breach, and what to do about it, coming up next on Security Now!.

**Leo Laporte:** This is Security Now! with Steve Gibson, Episode 988, recorded Tuesday, August 20th, 2024: National Public Data.

It's time for Security Now!, the show we cover the latest security news. Oh, there's a few little tiny stories to talk about with this guy.

**Steve Gibson:** Happy birthday to us. Happy birthday to us.

**Leo:** It's our birthday.

**Steve:** Happy birthday, happy birthday, happy 19th birthday to us.

**Leo:** That's Singing Steve Gibson, everybody, the security cowboy.

**Steve:** Yes, sir, it was August 19th, 2005. We were all security virgins, and we stepped into this not having any idea what we had to scrape off our shoes.

**Leo:** You weren't, yeah, you weren't a security virgin. But we were all podcast virgins because it was the very earliest days of podcasting.

**Steve:** Well, let me just say I'm way less virginized than I was back then.

**Leo:** You know a lot more. Whew.

**Steve:** Oh, goodness. Yup.

**Leo:** Yes, yes. And now, of course...

**Steve:** Beginning to believe my own PR on this because, wow. After today, oh, goodness. When our listeners go and check for their own personal data...

**Leo:** Oh, I already did this. It's depressing.

**Steve:** Oh. You're there, Leo, as I'm sure you know. I'm there. My wife is there. Everybody I've looked up is like, whoa, doggie.

**Leo:** Yeah, this is a bad one.

**Steve:** So, yeah.

**Leo:** So what are we talking about, Steve? This is the subject; right?

**Steve:** We're talking about today's podcast of course had to be titled National Public Data, which the press, you know, in their typical hyperventilation, thought that it was three billion people. And I thought, wait a minute. What is the current carrying capacity of the Earth? Because it is...

**Leo:** Well, it's more than three billion. But still, that seems a lot.

**Steve:** Yeah. That says that people who've never held a phone or gone to the Internet somehow have their Social Security Numbers up - they don't even have Social Security Numbers - up on the Internet. No. It's three billion, actually just shy of three billion, 2.9 billion records.

**Leo:** Wow.

**Steve:** Which pretty much tells an individual's history over the last several decades. Anyway, we're going to get to that and have fun with that. We've got a bunch of stuff. We've got, of course, because yesterday was our birthday, our 19th birthday, we are now, with Podcast 988, and fortunately no end in site...

**Leo:** Thank you.

**Steve:** In another 11 we'll make 999.

**Leo:** Yay.

**Steve:** And we're just going to seamlessly cross over into four digits. I will have to do a little tweaking of my software, as I mentioned several years ago. But we're not going to stop. So Year 20, here we come. We're going to - I have some interesting update on the topic that's sort of been a running theme for a while, and it's probably going to be one next week, but in a different sense, which is this challenge of certificate revocation, which is important, which is why it's gotten so much of our time and so much of the industry's time and attention. Something happened, Leo, at 2:00 a.m. this morning which we're going to be talking about and explaining, and also what it means. And also there's one last piece that I haven't talked about which we're going to nail down. So there's that.

Also we've got the six zero-days which were patched last week, on Tuesday, during Microsoft's Patch Tuesday. I also ran across, thanks to a listener, and I saw you mentioning it at the end of MacBreak Weekly, the 53 discovered episodes of the 1980s Famous Computer Caf radio show.

**Leo:** Were you ever on that show?

**Steve:** I wasn't. But we get to hear Bill Gates before his voice changed. So that's going to be fun. We also have the third release, third and final I should say, of IsBootSecure; a quick note on GRC's email, how that's going, which is to say really well; and some listener feedback. Then, again, to no one's surprise, because I imagine our audience has already heard about this, but we're going to take a deep dive into the background, meaning, and impact of the largest personal data breach in history, and how everybody how they can find themselves and all their personal information online. We're going to talk about what to do about that and what it means for the future. And as if that wasn't enough, oh, Leo, we have a Picture of the Week. This has got to be, on the geek rating scale, it's off the scale probably. Most people would look at this and go, what are these...

**Leo:** What are they smoking?

**Steve:** ...nerds talking about? Why are they - have they fallen off their chair in laughter? Because it's very much on the inside. But I had someone already respond, say this is definitely among the top five you've ever done, so...

**Leo:** Wow. Oh, I haven't seen it yet.

**Steve:** It's great.

**Leo:** We'll see it together for the first time.

**Steve:** It's oh, so clever. I just - whoever this genius was who did this, it's like, hat's off.

**Leo:** Very nice. All of this still to come. In just a minute we'll get the Picture of the Week on Security Now! 988, as we gradually head toward 999 and beyond.

**Steve:** We are drifting under power.

**Leo:** I love it. I love it. Okay, Steve. Let's get the - I'm pulling up the Picture of the Week even as we speak.

**Steve:** Just so good.

**Leo:** Do you want to describe it a little bit?

**Steve:** Oh, yes.

**Leo:** Go ahead, and I'll pull it up.

**Steve:** Okay.

**Leo:** I don't want to break the, you know, the surprise. Should I look at it first?

**Steve:** Yeah, you should look at it because it's just so wonderful.

**Leo:** I will look at it, and then you can describe it. Okay. So this is me. I haven't looked at it yet. This is how we're going to look at it together. I'm going to scroll it up now. "If water coolers were RAID arrays." Okay. Now I'm ready to show it.

**Steve:** It's so good.

**Leo:** That's fantastic. That is really great.

**Steve:** It is. So, okay, as you said, I gave this one the caption "If water coolers were RAID arrays."

**Leo:** I apologize. I'm mirroring it. Let me fix that. Go ahead. Keep talking.

**Steve:** Okay. So anyway, looks fine for me.

**Leo:** Yeah, okay. Well, it's just me then, okay, good.

**Steve:** Yeah. We have your standard water cooler, which, you know, is a water cooler with a big jug of water on top. And that's labeled "Standalone." There are seven frames here to represent all the various configurations of mass storage redundancy and management. The second one is labeled a "Cluster" because that's two water coolers, each with its water jug on top. The third one is labeled "Hot Swap," where of course we have one water cooler with its jug on top and, as you often see next to a water cooler, another jug standing by its side, waiting to be swapped in to replace the first jug when it's emptied. So that's the Hot Swap configuration. Now, for RAID 1 we've got - somehow this guy has managed to have...

**Leo:** They were having a lot of fun in the office.

**Steve:** Oh, my god, yes. Two jugs side by side, feeding into one water cooler. So they're next to each other. Kind of off, you know, pushed apart a little bit because they won't actually fit completely vertically. But that's the RAID 1 configuration. RAID 5 is three full-size water jugs similarly arranged. Somehow he's managed to get all three feeding into the top of one water cooler. Thus RAID 5.

Now, of course we have two other RAID configurations. RAID 0 is the one shown on the bottom. That, of course, RAID 0 is concatenating two drives; whereas, as we saw RAID 1 is also known as mirroring, so that's why RAID 1 had the two jugs side by side; while RAID 0 has one jug feeding into the next jug, so they're stacked on top of each other, two jugs feeding down into a single water cooler. And then finally, of course, the RAID 0+1.

**Leo:** Striped, it's the striped array.

**Steve:** Yes, it is, it's the combination of 1 and 0. So we've got two jugs side by side with two jugs above them feeding down into them. Anyway, it's just - one of our listeners sent this to me. I glanced at it, and I thought, oh, this is so good. And actually I had this around the middle of last week so I shared - because I love this picture so much that I shared it with the gang over in the newsgroup. And one of our active followers/contributors over there ran it through some sort of AI generative thing, which significantly cleaned it up. It made the text much more legible than it was and kind of cleaned up the pictures a lot. So I was very appreciative of that.

**Leo:** I love Gumby's suggestion that in this case RAID stands for a Redundant Array of Inexpensive Dasanis. Okay. Thank you very much.

**Steve:** Yeah, very good. Okay. So a contributor over in GRC's newsgroups posted some terrific observations following from last week's discussion of revocation. Andrew wrote: "I have a sneaky suspicion that Steve's long-term plan for revoked.grc.com is going to ferret out what software is swimming the waters in this space, naked, shall we say." Now, he was addressing my plan, which I had shared there, which was to back GRC's

servers away from the use of OCSP, Online Certificate Status Protocol, to see what happens. And at 2:00 a.m. this morning the last received and stapled OCSP status which my revoked.grc.com site had received from DigiCert a week ago, so it had its one-week life, it expired at 2:00 a.m. this morning.

I will share what happened after that in a minute. But let's first examine one last feature of OCSP that we have not talked about yet during this go-round of talking about revocation, and work out OCSP's Achilles heel, which is what made it impractical to require at scale.

Okay. So as we all witnessed in the last two weeks, when GRC's revoked.grc.com site began stapling an OCSP status that loudly stated its certificate had been revoked, and, you know, no one's browser, as we saw, would show the page, which suggests that OCSP appears to be working better than anything else ever has so far. But as we also learned, the CA/Browser Forum now plans to make OCSP support optional and to switch back to requiring the use or requiring Certificate Authorities to publish Certificate Revocation Lists, thus making them mandatory.

So at the time, last week, I replied to Andrew. I wrote: "If the industry has inexplicably chosen to abandon the system that all browsers are currently using with 100% success - as was just demonstrated with GRC's OCSP stapling working perfectly everywhere - then, okay, we'll start testing the replacement system, that is, Certificate Revocation Lists, to see how well it does."

Andrew added: "With browser providers like Google, Mozilla, and Apple providing the CRL important bits as a centralized service with rapid-update to their browsers in the field," he said, "It sounds like insanity."

Okay. To which I replied: "I 100% agree. If Let's Encrypt and the rest of the CA/Browser Forum are suddenly so worried about web privacy due to individual browsers reaching out to query Certificate Authority OCSP services, then we'll start a countdown on the mandatory support for CRLs."

Okay. So the final piece that I've not discussed at all - I may have mentioned it, but I'm not sure, though we did cover it in depth back at the time 10 years ago when we first talked about all this, is known as "OCSP Must Staple." So here's the problem that solves. A web server obtains a web server TLS certificate signed by its Certificate Authority. Right? That's what they all do. And the whole point of signing is that not a single byte of that certificate can be changed, or its signature will become invalid.

This means that when that web server wishes to include that Certificate Authority's recently received OCSP assurance about the certificate's validity, or its lack thereof, whatever the case may be, the Certificate Authority's signed OCSP status can only be appended to the certificate. It cannot in any way modify or be incorporated into the certificate, right, because that would break the certificate's signature. So that's why the term "stapling" has been adopted, since stapling is such a good analogy.

So the problem is, stapling an up-to-date OCSP status to a web server's certificate is optional. If a bad guy gets hold of a valid web server certificate, they'll gladly staple any OCSP good news to that certificate every time they send it out to a web browser; right? Since, as we know, that prevents browsers from looking any further. If they've got a valid stapled good news OCSP certificate, or even as we saw last week with GRC sending out a bad news OCSP certificate, they don't look any further.

But once the certificate has been revoked by its Certificate Authority, and its Certificate Authority's OCSP now contains bad news, as I said, as GRC's revoked site certificate did last week), no bad guy would continue stapling that bad OCSP revocation news to their

fraudulently obtained TLS web server certificate. Instead, they'll remove any stapling and hope that no one's browser checks the current validity of the certificate by querying the Certificate Authority's OCSP service or their CRL, their Certificate Revocation List, directly themselves. And so far as we know, browsers no longer make their own queries to OCSP on their own. Actually, there's one exception standing out, but we'll get to that.

Okay. So to prevent a stolen certificate from not having an OCSP response stapled to it, it's possible for a website that wants the best security for itself, and which always intends to staple, and can commit to stapling, to ask its Certificate Authority to include a flag built into its TLS certificate, which is known as "OCSP Must Staple." Since that "Must Staple" flag becomes an integral part of the signed certificate itself, it is immutable and, once issued, can never be changed. And the presence of that flag in any certificate that's received by a browser is an assertion that this certificate must only be honored and trusted if and when it's accompanied by a current and still valid OCSP assertion to that effect, talking about an updated status for the certificate. An OCSP statement must be stapled to the certificate, or it must not be trusted.

So that solves the problem of the bad guys choosing not to staple any bad news to their stolen certificate, and just assuming, as they can, certainly for Chrome and most other browsers, that the browser won't go out of its way to check on its own. With OCSP Must Staple, the browsers do not need to make a second query because they're being told if there isn't a stapled assertion, a recent, you know, currently unexpired assertion, that they cannot trust that certificate, specifically to deal with that case.

In my reply to GRC's newsgroup I wrote: "Let's Encrypt, and all other CAs following the CA/Browser Forum, can mandate that they will be setting the 'OCSP Must Staple' requirement in every certificate they issue after some date certain. And that will force all web servers to support stapling and to staple. So what's the problem with that?"

With enforceable stapling - which we haven't had until we bring the OCSP Must Staple into it. With enforceable stapling, after the stapling mandate were to take effect - if it was going to, but we already know this is not the direction the world has gone, but why not - every certificate will have OCSP stapled to it within 397 days once all pre-mandated certificates will have expired. So the entire industry would move there within the maximum lifetime of a web browser certificate, which is now 397 days. And the big win for privacy, which Let's Encrypt is saying they're all worried about all of a sudden, the big win for privacy is that with a fresh OCSP response stapled to every certificate, browsers can and will be inhibited from making their own queries for OCSP status because it's right there on the certificate. So they'll be sped up, and they have no reason to ask any further.

Thus we have fast revocation notification with zero privacy risk, since browsers will get their updates from the server's cert; zero performance overhead for the same reason, no need to ask anyone else or look any further; and reduced load on CAs' OCSP services since only the servers they've issued certificates to will be querying them, not everyone's browser all over the place that are relying on those certificates. And that querying interval can also be readily changed by the CA simply by changing the OCSP's response lifetime. Could be made longer, could be made shorter, whatever, I mean, on an ongoing basis. It's sort of a beautiful system. So why isn't that what's being done?

After not coming up with any answer that I liked, I did some digging in the CA/Browser Forum's documents. In the discussion surrounding that CA/Browser ballot measure SC-63 that we discussed briefly last week, the adoption of which was nearly unanimous, everybody's for it, I found this couple lines of text: "Independent of usage statistics, relying parties cannot consistently depend on OCSP stapling for security unless responses are stapled on all connections." Okay. "Further, even if the web server ecosystem had improved support for OCSP-stapling, and we could require the use of the Must Staple

extension, we'd remain dependent upon robust and highly reliable OCSP services, which have been an ongoing ecosystem challenge."

So what they're saying is that they recognize that the use of the Must Staple extension, which does solve this problem, and the privacy problem, and the load problem, also it creates a dependency, I mean, a serious dependency upon OCSP services being available. And they're saying so far they've not been highly reliable. I actually found a piece of intelligence from a project that Mozilla did showing that their browser gets 7% failures on OCSP queries. So that's not good.

Okay. Anyway, so that set of lines in the ballot measure gave me a clue. So as I said, on the one hand, what they appear to be saying is, all experience to the contrary, unfortunately, when we looked at this 10 years ago, things sort of were like this. And it doesn't seem that that much has changed in the intervening 10 years since we last looked at this. They're suggesting that OCSP Must Staple cannot be used because the robustness of OCSP services has never been sufficient, and still isn't today.

So thinking about this, one thing that the plain vanilla certificate system offers, the one, you know, just in the old days, Certificate Authority, web server, browser, that original plain vanilla certificate system is no requirement for real-time communication with anyone other than between the client and server, which after all is connecting to the server, wants to make a connection. The connection works. Part of the handshake is the certificate exchange which the browser verifies. So there's, like, zero overhead in that, and no other requirement for any other real-time communication. The server has a signed certificate. The client locates the server by its domain name using hopefully secure enough DNS. And during the client's connection to the server, the server provides its certificate to prove its identity at that domain name. It's an elegant system, and it is minimal.

Of course, the one place this beautifully minimal system falls down completely is when the certificate it's sending, which was signed, and it's got lots of life left in it, can no longer be trusted. There's no way for the browser to know that within this minimal system that we originally had. In the absence of any other facility, that certificate will be trusted until it expires, which in the case of Let's Encrypt will be a maximum of 90 days, or a maximum of 397 days for traditional now annual web browser TLS certificates. If our goal is for this otherwise simple and elegantly minimal system to deal with the need to revoke certificate trust before the certificate's natural end of life, we're going to need to add something else. And what we see is that the industry has been struggling from the beginning to come up with a solution that works well for everyone.

None of GRC's servers have had any problem with OCSP stapling. And I frankly doubt that anyone's would. And remember that the server starts looking a day in advance to update its stapled assertion. So a brief outage would not be a problem. Whereas Mozilla is probably referring to brief outages in their browser trying to do OCSP lookup on the fly, which, you know, it might be off now, but on again in five minutes. Who knows? If the CA was restarting their OCSP service at the moment, who knows?

Okay. So if all of the sites which are using stapling now had Must Staple in their certs, but if they were, for some reason, if all of those servers were for some reason unable to obtain an update from their Certificate Authority during that last day of the OCSP assertions lifetime, they would effectively go offline because with OCSP Must Staple in their certificate, no web browser would trust their expired stapling. Every affected website would look like GRC's deliberately revoked site looked after our podcast two weeks ago. That's not good.

Browsers have given up their own checking of OCSP for performance reasons, and they've been absolved of any guilt by stating their concern over the privacy of their

users, although that doesn't seem like such a big problem, but okay. So they rely upon stapling to do the work for them when stapling is present, and they do nothing when it's not present. But that means that any certificate that does not use Must Staple will always be vulnerable to long-term abuse if it's stolen, as I said, since no illegitimate server would include a negative OCSP response, which would cause all current web browsers to default to, if a negative response was there, it wouldn't work. But not including a stapling would cause all current web browsers to default to trusting the malicious website.

And the industry cannot improve certificate revocation and user privacy by moving to Must Staple because a DDoS of the apparently still not very robust OCSP service, which would have then been a requirement for all a Certificate Authority's customers, would result in a widespread web server outage. All if would take, say for example that some Certificate Authority, well, that the CA/Browser Forum said we're moving the entire industry to Must Staple, and everybody has that in their certificates, all the old certificates that, you know, a year goes by. All the old certificates that didn't have the Must Staple bit got renewed, and now they have the Must Staple bit. Stapling is now the thing we do. Revocation is wonderful.

All it would take, and it doesn't take much imagination at this point to imagine that, you know, it would happen, baby, is a one-day DDoS on some CA's OCSP service. All of the staplings that were trying to refresh themselves would not be able to get updated. Their attestations of the certificates still being trusted would expire. But because Must Staple was in the certificate, no web browsers would trust them. And there would be growing, as the days went by, but it would start after a day, growing mass outage of all the servers that were trusting, that were using that given under-attack CA.

So it's clear that, by adding an online facet to this, no matter what we do, we have a problem. I believe that this explains why we're not seeing OCSP going any further. You know, unless stapling is mandatory, it can be bypassed simply by not including a stapled certificate. And the danger of making stapling mandatory is that an OCSP outage, for whatever reason, malicious or accidental, lasting more than a day, would have devastating consequences as all of that Certificate Authority's certificates would over time become untrusted.

So where are we left? Certificate Revocation Lists, imperfect as they are, are less "online" than OCSP which, after all, OCSP stands for Online Certificate Status Protocol. The bottom line is that in the reality of today's Internet, "online" is not something that can be made to work. Its strength is also its failure.

Okay. So as I mentioned last week, I think I just sort of mentioned it in passing that I was thinking about maybe what would happen if I deliberately blocked all GRC access to DigiCert's always online for me OCSP Service. I did that. Before I did that, I double-checked that my certs did not for some reason have "OCSP Must Staple" enabled, or I would have, you know, put myself, I would have created the same outage I was just talking about.

The last-received OCSP status from DigiCert for the revoked.grc.com site was set to expire around 2:00 a.m. this morning. I did it a week ago. And sure enough, when I checked this morning, GRC's revoked.grc.com server was no longer stapling that expired OCSP status to its thoroughly revoked - uh-huh, there it is, Leo - to its thoroughly revoked TLS certificate. And what do you think happened? Yep.

**Leo:** Oh, boy.

**Steve:** Every web browser other than Firefox has resumed showing the revoked.grc.com website. Chrome loves it, Safari loves it, Edge proudly shows its page. Everyone's happy with the site despite the fact that its certificate was revoked 21 days ago. Every web browser except Firefox is once again completely happy with the site.

**Leo:** Well, good on Firefox. Why did Firefox not...

**Steve:** Because, if you go, Leo, in Firefox to - you can do about:preferences#privacy, or open Firefox and go to Settings, and then Privacy and Security on the left, under the main topics, then scroll about two thirds of the way down. And you will find a checkbox which to their endless credit is enabled by default. It says: "Query OCSP responder servers to confirm the current validity of certificates."

**Leo:** And there it is. It's revoked.

**Steve:** Yup. Firefox is revoked. And if you go into Settings, Privacy and Security, scroll two thirds of the way down, you'll see OCSP querying is enabled. To verify, since it was on mine, I installed Firefox in a virgin Win10 VM this morning. The first time it was installed. Sure enough, it was enabled by default. And there it is. Query OCSP response servers. If you turn that off and refresh the revoked page, comes right up.

**Leo:** You're right. So let's keep that on.

**Steve:** Yes.

**Leo:** And what's funny is, despite all the concerns, I didn't notice anything slow with Firefox. It works fine; right?

**Steve:** Yeah. Yeah. And we've always had it on. Now, I don't know what percentage of servers were stapling. Mine always was. Mine isn't now. Neither GRC.com nor www nor revoked are stapling because I'm preventing those servers from obtaining, unfortunately, in my version of IIS you're unable to disable that through configuration. So I had to - I actually used my hosts file. I just blackholed ocsp.digicert.com, and my servers were no longer able to obtain that status. I just set it to 127.0.0.1, you know, the localhost IP.

So the reason for all this, we've now proven, is that the revoked.grc.com server is not stapling its negative OCSP response, you know, which said, "Oh, by the way, this site's certificate, the one I've just been stapling, you know, has been revoked. It's not saying anything now." And so in the absence of either a positive or negative OCSP status, all browsers other than Firefox trust the revoked, but otherwise valid, certificate.

In other words, here we are, 21 days after that certificate's revocation, presumably revoked - it may have been revoked for administrative reasons or because somebody stole it. We know that's not the case here, but I'm simulating that. Somebody, a bad actor, could have stolen any other site's certificate, and the site could have known about it, immediately revoked it with its Certificate Authority. Chrome could care less. Chrome doesn't know. And we've had three weeks of this right now.

**Leo:** That's not an accident. That's Google assertion as they have always asserted that the whole system's broken. So they're just going to ignore it.

**Steve:** Well, we know that for EV certs that they do some special looking. And even Apple with iOS and presumably macOS, EV certs they care about. So they maintain some sort of a certificate revocation list. GRC's certs 10 years ago were EV, so I was seeing that treatment. But like a lot of the rest of the world, when all the browsers stopped giving you any special treatment, you're just throwing your money away to have an EV cert. So I switched back to domain validation certs which, you know, is where the web is going.

**Leo:** Right.

**Steve:** So anyway, Firefox stands alone in correctly refusing to show the pages being served under the guise of this very well-revoked TLS certificate. We'll see what happens. I don't think this is going to change because it's not an EV cert. Neither Chrome nor Apple are giving this any special attention. And the entire revocation system, we are now seeing it function the way it really does, which is only if you ask the authority's OCSP server directly do you get an answer. And Firefox is the only browser that does that.

**Leo:** Wow. Truly amazing. You want to take a break?

**Steve:** Yup, thanks.

**Leo:** Because Patch Tuesday there's a lot to talk about.

**Steve:** Ninety security vulnerabilities.

**Leo:** Oh, man.

**Steve:** Yeah.

**Leo:** Wow.

**Steve:** But Leo, these are the last 90.

**Leo:** Oh.

**Steve:** They found them all.

**Leo:** Oh, good. That's it. This is it.

**Steve:** Yes.

**Leo:** No more Patch Tuesday.

**Steve:** Canceling September.

**Leo:** It's all fixed.

**Steve:** Nothing to fix.

**Leo:** Nice.

**Steve:** It's all working perfectly.

**Leo:** It's about time.

**Steve:** Working perfectly. It won't boot, but it's working perfectly.

**Leo:** You know, that's the secret. Just don't boot it. I'm still waiting for my super-duper Snapdragon Windows Copilot developers machine. I thought it was going to come last week. But I'll ask Richard about it tomorrow. But I guess I'll be running Windows 11 in here, too, so that'll be fun. That'll be interesting. I'll be able to Patch Tuesday along with the rest of you. We now stream live to everywhere, including Steve's favorite platform, X.com. Kidding. Kidding.

But we do have hundreds of viewers every time we do this on X, so we welcome you. YouTube.com/twit/live. Twitch.tv/twit. Facebook, LinkedIn, X.com, of course our Club TWiT Discord, and Kick. Seven different ways to watch us live, every Tuesday, right after MacBreak Weekly, about 2:00 to 5:00 p.m., Pacific 5:00 to 8:00 Eastern time, 21:00 UTC. Make sure you watch live. But even if you do, subscribe, because you're going to want that library - right, Steve? - of great Security Now! episodes, 19 years' worth. Download them all. Collect all 988.

**Steve:** Yeah, and, you know, we don't mention often enough, and our newer listeners probably don't know, that back then we were sort of still in the early knowledge dump phase of the podcast. I did a series, several series. One of like five or six was how the Internet works. And then there was the other one was how CPUs, like, you know, how computing technology works. And we've had many people who've, like, created like box sets of those, I mean, like...

**Leo:** Steve's premise was you're going to need the foundational knowledge before you can understand what we're going to be talking about. So, yeah, I mean, that was, I don't know, in the first hundred episodes, I think.

**Steve:** Yeah.

**Leo:** But it's all there at TWiT.tv/sn. You can go back, back, back, back, all the way.

**Steve:** Got to dig. Got to dig back a little ways.

**Leo:** People have written scripts to scrape it and all that stuff. We don't make it that easy, I understand. But I don't know, if I can find those scripts, I'll dig them up. Otherwise you can just download.

**Steve:** And then of course we have the ever-favorite Portable Dog Killer episode.

**Leo:** Yes. Which was our, weirdly enough, our Christmas episode for many years.

**Steve:** No animals were hurt during the production.

**Leo:** There's a lot of history in this show. I told you that Burke brought in the modern-day version of this. It looks like a bird house. You're supposed to hang it on your fence. If you've got a neighbor's dog that barks, it senses the barks and sends out a tone that only dogs can hear, just like your Portable Dog Killer. But it's disguised so your neighbors don't know why your dog suddenly goes agh. What was that? All right. On we go. Let's get this...

**Steve:** It's like putting it like a secret bark collar on your neighbor's dog. Yeah.

**Leo:** It's kind of like that. That's exactly what it is. When they bark, they go, ooh, I don't like that. All right.

**Steve:** Okay. So Patch Tuesday. Last Tuesday was August's Patch Tuesday for Microsoft. And just so that everyone knows, I was not serious about all the bugs being found because that doesn't seem to be a problem that Microsoft has of worrying, like, are we going to hold Patch Tuesday this week or not?

**Leo:** This will go on for years.

**Steve:** Or this month. No. They're having it. Okay. So it's become sort of standard for the second Tuesday of the month for Microsoft and for many other publishers who also appear unable to ever get the important bugs out of their code. In this month's installment of trying some more, Microsoft released updates to fix at least 90, nine zero, security vulnerabilities in Windows and their other software, which included a startling six zero-day flaws that were, or maybe are still, being actively exploited by attackers. Flaws were found and fixed in Office, .NET, Visual Studio, Azure, Copilot, Microsoft Dynamics, Teams, Secure Boot, and of course Windows.

Among the six zero-days fixed this month, half of them, thus three, are local privilege escalation vulnerabilities which, you know, while when we talk about them they are severe, inasmuch as they really enable an existing attack to be made much worse by

giving somebody who's already managed to get inside a machine the system-level root privileges that they need in order to do, you know, to get up to much more mischief. So we know little about these, although you can bet that would-be attackers are hard at work reverse engineering the changes that Microsoft shipped in order to figure out what was going on before and put them to nefarious use for systems that haven't yet been patched, you know, when they're already able to get in.

There are, however, some worse problems. A remote code execution vulnerability when Microsoft's Edge browser is operating in Internet Explorer Mode. Although IE mode is not enabled by default in Edge, thank goodness, the fact that this is or was being actively exploited, like today, suggests that there are occasions where an attacker can arrange to either enable it somehow or has identified a user or an organization who has enabled this, probably because they've got some very backward compatibility need, back to the last version of IE.

So, you know, it makes - turning that on and using it makes Edge look like IE. And I'm trying to think why I did that the other day. There was something I was doing forensically. It might have been back when I was messing with the cookie system again. But I had some need to do something with IE mode in Edge. And, you know, they don't make it easy. But there are people who do need it. And so the problem is that if you're using it, there's a remote code execution vulnerability, believe it or not. So it's good that it's not on by default.

Another zero-day is a bypass in their "Mark of the Web," which we've talked about extensively before, security feature, which causes Windows to be far more mistrustful of any files obtained from the Internet. As we've seen in the past, however, this MOTW bypass is always used as a part of a larger exploit chain, but its bypass does enable something to be done that was supposed to be impossible. And it's not in the user's best interest, whatever that was.

This month's third and final zero-day is a remote code execution flaw in Microsoft Project. Microsoft and several security firms have pointed out correctly that this vulnerability is only useful against users who had previously disabled notifications about security risks of running VBA Macros in Microsoft Project. So, you know, Project has been a source of lots of problems. So Microsoft just puts up a note and says, are you sure you want to run a macro? By the way, did you know one is running, or wants to? And many people go, what? Who's doing that?

So those are the six zero-day flaws out of the total kettle of 90. And they were under active use a week ago when Microsoft patched them. Hopefully, you know, again, we know that patching in a timely manner is something that has bitten enterprises in the past. So some enterprises are reluctant to do so because they don't want to have, like, apps that they depend upon stop working. This is an instance where, depending upon your profile, you may want to get caught up with updates. Sounds like it would be a good idea.

And Leo, I mentioned that you caught my attention at the end of MacBreak Weekly, which I was listening to as we cross over into this show. Thanks to a listener of ours, Larry Deniston, who brought this to my attention, I have some news that might be of interest to both our old-timer listeners and our younger audience, who may have heard the names of these people who've in many cases grown to become somewhat legendary within the PC industry that we all love. Audiotapes of a mid-'80s, 1980s technology radio show, which was known as "The Famous Computer Caf," were found earlier this year by an archivist who restored and digitized them.

Yesterday, Monday, the Internet Archive posted. They wrote: "A previously lost cache of celebrity and historical interviews from a long-dormant radio show have been discovered,

digitized, and made available for all. The Internet Archive is now home to 53 episodes of The Famous Computer Caf, a 1980s radio show about the new world" - new at the time - "of home computers. The program included computer industry news, product reviews, and interviews, and aired from 1983 through 1986 on radio stations in Southern and Central California.

"The creators of The Famous Computer Caf saved every episode on reel-to-reel tapes, but over the years the tapes were forgotten, and ultimately lost. Earlier this year, archivist Kay Savetz recovered several of the tapes in a property sale and, recognizing their value and worthiness of professional transfer, launched a GoFundMe to have them digitized and made them available at Internet Archive with the permission of the show's creators.

"While full of time-capsule descriptions of 1980s technology news, the most exciting aspect of the show has been the variety and uniqueness of the interviews," they wrote. "The list of people that the show interviewed is a who's-who of tech luminaries of the 1980s - computer people, musicians, publishers, philosophers, journalists. Interviews in the recovered recordings include Timothy Leary, Douglas Adams" - of course "Hitchhiker's Guide to the Galaxy" fame - Bill Gates, Atari's Jack Tramiel, Apple's Bill Atkinson, and dozens of others. The recovered shows span November 17, 1984 through July 12, 1985.

**Leo:** Isn't that cool. So cool.

**Steve:** So for ease of access, I've made this GRC's shortcut of the week for this episode, which this episode is 988. So if you go to grc.sc/988, that will jump your browser to the Internet Archive's blog posting, which includes a link to a Google Docs spreadsheet listing all 53 recordings, who's on them, and then their direct links to their Archive page at the Internet Archive. So very cool.

**Leo:** Really neat stuff.

**Steve:** Yeah.

**Leo:** You know, I wish I had archives of all of the shows we did, Dvorak and I and so forth. It's great that they saved so many of them. And it's great that they were saved. The problem with those reel-to-reels is they're going to die in a few years.

**Steve:** Yeah, you get cross-wrap imprinting and, you know, heat and...

**Leo:** They flake, yeah.

**Steve:** ...humidity and, yeah. And in many cases the magnetic coating flakes off of the plastic backing. Or Mylar, rather.

**Leo:** So kudos to Kay Savetz and to the Internet Archive for preserving and distributing these. That's really great, yeah.

**Steve:** A quick follow-up on GRC's newest IsBootSecure freeware. Last Wednesday I posted Release 2, which added keyboard accelerators and tooltips to the buttons, and I fixed a noncritical misreporting edge case. Then Release 3 later the same day, last Wednesday. It added a "silent" option which causes IsBootSecure to fully suppress its user interface. So it runs silently on any Windows machine. It examines the machine and immediately exits with an exit code that Batch or PowerShell scripting is able to capture and check. This allows the app to be deployed by scripts within an organization, organization-wide, to check and inventory an entire enterprise's inventory of PCs to ascertain their boot-time status. And not just do they have a mistrusted platform key or not, but you can quickly see whether those machines are booting with Secure Boot enabled or not. So it returns a status zero through seven, which encompasses all the various things that it might find.

And with that finished, I got back to work on SpinRite 6.1's documentation. I've finished all of the static content, and I'm now down to producing the video walkthroughs which I'm looking forward to creating so that non-owners will have a sense for actually, like, watching SpinRite run while it works. So that's going to be fun.

And I'm also very pleased to report that GRC's email system is working very well. I sent out 8,443 announcements about this podcast a couple hours ago. Only five were bounced as undeliverable for some reason. So, you know, five out of 8,443...

**Leo:** Pretty darn good. Yeah.

**Steve:** ...I'm delighted with. I did also want to mention that, and I'm sure some people have found out for themselves, that I finally started last week bouncing any email, any incoming email which had not been registered with GRC beforehand. Anything that is sent to securitynow@grc.com, rather than being redirected into a separate folder where I'm able to monitor it as I had been, is now bouncing back. For the first couple months I just wanted to help people who were sending things through. So I would collect a bunch and then just send them back a notice explaining that, well, this would have gone into our, you know, like bounced into our spam and never be seen or sent back. So please figure out, like, why.

Mostly people registered one account and then sent from another. Like I could see their account name registered with Proton Mail, but they were sending from iCloud.com. So it's like, okay, you need to send me from the one that you registered, or that's not going to work.

And I also did get some complaints from people who look around for the email address to send to, and they are pissed off when they can't find it anywhere. I understand the annoyance, but I want to keep this more or less just between us. You know, I'm not advertising the securitynow@grc.com account anywhere on GRC.com's site. It's just for, you know, us insiders. So securitynow@grc.com, and you're not going to find it written down anywhere.

Michael French said: "Hi, Steve. I'm stumped on how to get information through WiFi firewalls that block everything except HTTP on port 80 and HTTPS on port 443. I run an OpenVPN server at home in Alabama on TCP port 443." So, right, just looks like a web server. He said: "But some firewalls outside of my home still block my communicating with it after only a few seconds of operation, presumably from their implementing deep packet inspection." He said: "I just returned from Europe and found that all the countries I visited (UK, the Netherlands, Belgium, and France) have rigged their WiFi firewalls to block everything except HTTP and HTTPS. I would sure appreciate your suggestions on how to get communications through these over-restrictive WiFi firewalls. Thanks, Mike."

Okay, so this was an interesting puzzle. It cannot be deep packet inspection unless Michael had been installing and trusting a certificate from the various access points he was using, which I'm sure was not the case. Any TLS connection being made to port 443 will be 100% opaque to anyone monitoring the packets. They'll see what looks like a normal connection to a remote web server with a back-and-forth handshake, but only from the outside. They will have no way of knowing what's going on inside.

My best guess about what might be going on comes from Michael's comment that he's getting disconnected after only a few seconds of operation. So first of all, I should say I'm wondering why it's consistent, why his experience is so consistent in the UK, the Netherlands, Belgium, and France. I mean, that's a little suspicious to me that he's seeing, like, the same thing from so many different places. It makes me think it's something more about what's happening at his end.

But although persistent HTTP connections between a web browser and web server can be made and sustained, even then they're usually dropped after all pending queries and replies have been exchanged. It's unusual for a passive and unused connection to be maintained. So it might very well be that these WiFi access points are watching the flow of traffic and are deliberately dropping any connections that go idle. Doing that would not interfere with normal web browser use, while it would be an effective way of blocking other "web-like" traffic such as an HTTPS VPN. So that's my thought, Mike, about what might be going on.

The problem is the VPN doesn't, even from the outside, the flow of packet traffic does not look like an actual browser web server interchange, which is generally a burst of stuff and then the connection goes away. You know, you're having probably a brief burst, and then you basically have a remote network connection, and something might be looking at the packets going, even without seeing into them, just the size and, you know, the timing of them, and decide this doesn't look like a web server and a web browser. We're going to bail.

Doug White said: "Aloha, Steve. Listened to the Tuesday podcast and thought I'd mention something when it came to transferring DNS. I switched over to Hover from GoDaddy, something I've put off for years, and wanted to mention that the DNS server entries after the transfer to Hover still pointed at the GoDaddy DNS servers. I had to look to find the Hover DNS server names and replace the GoDaddy entries in the Hover settings. I'm guessing it's so that everything still works after the switchover, but I wasn't alerted to the fact, that I'm aware of, that I needed to make that change. Cheers."

And I'm sure that Doug's correct that in switching his domain registrar from GoDaddy to Hover, Hover would have examined his previous registrar's domain nameserver entries and would have deliberately left them, copying them into his relocated registration at Hover. Eventually, presumably, GoDaddy could be expected to suspend their support for his DNS once his registration had been relocated, so this would have probably eventually come to light, like stuff would have stopped working for him. So after moving to a new domain registrar, the point he's making, and it's a good one, which is why I wanted to share it, you'll want to make sure that its registered nameserver records are pointing where you intend to have them be pointing, which is probably to the DNS server offered by that registrar.

Scott asked: "Is there an advantage of a seven-day stapled OCSP attestation over a TLS certificate with a seven-day expiration? With certificate automation, there's no reason an expiration needs to be 30 days or a year or a week. If seven days is enough time to catch a revocation, just expire the cert that quickly. Revocation," he says, "only really seems like it makes sense if it's instant."

So what I believe we've clearly seen now is just how much all of this revocation and certificate life business involves a tradeoff. Let's Encrypt has been automated from day one, but they chose 90 days for their certificates, when they could have chosen seven days. Why? One advantage to 90 days, especially when they were starting out in the beginning, was that it would significantly reduce the load on their certificate issuance and delivery infrastructure by a factor of nearly 13, from between nine and 70 days. But it's true that so long as a network outage or attack would not hold Let's Encrypt off the air while their certificates were expiring and were unable then to renew, then their certificate recycle time could be as short as they like.

But I know that if my servers were using Let's Encrypt certs, instead of DigiCert's 397-day life certificates, I would want as much life per certificate as I could get, you know, just for the safety margin that that provides. Because one thing that has happened to our industry is that you're no longer very effective on the web if you don't have a valid TLS certificate. You know, browsers actually, depending upon which one, you can normally force them past an expired cert.

You know, I'm sure that those of us who surf to less-often-visited sites will sometime encounter a site that says, oh, this can't be trusted, the certificate expired. If you look at it, it expired yesterday. So it's like, okay, you know, the guy who's in charge of that is on vacation; or, you know, it's an unimportant server that hasn't come to the person's attention yet. So it's like, okay, fine. And then you push past all these warnings and bewares and cautions, and then you get to the site anyway. But still, as I said, you're going to see your traffic fall off for sure if your site starts serving an expired certificate.

And in a final different twist on expiration, Brandon Foust sent a very short note. His email just said: "Will it boot on 6/2/2031?" And he sent me a picture of his Dell something computer's UEFI platform key where it shows it's not one of the bad ones, it's issued to Dell Inc. Platform Key, issued by Dell Inc. Platform Key, and it says "Valid from 6/1/2016 to 6/1/2031." So he says, what about the next day, 6/2/2031? So he is wondering what happens on June 2nd seven years from now. And the answer is that in this instance the date does not matter. It would be up to someone deciding whether or not to trust the root certificate based upon its date, and it's the UEFI firmware that's using this platform certificate to check the signatures of the other signatures. So it's the root, and it has signed other certificates in its database.

So distrusting the platform key root certificate would require a deliberate act within the UEFI firmware, which is not something that it cares about or would do. So the only reason we have a "not valid before" and "not valid after" date is because certificates have to have them in order to be valid certificates. So they just put something in there. As it is, it's a 15-year certificate, so they weren't in any hurry to expire it.

Okay. And Leo, let's take a break, and we're going to talk about, oh, boy, the clearly most worrisome data breach that we have had yet.

**Leo:** All right, Steve. We're going to get to the big, big story that everybody's been talking about, the National Public Data Breach.

**Steve:** For good reason. Wow.

**Leo:** I'm really curious what you have to say about this because there's quite a bit of kind of back-and-forth controversy over exactly what was revealed. There seems to be a lot of errors in the data. Troy Hunt had a long post that made me more confused, frankly.

**Steve:** That's what we're going to share, actually. I've edited Troy's post in order to bring his information to us.

**Leo:** And obviously it's on the Pentester site, which is a great way to find your data and my data and everybody else's data. Yeah, soon as I entered my name, and it didn't need much. You know my birthday. That's all you needed. You found my father's information, my addresses, it was - this is everything that these creeps at National Public Data were selling.

**Steve:** Yup.

**Leo:** Now it's free. Congratulations.

**Steve:** Grc.sc/npd, for those who are listening live, grc.sc/npd, National Public Data.

**Leo:** And we will get to that in just a bit. But first, let's talk about our sponsor for this segment on Security Now!. And actually our next sponsor is very appropriate, and you'll want to stay tuned for that one, too.

All right, Steve. We are really ready for this one. I am, I'm all ears. And as you told us earlier before the show, you emailed me, you said, if you've got DeleteMe on this show today, this would be a good day to have them. We will. We'll follow up with a special offer from DeleteMe.

**Steve:** There was something that surprised me, I think it'll surprise our listeners, too, which it was an observation that Troy Hunt made. Troy, of course, is famous for his Have I Been Pwned, HIBP site.

**Leo:** Right.

**Steve:** And so we'll talk about that. I was very impressed. But before we wrap up today's podcast, and I mentioned this already just before you told us about ThreatLocker, Leo, I'm going to provide everyone listening with a URL for a searchable online database containing the records from this breach, or at least a subset of them, but the ones that make sense. You enter your first and last full legal name, your state of residence - for example, a nickname doesn't work - your state of residence and the year of your birth. And you'll be, and I mean everyone of our, you know, U.S., Canadian, and UK apparently, as we'll see, that seems to be where this list is most focused, will be immediately presented with a list of all the people who share your name, state, and date of birth.

You'll find that the list is sorted by middle name or initial, if you have one, and you'll almost certainly find yourself listed there, often redundantly, many times at different physical addresses, all which will be familiar to you because they will be correct, often with your telephone number, and always with the correct last two digits of your otherwise redacted full Social Security Number.

**Leo:** That's what scared me.

**Steve:** It should.

**Leo:** And they were correct. They were correct.

**Steve:** This is as real as it gets. And for that reason I'm going to start out this week with the main takeaway from what we're going to be describing, which, you know, is probably accurately by those who would know, they're describing it as the largest data breach in history. I would argue that it is clearly the most critical to people data breach we've seen so far. And the takeaway is, we've spoken previously, a number of times, about the need to freeze credit reporting at the three primary credit reporting agencies. If that previous coverage was still insufficient to motivate you, your friends, or your family to take the steps to do so, then if this one doesn't do the trick, I'm pretty sure nothing ever will because this is the pay dirt, unfortunately, for the bad guys.

So naturally this has been a big headline grabber. The Verge's coverage was headlined "The weirdest '3 billion people' data breach ever."

**Leo:** Yeah.

**Steve:** BleepingComputer covered this under their headline "Hackers leak 2.7 billion data records with Social Security Numbers." Brian Krebs' piece was titled: "NationalPublicData.com Hack Exposes a Nation's Data." And if it wasn't too long to be the title of today's podcast, Leo, I so much would have loved to use the start of National Public Data's own admission which read: "There appears to have been a security incident."

**Leo:** Yes.

**Steve:** Gee. Ya think?

**Leo:** Wow. Talk about understatement. Holy cow.

**Steve:** Wow. Yeah. So from across all the coverage, the person who's probably better suited than anyone else to put all of the pieces together, while providing some perspective from his years of involvement with exactly these sorts of incidents, is, as we mentioned, the known to us as "Have I Been Pwned?" Troy Hunt.

And indeed, Troy's coverage is the best I've seen anywhere. He titled his write-up "Inside the 3 Billion People National Public Data Breach." And he wrote, which I've edited for the podcast, he said: "I decided to write this post because there's no concise way to explain the nuances of what's being described as one of the largest data breaches ever. Usually it's easy to articulate a data breach, a service people provide their information to have someone snag it through an act of unauthorized access and publish a discrete corpus of information that can be attributed back to that source." And I should mention, and you'll hear this throughout Troy's posting, he's really focused on attribution. He wants to understand where the data came from.

He writes: "But in the case of National Public Data, we're talking about a data aggregator most people had never heard of, where a 'threat actor' has published various partial sets of data with no clear way to attribute it back to its source. And National Public Data is already the subject of a class action lawsuit, to add yet another variable to the mix." And I'll just interrupt to note that Bloomberg Law reported that this first case is Hofmann v. Jerico Pictures, Inc., which was filed - a suit that was filed in the Southern District of Florida. And they're reporting a couple interesting little tidbits that I wanted to share.

They wrote: "Jerico Pictures Inc., a background-check company doing business as National Public Data, exposed the personal information of nearly three billion individuals in an April data breach, a proposed class action alleges. On April 8th, a cybercriminal group by the name of USDoD posted a database entitled 'National Public Data' on a dark web forum, claiming to have the personal data of 2.9 billion people, according to the complaint filed Thursday in the U.S. District Court for the Southern District of Florida, which said the group put the database up for sale for $3.5 million. If confirmed, the breach could be among the biggest ever, in terms of the number of individuals affected. It's unclear exactly when or how the breach occurred, according to the complaint, and the provider still hasn't provided notice or warning to affected individuals as of the filing.

"The complaint said, to conduct its business, National Public Data scrapes the personally identifying information of billions of individuals from non-public sources, meaning plaintiffs didn't knowingly provide the data to the company. Some of the information exposed includes Social Security Numbers, current and past addresses spanning decades, full names, information about relatives - including some deceased for nearly two decades - and more, according to the complaint. National Public Data did not immediately respond to a request for comment." So, you know, that is, other than their post, which says "There appears to have been a security incident."

**Leo:** We're not sure what the hell happened.

**Steve:** Yeah, you know, we're looking into it. We'll let you know if we think we have something more to share. Anyway, they said, Bloomberg said: "Named plaintiff Christopher Hofmann, a California resident, said he received a notification from his identity-theft protection service provider on July 24th, notifying him that his data was exposed in a breach and leaked on the dark web. He accused National Public Data of negligence, unjust enrichment, breaches of fiduciary duty, and third-party beneficiary contract.

"Hofmann asked the court to require National Public Data to purge the personal information of all the individuals affected" - whoops, well, that's not going to happen, that's their entire life.

**Leo:** It's a little late for that anyway. It got out; right?

**Steve:** Yeah, yeah. "And to encrypt all data collected going forward. In addition to monetary relief, he also asked for a series of requirements, including that National Public Data segment data, conduct database scanning, implement a threat-management program, and appoint a third-party assessor to conduct an evaluation of its cybersecurity frameworks annually for the next 10 years." Okay. So that's the first of likely many similar...

**Leo:** I think there's eight at last count. So more to come, of course. [Crosstalk] on this one.

**Steve:** Okay, yeah. Wow. Of course, I mean, this is obviously going to happen. Everyone's going to jump in. Okay. So let's see what more Troy Hunt has for us. He continues. He said: "I've been collating information related to this incident over the past couple of months, so let me talk about what's known about the incident, what data is circulating, and what remains a bit of a mystery.

"Let's start with the easy bit. Who is National Public Data (NPD)? They're what we refer to as a 'data aggregator,' that is, they provide services based on the large volumes of personal information they hold. The front page of their website says: 'Criminal Records, Background Checks, and more. Our services are currently used by investigators, background check websites, data resellers, mobile apps, applications, and more.'"

He says: "There are many legally operating data aggregators out there, and there are many that end up with their data in Have I Been Pwned, for example, Master Deeds, Exactis, and Adapt, to name a few. In April we," he says, "started seeing news of National Public Data and billions of breached records, with one of the first references coming from the Dark Web Intelligence account, which is @DailyDarkWeb." So he quotes a tweet from April 8th in his posting. The tweet is headlined "USDoD Allegedly Breached National Public Data Database, Selling 2.9 Billion Records."

In the show notes I have a picture of this, and basically which is the standard, we've got something for sale. And in this case, you know, they describe it loosely. They give a sample Social Security Number and a person and a way to verify, a means of looking it up. And they say 200GB compressed, 4TB uncompressed. And it says "includes USA, UK, and CA," you know, Canada. And for the price, this is where they're asking $3.5 million for this treasure trove of data.

Okay. So Troy says: "Back then, the breach was attributed to USDoD, a name to remember as you'll see that throughout this post. And," he says, "this is the first reference to the 2.9 billion number we've subsequently seen flashed all over the press, and it's right there alongside the request for $3.5 million for the data. Clearly," he says, "there is a financial motive involved here, so keep that in mind as we dig further into the story. The image also refers to 200GB of compressed data that expands out to 4TB when uncompressed, but that's not what initially caught my eye," he says.

"Instead, something quite obvious in the embedded image doesn't add up. If this data is 'the entire population of USA, CA and UK,'" he says, "which is around 450 million people in total, what's the 2.9 billion number we keep seeing? Because that doesn't reconcile with reports about 'nearly three billion people' with the Social Security Numbers exposed. Further, Social Security Numbers," he notes, "are a rather American construct, with Canada having SINs (Social Insurance Numbers), and the UK having, well," he says, "NI (National Insurance) numbers are probably the closest equivalent." He says: "This is the constant theme you'll read about in this post, stuff just being a bit off. But hyperbole is often the theme with incidents like this, so let's take the headlines with a grain of salt and see what the data tells us."

He said: "I was first sent data allegedly sourced from NPD in early June. The corpus I received reconciled with what vx-underground reported on around the same time." And he said: "Note their reference to the 8th of April, which also lines up with the previous tweet." He said: "On June 1st, vx-underground tweeted: 'April 8th, 2024, a Threat Actor operating under the moniker USDoD placed a large dataset up for sale on Breached titled "National Public Data.' They claimed it contained 2.9 billion records on United States citizens. They put the data up for sale for $3,500,000.'"

So Troy says: "In their message" - that is, vx-underground's, "they refer to having received data totaling 277.1GB uncompressed, which aligns with the sum total of the two files I'd received," he wrote. "These also mentioned the data contains first and last names, addresses, and Social Security Numbers, all of which appear in the first file among other fields. These first rows also line up precisely with the post Dark Web Intelligence included in the earlier tweet. And in case you're looking at the data and think 'that's the same SSN repeated across multiple rows with different names,' those records are all the same people, just with the names represented in different orders and with different physical addresses, typically all in the same city.

"In other words, multiple rows, in one case six rows all represent one person," he said, "which got me thinking about the ratio of rows to distinct numbers. Being curious, I took 100 million samples and found that only 31% of the rows had unique Social Security Numbers. So extrapolating that out, 2.9 billion would be more like 899 million." He said: "This is something to always be conscious of when you read headline numbers like 2.9 billion. Doesn't necessarily mean 2.9 billion people, it often means rows of data. Speaking of which, those two files contain 1,698,302,004 and 997,379,506 rows respectively..."

**Leo:** Math is hard.

**Steve:** "...for a combined total," yes, "for a combined total of 2.696 billion rows." So that's where the headline numbers come from; right? It's, you know, it's very close. So that's the number of total records in this total dataset. He said: "And in this story there's no question that there is legitimate data in there. From the aforementioned BleepingComputer story: 'Numerous people have confirmed to us,' meaning to the Bleeping Computer guys, 'that it included them and their family members' legitimate information, including those who are deceased. And in vx-underground's tweet, they mentioned: 'It also allowed us to find their parents and nearest siblings.'"

**Leo:** Yeah, my dad's in here. But I have to say all the data is pretty old.

**Steve:** Right.

**Leo:** Yeah.

**Steve:** Right, right.

**Leo:** You can tell it's from an earlier collection.

**Steve:** Unfortunately, the data that matters, like Social Security Numbers, never change.

**Leo:** Yeah, never change. Yeah.

**Steve:** Yeah. So he said, anyway, "We were able to identify someone's parents, deceased relatives, uncles, aunts, and cousins. Additionally, we can confirm this database

also contains information on individuals who are deceased. Some individuals have been deceased for nearly two decades."

And then, in his posting, Troy wrote, he said: "A quick tangential observation in the same tweet: The database DOES NOT [all caps] DOES NOT" - that's not my emphasis. That was in Troy's posting. "The database DOES NOT contain information from individuals who use data opt-out services."

**Leo:** Oh.

**Steve:** "Every person," he wrote, "who used some sort of data opt-out service was not present."

**Leo:** Oh ho ho.

**Steve:** Now, Leo, this is such an obvious lay-up for one of this network's sponsors that now would be a good time to pause.

**Leo:** Should I talk a little bit about our sponsor? Wow. That is...

**Steve:** Isn't that something? I had no idea that it would be that effective.

**Leo:** Let me just go back to npd.pentester.com and look for somebody I know who uses DeleteMe, and just see - oh. Oh. Very interesting. But, see, you never lived in San Diego or Ventura, did you? It's got somebody by your name, but it's not you. Wow.

**Steve:** And if it doesn't have about 10 records for her - because any adult of our age will have left a history behind.

**Leo:** Yeah.

**Steve:** You have a bunch. I have a bunch.

**Leo:** I had a bunch. But Lisa, who's been using DeleteMe for some time, is not in this database. Holy camoly. There is a Lisa Laporte who lives in Ventura and San Diego, I'm sorry, dear, but not my Lisa Laporte. Holy - I don't want to show this other Lisa Laporte's information on the air.

**Steve:** Everybody can see it. They just go to...

**Leo:** They just go look for it, yeah.

**Steve:** Yeah. I mean, that's what's unnerving.

**Leo:** And she is not in there. That is really - I'm glad you pointed this out, Steve. That's kind of amazing.

**Steve:** It is. It's...

**Leo:** It really - so who said that? Oh, I'm showing it now, wait a minute. Let me - I don't know. Turn that off. Thank you. Anyway, that's a remarkable fact, that if you were using - and it doesn't mean just DeleteMe. Presumably any reputable data removal agency you wouldn't be in this breach. Oh, that's something.

**Steve:** Yeah. It's a big deal. That's really...

**Leo:** It's huge.

**Steve:** That's really huge.

**Leo:** Yeah.

**Steve:** Okay. So, but the point Troy was also making about all data being absent from those using data opt-out services was interesting. He followed up by writing: "This is what you'd expect from a legally operating data aggregator service."

**Leo:** Yeah. This is what they do, yeah.

**Steve:** Well, but a legally operating data aggregator service does honor deletion requests. So he says: "It's a minor point, but it does support my claim that the data came from NPD." In other words, you know, if this was all dark web data aggregation that didn't actually come from NPD, then this sort of DeleteMe-style data opt-out wouldn't have had an effect. It did, which means that it came from some legally operating data aggregator that, you know, that was breached.

**Leo:** Is that the irony of this? We know it's real because they're legal. So awful.

**Steve:** I know. So he says: "None of the data discussed so far contains email addresses." Now, we should note Troy has a strong email bias because that's what Have I Been Pwned keys on. So he talks about email a lot, and it matters to him. He says: "None of the data discussed so far contains email addresses. That doesn't necessarily make it any less impactful for those involved" - certainly doesn't for me - "but it's an important point I'll come back to later as it relates to Have I Been Pwned."

He says: "So this data appeared in limited circulation as early as three months ago. It contains a huge amount of personal information," he says, "even if it isn't actually 2.9

billion different people." And we know it's not because you and I, Leo, are represented there 10 times.

**Leo:** Oh, yeah, I'm in 20 rows, yeah. Yeah, yeah.

**Steve:** Yeah. He says: "And then, to make matters worse, it was posted publicly last week." Then he quotes a tweet: "On August 6th, the Wolf Technology Group tweeted: 'National Public Data, a service by Jerico Pictures Inc., suffered a massive breach. Hacker 'Fenice' [F-E-N-I-C-E] leaked 2.9 billion records" - now, notice here. Here, like, when the techies talk, you get, like, records. Well, as soon as, you know, like the non-technical press gets it, it turns into people. It's like, no - "...leaked 2.9 billion records with personal details, including full names, addresses, and Social Security Numbers in plain text." It's worth reminding people, Leo, that even though the NPD Pentester site that I linked to through grc.sc/npd, even though it politely blanks out, it redacts all but the last two digits of Social Security Numbers, they're all there in the data. This is just for the web display that those are asterisked out.

**Leo:** I can vouch that those two numbers that they do reveal are correct, for me anyway.

**Steve:** Exactly, and for me as well. So, he said: "The breach poses significant risks for identity theft and financial fraud. Jerico Pictures Inc. faces potential lawsuits and legal challenges due to the incident."

So Troy writes: "Who knows who 'Fenice' is, and what role they play; but clearly multiple parties had access to this data well in advance of last week." He said: "I've reviewed what they posted, and it aligns with what I was sent two months ago, which is bad. But on the flipside, at least it allowed services designed to protect data breach victims to get notices out to them. Inevitably, breaches of this nature result in legal action, which, as I mentioned in the opening, began immediately a couple of weeks ago. It looks like a tip-off from a data protection service was enough for someone to bring a case against NPD.

"Up to this point, pretty much everything lines up, but for one thing: Where is the 4TB of data? And this is where it gets messy as we're now into the territory of 'partial' data. For example..."

**Leo:** See, this is one thing I wanted to ask. Does this mean this NPD Pentester site bought the - where did they get the data? Did they buy it?

**Steve:** It's a good question, although, well, what we do know is that I think later down here we're going to see it was then posted publicly.

**Leo:** Oh, the whole thing. Okay.

**Steve:** So it's now - that's the worst news is no one has to buy, you don't have to be on the underground.

**Leo:** Everybody's got it.

**Steve:** It's free, baby. Yeah. Yeah. So he says, and this is where it gets messy, is we're now into the territory of partial data. And this drives Troy nuts. "For example, an 80GB corpus was recently posted to a popular hacking forum. While it's not clear whether that's the size of the compressed or extracted archive, either way it's still a long way short of the full alleged 4TB. Earlier this month, a 27-part corpus of data alleged to have come from NPD was posted to Telegram. The compressed archive files totaled 104GB and contained what feels like a fairly random collection of data. Many of these files are archives themselves, with many of those containing yet more archives."

He says: "I went through and recursively extracted everything which resulted in a total corpus of 642GB of uncompressed data across more than 1,000 files. If this is 'partial,' what was the story with the 80GB 'partial' from last month? Who knows. But in those files were 134 million unique email addresses." Now, that's what Troy likes because he wants to put that into HIBP.

He said: "Just to take stock of where we are, we've got the first set of Social Security Number data, which is legitimate and contains no email addresses, yet is allegedly only a small part of the total NPD corpus. Then we've got this second set of data which is larger and has tens of millions of email addresses" - right, 134 million unique email addresses - "yet is pretty random in appearance. The burning question I was trying to answer is, is it legit?

"The problem with verifying breaches sourced from data aggregators is that nobody willingly - knowingly - provides their data to them, so I can't do my usual trick of just asking impacted Have I Been Pwned subscribers if they'd used NPD before. Usually, I also can't just look at a data aggregator breach and find pointers that tie back to the company in question due to references in the data mentioning their service. In part, that's because this data is just so damn generic. We have first and last name, address, Social Security Number. Attributing a source when there's only generic data to go by is extremely difficult."

He said: "The kludge of different file types and naming conventions worries me. Is this actually all from NPD? Usually you'd see some sort of continuity, for example, a heap of .JSON files with similar names, or a swath of .SQL files with each one representing a dumped table. The presence of a uniquely named CSV file ties this corpus together with the one from the earlier tweet, but then there's stuff like 'Accuitty_10_1_2022.zip.'" But I should note it's A-C-C-U-I-T-T-Y. Huh? And so he says: "Could that refer to Acuity (single 'c' and single 't') which I wrote about in November?" He says: "HIBP isn't returning hits for email addresses in that folder against the Acuity I loaded last year. So, no, it's a different corpus. But that archive alone ended up having over 250GB of data with almost 100 million unique email addresses, so it forms a substantial part of the overall corpus of data."

Then he says: "The 3.6 billion gigabyte, it's 'criminal_export.csv.zip' file caught my eye, in part because criminal record checks are a key component of NPD's services," he says, "but also because it was only a few months ago we saw another breach containing 70 million rows from a U.S. criminal database. And see who that breach was attributed to? USDoD, the same party whose name is all over the NPD breach. I did actually receive that data, but filed it away and didn't load it into HIBP as there were no email addresses in it. I wonder if the data from that story lines up with this file? Let's check the archives."

He says: "Different file name, but hey, it's a" - okay, so I should have said before, that criminal_export.csv.zip file is 3,608,086 kilobytes. This other one that he filed away before, he says, "But hey, it's a 3,608,086 kilobyte file, so exactly the same size. Given the NPD breach initially occurred in April, and the criminal data hit the news in May, it's entirely possible the latter was obtained from the former, but I couldn't find any mention

of this correlation anywhere." And he says: Side note: This is a perfect example of why I retain breaches in offline storage after processing because they're so often helpful when assessing the origin and legitimacy of new breaches." Right? Because the bad guys are aggregating stuff, too. And, like, the more billions of records they have, the more money they can ask for, even if it's, you know, if it's really a Mutt data breach.

Okay. So, he says: "Continuing the search for oddities, I decided to see if I myself was in there. On many occasions now, I've loaded a breach, started the notification process running, walked away from the PC, then received an email from myself about being in the breach." He says: "I'm continually surprised by the places I find myself, including this one."

So he says: "Yep, it's an email address of mine. Yet, oddly, none of the other data is mine. Not my name, not my address, and the numbers shown definitely are not familiar to me. I suspect one of those numbers is a serialized date of birth. But of the total 28 rows with my email address on them, the two unique DoBs put me as being born in either 1936 or 1967. Both are a long way from the truth."

**Leo:** He's a lot younger than that, yeah.

**Steve:** So he finishes: "A cursory review of the other data in this corpus revealed a wide array of different personal attributes. One file contained information such as height, weight, eye color, and ethnicity. The 'uk.txt' file merely contained a business directory with public information. I could have dug deeper, but by now there was no point. There's clearly some degree of invalid data in there. There's definitely data we've seen appear separately as a discrete breach. And there are several different versions of 'partial' NPD data," he says, "although the 27-part archive discussed here is the largest I saw, and the one I was most consistently directed to by other people." That was the one that was posted in Telegram.

He says: "The more I searched, the more bits and pieces attributed back to NPD I found. If I were to take a guess, there are two likely explanations for what we're seeing. This incident got a lot of press due to the legitimacy of the initial dump of Social Security Numbers, and the subsequent partial dumps are riding on the coattails of breach hysteria. It appears that NPD may have siphoned up a heap of publicly circulating data to enrich their offering, and it got snagged along with the initially released Social Security Number data." In other words, he's suggesting that NPD themselves were just sucking up a whole bunch of random crap of much lower quality than their main offering so that they could brag about, you know, themselves.

**Leo:** Sure, why wouldn't they?

**Steve:** How many billions of people's data they have.

**Leo:** Makes perfect sense.

**Steve:** And he says: "These conclusions are purely speculative, though; and the only parties that know the truth are the anonymous threat actors passing the data around, and the data aggregator that's now being sued in a class action. So, yeah, we're not going to see any credible, reliable clarification any time soon."

Okay. So Troy's focus is understandably on his Have I Been Pwned web service and the email addresses his site uses to allow users to look up their own records. And he uses them to inform people with those email addresses when their data has appeared in a new breach. And I thought that Troy's obsessive pursuit of, and verification of, the exact source of this breached data was interesting.

As we launched into this topic I said: "Before we wrap up today's podcast, I'm going to provide everyone listening with a URL for a searchable online database containing the records from this breach." Now, it's been so exciting that Leo and I have not been able to keep that a secret.

**Leo:** We keep talking about it.

**Steve:** We keep talking about it. To aid everyone's memory, I made a shortcut for it, which begins with our standard grc.sc, you know, for shortcut, then just slash and "npd." So grc.sc/npd, of course, for National Public Data. Using that shortcut will bounce your browser over to npd.pentester.com. Pentester.com is a 100% legitimate penetration testing subscription service that can be used by websites to check their sites' security. Pentester describes itself as: "We are a cybersecurity technology platform that has sourced the tools, methods, and techniques attackers use. Our system allows owners and operators to find potential risks and exposures before the attackers do."

So they're just using it, they're using this, they acquired the NPD data. They immediately stuck it in the cloud and have indexed it and are making it available just to draw traffic to their pentester.com site in order to get some new business. So, makes sense. Okay. So that's all that is. It's just a quick way for anybody to go, you know, GRC.sc/npd. Tell your friends and family. When you go there, I say put your full legal first and last name in because a nickname doesn't do it. These records are from various legal documents, you know, of the sort that will include one's Social Security Number. I did that, and as I've said, and as Leo has said, the results took our breath away. The search for me revealed multiple records for every address that had been associated with me through the years, in every case with my correctly associated Social Security Number.

And again, I just want to reiterate, and make this point to your friends, it's important for everyone to understand that what's shown here is just meant to be a proof of presence of everyone's highly personal data within this massive database. The fact that this search only displays the last two digits of everyone's Social Security Number does not mean that that's all there is. No. Nor that the search fields are all there is. Again, no. Everything about us, our full physical addresses, our various past and present phone numbers, our complete Social Security Number, and a massive amount of other probably very personal data is all there for the taking. It's just not all being displayed publicly. They redacted it just to be polite.

It's unclear what the exact geographic spread of the population of this database is. From the reporting, it appears to be at least the U.S., Canada, and perhaps the UK. And since this podcast has a global listening audience, we may have many listeners who will not find themselves listed within this. But given what I've seen from poking around at people whose year of birth I know, I would be surprised if all of our domestic U.S. and Canadian listeners were not quite chagrined by their lookup of their own data, and that of others whom they know and care about. Anybody who had, you know, subscribed to DeleteMe previously, like Lisa did, will probably be very pleased that their money is being well spent because they're apparently not going to find themselves there.

So it is my sincere hope that the ridiculously massive scope and scale of this breach, I mean, when our government's representatives put themselves in here and find

themselves there, that's going to be a bit of a wakeup call. So I think this is going to shine a very bright light upon this otherwise dark personal data aggregating corner of the Internet, and that we might well see some changes made in the laws that have allowed this practice to evolve and thrive.

The data that is out there has escaped already, and it will now forever be public. There's nothing that can be done about that now. And it's clearly wrong that it's up to individuals to pay to out-out of this personal data collection process. But Troy's finding and our confirmation that we've had so far that the data of those who had previously done so was conspicuously absent from this breach, serves as a true testament to the effectiveness of today's opt-out services. It actually works.

So if I were a young person today, for whom it is not too late to prevent one's entire personal life and history from being aggregated, sold, and eventually leaked out onto the Internet, I would seriously consider adopting paid-for measures today to require all data aggregators to remove any records they already have, and then I'd bide my time until the slow-moving legal and governmental system catches up with what's going on. This event clearly demonstrates that the way things are now needs to be changed.

As for the rest of us, whose last several decades of personal data is now so obviously out in the Internet flapping in the breeze, all we can really do is tightly lock down all access to our credit histories so that no one can apply for credit in our names. We last talked about this need on April 15th, and at that time I created the GRC shortcut of "credit" - so grc.sc/credit - which will bounce your web browser over to the Investopedia page which talks about freezing credit and provides updated links to each of the three main credit reporting bureaus where you need to freeze your credit.

It may also be that the policies of the credit bureaus will need to change, and can be made to change, so that rather than by default they are passively allowing anyone to access our credit histories unless we take proactive action, it will be incumbent upon them to first obtain our clear real-time permission to allow a specific individual or entity to have any access. But that's not the way it is today. Regardless, we know that any change will be slow and incremental, and it will take time to even get started.

So as this podcast enters its 20th year, it appears that one of our favorite phrases, "What could possibly go wrong," will continue to keep us on our toes for quite some time to come.

**Leo:** And to answer DJLookup's question in our Twitch chat, no, the data is out there. DeleteMe can't remove it because it's just out there.

**Steve:** Yes.

**Leo:** It's not with the broker anymore.

**Steve:** The point of the legal aggregator, the legal aggregators will respond, but the bad guys are laughing. They've already got it.

**Leo:** Right, they got it. Right. So as Eric Dutton says, "The horse has already left the barn." By the way, I don't know if you saw this yesterday, Krebs on Security had a piece about how this might have happened, which I thought was quite interesting. His contention, or his story is that there is a site that was almost identical to the NPD

site called RecordsCheck.net. On that site was a file, members.zip, which had in the clear names and passwords for the site's administrator. So it may really be that they just, you know, the second site was created, as apparently was the first, by an Indian website, or Pakistani, sorry, website creation company called CreationNext.com.

And so it's very possible that they created both the NPD site and the RecordsCheck.net site for Sal Verini, who's the principal who founded NPD. And this left passwords on there because that second site wasn't used. There's also, as long as we're talking stories, this just came out. I'm a Flightaware subscriber. Flightaware warned some customers' info has been exposed, including Social Security Numbers. You just can't win this game, can you. It's bad news. It really is. All around.

**Steve:** That's why I think that, I mean, the only thing we can do is keep our credit frozen [crosstalk].

**Leo:** Yeah. And the good news is, thanks to federal law, it is free to freeze and unfreeze your credit report at all the major reporting bureaus. They don't want you to do that. That's how they make their money is selling your data. You want that. So mine are all frozen, and I know yours are, and we've recommended that for a long time. So you've got that great URL, grc.com/freeze, if people want to learn more.

**Steve:** Slash credit.

**Leo:** I'm sorry. Say again?

**Steve:** It's grc.sc/credit.

**Leo:** Sorry, grc.sc/credit. And, yeah, it'll take you a few minutes. It's not hard to do.