How Revoking!

Description: What's been learned over the past week about the PKfail Platform Key misuse issue? What is "IsBootSecure?" and why does that sound suspiciously like a new piece of GRC freeware? There's plenty of news on the third-party cookie front. What's going on with Firefox and what position has the World Wide Web Consortium (W3C) taken on this important issue? Now that we're a few weeks downstream of the CrowdStrike disaster, the attorneys have come out to play. What are we learning about the legal side of this massive outage? What's been going on with GRC's incoming "SecurityNow" email system? And we finish by looking at DigiCert's recent mass certificate revocation event: Why it happened? What happened? Did it matter? Was it necessary? And how does it compare to Entrust's past behavior?

High quality (64 kbps) mp3 audio file URL: http://media.GRC.com/sn/SN-986.mp3
Quarter size (16 kbps) mp3 audio file URL: http://media.GRC.com/sn/sn-986-lq.mp3

SHOW TEASE: It's time for Security Now!. This episode with Steve Gibson blows the lid off of a number of my deeply held convictions. For instance, third-party cookies. Does Firefox block them? No. No. Even though they say they do. Does certificate revocation work? No. It turns it never has. Steve explains, shows you how to test your browser. And then finally, yes, Steve has another brand new freebie program, he wrote it over the weekend, that you're going to find very useful. There's a lot of great stuff coming up next on Security Now!.

Leo Laporte: This is Security Now! with Steve Gibson, Episode 986, recorded Tuesday, August 6th, 2024: How Revoking!

It's time for Security Now!, the show where we cover the latest security news, the show you wait all week long for. And here he is, the man of the hour, Steve Gibson, our host. Hi, Steve.

Steve Gibson: Hey, Leo. It's great to be with you once again for, well, the last episode we're recording in the TWiT Eastside Studios.

Leo: I think a lot of people probably, especially on this show, listen. And they won't notice any difference at all.

Steve: No. There will be no effect. No.

Leo: Yeah. Even if you watch, I mean...

Steve: It will be just as good as it's always been.

Leo: It's pretty much the same. Really, it's mostly me that's suffering from this. You're going to have exactly the same experience. The listeners probably have 99% the same experience. I'm the one who's going to be all alone. Who's going to bring me lunch?

Steve: I know, Leo. Leo, it's just tough. I don't know.

Leo: I told Lisa, "You're going to bring me a sandwich every day at 1:00; right?" She said, "What, are you crazy?"

Steve: We're going to have to watch cats crawling around on you.

Leo: And there will be cats.

Steve: On the video.

Leo: There will be cats. So what's up in the news this week?

Steve: Okay. So lots of good stuff. We've got a bunch of things that have been learned over the past week about the big topic from last week, which was this PK file exploit, the Platform Key misuse issue. And we're going to answer the question, what is "IsBootSecure?" That's the name of it, IsBootSecure? And why does that sound suspiciously like a new piece of GRC freeware?

Leo: Uh-oh.

Steve: Ah. Then there's plenty of news on the third-party cookie front. What's going on with Firefox, and what position has the World Wide Web Consortium, the W3C standard-setting body, finally taken on the third-party cookie issue after, as we know, Chrome basically capitulated to, okay, fine, I guess we're not going to be able to get rid of those. Also, now that we're a few weeks downstream of the CrowdStrike disaster, the attorneys have come out to play. What are we learning about the legal side of this massive outage, things like limitations of liability and so forth? Also I'm going to talk briefly about what's been going on or what is going on with GRC's SecurityNow! email system. And then we're going to finish - thus the title "How Revoking!" - by looking at DigiCert's recent mass certificate revocation event, why it happened, what happened, did it matter, was it necessary, and how does it compare to Entrust's past behavior? So I think lots of interesting information and news for our listeners this week.

Leo: Can't wait.

Steve: And of course we have an apropos Picture of the Week.

Leo: Yes. I did peek this time, I'm sorry. It's very funny. Now, time for the Picture of the Week. Steve?

Steve: So this was just a great picture one of our listeners sent me. I gave it the caption, "Is it too late to change my mind?" This shows a very unhappy person wearing a CrowdStrike Intern T-shirt, obviously labeling him as a CrowdStrike intern. He's holding up a little laptop PC with the screen open and the frowny face "your machine has just crashed" blue screen.

Leo: Oops.

Steve: And we can't really identify where he is. There's no obvious airport-ness about the background except that there's a bunch of people who have kind of got carry-on-ish things, and they're standing on industrial-strength indestructo carpet, the kind that you see in the airport where...

Leo: Oh, yeah.

Steve: You know. So it suggests that this, you know...

Leo: I've slept on that carpet a few times. I know that carpet, yeah, yeah. Wow.

Steve: So it does, it does look like this maybe had been an opportunistic shot that was taken when his flight was canceled, as well as everybody else's. And, you know, he turned his computer on, it's like, oops, me, too.

Leo: Yeah, yeah, wow.

Steve: Yeah. So I don't know that I would recommend he changes his mind, but...

Leo: Wear that T-shirt with pride.

Steve: Well, that's a good point. You might want to turn it inside-out for a while.

Leo: Actually, I would love that T-shirt. Wouldn't you like to wear that T-shirt once in a while? That would be fun. Wow.

Steve: Yeah, it could be "I survived." Yeah, yeah.

Leo: Yeah, yeah.

Steve: Okay. So last week's discussion of Binarly's discovery that by their estimate the manufacturers of around one out of every 10 PCs, even those sold recently, have never bothered to replace the "sample" and inherently insecure Platform Key which AMI clearly marked as "DO NOT USE" and "DO NOT TRUST," has continued to bear fruit. Over the weekend one of the people in GRC's newsgroup discovered that one of his recently purchased Dell PCs - I think it was last month he got this thing - did indeed contain one of the insecure "DO NOT USE" Platform Keys.

Now, since last week's podcast I received a great deal of email from listeners - and we'll be talking about a great deal of email and what that means a little bit later - who were unable to get the command-line that Binarly had supplied, and which I quoted in the show notes and mentioned last week, to work. And I'm not surprised. When I tried it myself, it was the definition of a mess. For one thing, in order for any of the UEFI commands to work, which is one of the commands you must use, an extra not-normally-present UEFI module must first be installed. So you have to do that.

And then, since the module apparently needs to run a script, the normally protective PowerShell script blocker must be disabled by disabling some of PowerShell's security. And then it turns out that, if Secure Boot is not enabled, the command doesn't work at all. So if the machine is booted from BIOS rather than UEFI, you know, it's not even working. And then, to make matters worse, the Binarly command-line silently checks the output text from the command and, like, pipes it into a string finder, well, the output might well be complaints about any of the above being a problem.

But even so, then a match is searched for the DO NOT USE/DO NOT SHIP strings, which will not be present in an error output. So as a result, this will tend to return massively false negative results, since any output which is not the Platform Key's actual certificate will not contain those strings. So even though the bogus AMI sample Platform Key might be sitting right there on the motherboard, you would never know. So longtime listeners of this podcast probably know where this is going. By Friday morning around 9:00 a.m. I had collected all of this feedback and tried using the command-line myself. So I thought, okay, this needs another one of GRC's little Windows apps to...

Leo: Oh, my god. This is why we love you, Steve. This is amazing. Unbelievable. I'm just going to write - I'll write it myself.

Steve: That's right, to let anyone quickly and easily get to the truth of what's going on. So I was all set up to continue working on SpinRite 6.1's documentation, which is my current focus since I need to get that behind me. I opened a new code project and began writing code to extract and analyze the motherboard's Platform Key. And everybody can try it today.

While I was doing that, and publishing incrementally more capable pre-releases for the gang in the newsgroups to test over the weekend, we were also playing the name game, as we call it. That collective name brainstorming had previously resulted - actually it was from someone you know, Leo, Paul Holder - it had previously resulted, it was his idea to give the fake USB drive testing app ValiDrive, which I love.

Leo: Yeah.

Steve: And oh, my god, years ago somebody else came up, I was working on that DCOM piece of freeware, and the guy said, "Well, you have to call it the DCOMbobulator." I'm like, "Oh, yes I do." Anyway...

Leo: I thought you made up all these names yourself. Aww.

Steve: I think maybe all the other ones. Those are the only two, I think, that I got from...

Leo: Trouble, what was it, Trouble in Paradise?

Steve: Oh, yeah, that was mine.

Leo: That was the first one we talked about.

Steve: Well, you and I and Kate on the old Screensavers show.

Leo: The Screensavers, yeah, yeah.

Steve: In South San Francisco, yup.

Leo: Click of Death, TIP.

Steve: Yeah, Trouble in Paradise. And OptOut and all the other things, those are my names.

Leo: ShieldsUP!, yeah, yeah.

Steve: But in this case I was coming up dry. So there was a lot of conversation with ChatGPT among those in the newsgroups. Lots of brainstorming. Finally, pretty much exhausted at the end of a long weekend of, like, endless names, we just settled on IsBootSecure? Which, you know, it's kind of a good name.

Leo: I like it.

Steve: Yeah. So if you go to GRC.com right now, there is an interim fully working version of this. You can find it under the Main Menu > Freeware Security > IsBootSecure?, or just GRC.com/isbootsecure. It's currently a very small, dare I say 21K? And actually about half of that is the signature that I had to give it. So, you know, because it's got to be digitally signed. It's a little 21K Windows executable that can simply be run on any Windows machine. And I'm not sure about Wine yet. Again, this just happened. The paint is still wet on this. And it will tell you exactly what's going on. And Leo, I see it there. You just ran it, apparently. Or you're looking at the show notes. That might be...

Leo: That's in the show notes, yeah.

Steve: Oh, okay, yeah. Because I...

Leo: I don't actually have a Windows UEFI machine to run it on.

Steve: Anyway, it detects which firmware boot mode the system is in, BIOS or UEFI. If UEFI, it determines whether the UEFI firmware is in Secure Boot mode or not, which is not obvious, you know, unless you reboot. So it shows you that. And if Secure Boot mode is enabled, and on some machines even if it's not enabled, it will then extract, which is to say if it's available, the Platform Key from the motherboard and display the certificate's most useful information - the serial number, its issuer name and subject name. That's where the DO NOT TRUST or DO NOT SHIP strings will be found, if they're present. And it displays the certificate's not valid before and not valid after dates, just because they're kind of interesting.

Leo: Did you put in the not valid after date yourself? Or is that the actual not valid after on one of your certificates?

Steve: Yeah, the...

Leo: Why I ask, that's four days before the end of the Unix Epoch.

Steve: That's interesting. That is interesting.

Leo: Yea. It's not the Epoch, which is the 19th. It's four days earlier.

Steve: Yeah. I did not put that in, and I haven't looked at the certificate format. But it might be Unix time in the cert, in which case...

Leo: I think it is, yup.

Steve: That would make a lot of sense, wouldn't it.

Leo: Yeah.

Steve: Cool. Good catch, Leo.

Leo: No, when I see 2038, my ears perk up.

Steve: Yeah, and you know, let's see, how old will we be, and what podcast will be on?

Leo: That I don't want to think about. Oh, my goodness. For people who don't know, Unix time started January 1, 1970, but it's a 32-bit integer; right? So after 32 bits it will roll over, just like the Y2K problem. And that turns out to be 3:14 a.m. UTC, on 19 January 2038.

Steve: Well, that's only 14 years from now, Leo.

Leo: So we'll not be that old. In our 80s.

Steve: At this point, we're only a little over halfway there. So...

Leo: Jerry Pournelle used to come on the show in his late 80s. I think we could do it.

Steve: Yeah. And you just have to somehow get up the stairs to the attic.

Leo: Yeah. I need one of those little elevators, it's seven steps, and it says mmmmmmmm, and I'll be ready to go.

Steve: Okay. Anyway. So right now this is just a Windows text screen which pops up. Oh, also, as an extra bonus, it displays the standard Windows certificate dialog box containing your Platform Key certificate so that any of the other details of the certificate can be examined. For a while over the weekend it was also exporting the certificate into a file. But since it didn't make sense to have it always doing that, I've turned that off for the moment. That will be coming back. So what's available right now does all that. But as I said, it does it as a simple text window since what I have was mostly meant as a tech development proof of concept.

What I'm going to do now, starting this evening after today's podcast, is turn this into one of GRC's standard little Windows GUI apps. So I'll interpret what's going on to very clearly display the important bits and show the conclusion of what's happening to its user. And then if you wish to view the full certificate, there'll be a button for that. And if you want to save it to a file, there'll be a button for that.

So anyway, since this Platform Key mess is going to be a persistent and enduring concern, since there's no telling how long it's going to take to address these bad keys, if they are ever addressed, I felt as though this was exactly the sort of freeware app that GRC should offer. It can be downloaded, as I said, right now. It's just in its pre-release text window incarnation, but it works. And I'm sure I'll be announcing the final app on next week's podcast, you know, in a week. So okay. So much for PKfail. We'll briefly touch on it again next week.

Let's talk about Firefox's third-party cookie mess and GRC's Cookie Forensics test. Two things are going on with Firefox and with what GRC's Cookie Forensics page shows. Part of the problem is that Firefox is being a bit, dare I say, "foxy," and appears to not have updated its user interface after it integrated its so-called "Total Cookie Protection" into Firefox. There are no UI controls for it in evidence anywhere. There's a note about it in the UI in a link, but you don't turn it on or off or do anything.

So as many people have noted, currently the only way to make GRC's Cookie Forensics test happy, meaning to say, oh, yup, no third-party cookies at all, is to use Firefox's custom setting and really forcibly turn off third-party cookies, like we said last week. All that's correct. At the same time, just turning off third-party cookies on other browsers works. Whereas it sort of doesn't seem to with Firefox. So that's what brought this conversation up, the whole dialogue last week.

There are a couple of interacting things going on. Most important is that I never designed GRC's testing with an awareness of the inter-site third-party cookie isolation that Mozilla has now built into Firefox under the banner of "Total Cookie Protection." As I commented when the idea first surfaced, I think this is a really terrific idea. It potentially changes the game by "stovepiping" cookie collections and aggregating them into their individual sites where they're being set.

As Mozilla described it, instead of browsers having one big cookie jar that's shared among all websites, which is the way it's already been and has always been, and of course that's the reason tracking works, is that the advertiser's cookie is in this big common cookie jar, which it's able to access no matter where the user goes. Instead, Firefox creates individual isolated mini cookie jars, one for every first-party domain that a browser is visiting.

So an Ad-Tech company is welcome to place their third-party cookie at site "A." But that cookie will be tagged with a site "A" context, and it will only be returned, again, when visiting site "A." So when a user visits another site where the same Ad-Tech company is also using cookies, that Ad-Tech company will not receive the cookie it set over on site "A" so it will not know that this visitor is the same one who was over there viewing ads. So this is a huge change, and it's a big win.

But I registered the independent domain "grctech.com" on New Year's Day of 2002 because I needed to have an unaffiliated third-party domain of my own available, you know, not a subdomain of GRC because cookies know about subdomains. I needed an unaffiliated domain which I could build around a Cookie Forensics system. And back then, there was no notion of the cookie site isolation that we have today with Firefox. As a consequence, Firefox may well be isolating cookies by domain, but my 22-year-old test doesn't know about that. It's just seeing that Firefox appears to be very reluctant to fully disable third-party cookies. It can be forced to, but it doesn't really want to. And the reason Firefox can do this is because it has the site isolation trick up its sleeve.

What we're going to need in order to verify that site isolation is working and real is an updated Cookie Forensics test that's aware of the possibility that while third-party cookies themselves might be enabled, they might not be shared with other first-party sites. And I don't recall why this was on my mind 12 days ago, back on July 25th, because that was before last week's podcast. But the question of the need to confirm the presence of inter-site cookie isolation must have occurred to me since Hover shows that I registered another independent domain, "grctech.dev," on that Thursday. I needed a second unaffiliated domain to create a second first-party site to see whether it would be able to see the cookies that had been set by "grctech.com" when visitors were at "grc.com."

So, at some point in the not-too-distant future, I'll make some time, I'll dust off the cookie forensics site, figure out how it works because I knew 22 years ago and I haven't looked at it for a long time, and then create an updated facility for the specific characterization of browser cookie handling which incorporates an understanding of inter-site isolation since this seems like a good thing. And clearly it's based on what just happened with Chrome and the EU or the UK, you know, third-party cookie handling is going to be as important today as it was back in 2002 when I first got a bug about this and decided to fix it.

Okay. But there's more. It turns out that simply blocking all cross-domain third-party cookies can break things. A more nuanced approach is going to be needed, apparently, to create a real-world solution that does not cause more harm than good. I want to share Mozilla's discussion of this since it helps to demonstrate how complex the web has become, and also how dependent upon the details of its operation other services have become. And Leo, your timing is perfect because this would be a great time to take a break. And then I want to share what Mozilla is explaining about the complexity of third-party tracking.

Leo: Absolutely. I was taking the old server out to the car. I realized that I'm running...

Steve: As one does.

Leo: As one does. I was running the web server and three, four, five Minecraft instances here in the other. And I realized, oh, I guess I'd better take that home. I hope I have enough bandwidth for your Minecraft enjoyment, but Club TWiT members get to play in our little Minecraft sandboxes. Now back to Steve and Security Now!.

Steve: Okay. So if anyone wants a good example of the definition of a kludge, unfortunately we're about to encounter one.

Leo: Uh-oh.

Steve: As I said, it turns out that simply blocking all cross-domain third-party cookies, like Mozilla's Total Cookie Protection does, it can break things. So you need more nuance. Unfortunately, there's no clean solution. So get a load of this. This is Mozilla's explanation, which is titled "Third-Party Trackers," and they're going to explain to us what they've had to do.

They wrote: "Cookies are invisible pieces of data that a website can ask your browser to store on your device. The next time you visit the same website, it can ask the browser to read that cookie. That's how a website can "remember" things, such as your preferences for that website and that you're logged on.

"Another use for cookies is to transfer information from one website to another. For example, a sales website can store information about your purchase in cookies and redirect you to a payment or a review website. From the website's point of view, the cookies created by the sales website are called third-party cookies. So the point is they're demonstrating there's a non-tracking use case for third-party cookies."

They said: "There are also several web libraries that developers use to add functionality to their websites. These libraries can set cookies on your device, too. If cookies are set by a library that's on a different domain from the website's domain, they are also third-party cookies. Again, another use case.

They said: "Popular libraries are used by numerous websites. When you visit a website that uses a particular library, that library can set a cookie on your device. If you later visit another website that uses the same library, that library can read the cookie that was

set when you visited the previous website. These third-party cookies, set and read by libraries from multiple websites, are called cross-site cookies."

So they say: "There are two main reasons websites and libraries use cross-site cookies. Cross-site tracking is by far the most common use of cross-site cookies. Trackers use cross-site cookies to collect information about the websites you visit and send them to other companies, often for advertising purposes. When you feel like an advertisement is following you while you browse, this is the result of cross-site tracking. If the same tracker is present on multiple sites, it can build a more complete profile about you over time.

"The other type of cross-site cookie is a functional cookie. Some websites rely on these cookies in order to function properly. For example, some websites may need access to cross-site cookies to let you use their service to sign into another website, like Facebook Login does this; or to process a payment for that website, Amazon Pay does that.

"Firefox's Enhanced Tracking Protection," they write, "blocks cookies from cross-site trackers and isolates cookies from all other third parties. This helps prevent your browsing activity on one website from being visible to other websites. Total Cookie Protection is now enabled by default. It is an advancement built into Enhanced Tracking Protection that works by maintaining a separate cookie jar for each website you visit."

They said: "While cross-site cookies from trackers are blocked in Firefox by default, a site may signal to the browser that it needs to use them for important functionality." What? Okay. "In this case, Firefox will allow a third-party..."

Leo: Please, can I have those cookies?

Steve: I know, Leo.

Leo: There's no functionality that third-party cookies could possibly implement that I want.

Steve: I know. This starts to get really messy. So they said: "In this case, if a website says, oh, but sir, may I please have one, Firefox will allow a third-party website to use cross-site cookies" - get this - "five times." What? I know, Leo.

Leo: You've got to use uBlock Origin. You know? Just block it all. Oh, my god.

Steve: This is so bad.

Leo: Oh, my god.

Steve: So they said: "In this case, if a site asks, Firefox will allow a third-party website to use cross-site cookies five times. Or" - I'm not making this up - "or up to 1% of the number of unique sites you visit in a session, whichever of the two is larger."

Leo: Where did they get that from?

Steve: I know. It gets worse: "...without prompting you. After that, Firefox will prompt you to block these cookies." What? "Without your consent, Firefox blocks these cookies from that point because a site requesting access that many times may be a tracker." Okay, now...

Leo: Maybe? You think?

Steve: To interrupt this horror for a minute...

Leo: Holy cow.

Steve: ...the first time I read that, I had to go back to make sure I'd read it correctly. I thought, what? This is my browser. My browser. You can't do this to my browser.

Leo: You know where some of this comes from, I think, is, at least in the EU and the UK, advertisers and sites are complaining. This is why Google says we had to stop, you know, implement that change.

Steve: Back off, yeah.

Leo: And I think that Firefox is concerned about getting attacked in the EU by the government because advertisers are complaining. And it's very disappointing.

Steve: It is. Okay. So, but Leo, it's going to still get worse. Okay. So but first let me just make clear, this means that Firefox now blocks cross-site cookies by default. They are also providing websites with a mechanism by which to ask to please have cross-site cookies turned back on - but only just a little."

Leo: I'm only a little pregnant. Just a little. A tiny, tiny bit.

Steve: It's just a flesh wound.

Leo: Yeah.

Steve: Presumably only enough to accomplish some specific task that needs to communicate with cookies across sites. So apparently they understand that if this was just a blanket request, then websites would eventually all claim to need it all the time; right?

Leo: Sure, all the time, yeah.

Steve: They just add, you know, it's like the Do Not Track. It'd be, oh...

Leo: Who is asking for it, though? Is it the third party?

Steve: Yes. It's a third party.

Leo: It's a third party.

Steve: Yes.

Leo: So if you're going on a site, and you go to a Starbucks site and have a Facebook thumbs-up like button, Facebook, which owns that button, could say, oh, please leave these tracking cookies on because - why?

Steve: Leo, it gets even worse because Firefox makes exceptions based on the domain that's asking.

Leo: Ugh. Ugh.

Steve: Oh, I know. It's so bad. It's so bad. Okay. So...

Leo: And really they're doing this so that they can still have that checkbox that says block third-party cookies. Because they know people look at that, and they won't have read this whitepaper. So they won't be - they'll think they're blocking third-party cookies when they're not. And that bothers me even more.

Steve: Right.

Leo: It's deceptive.

Steve: Right. Well, that's what we discovered last week was that GRC's forensics was saying, what? I've got a bunch of red bubbles here saying that it's not being blocked.

Leo: It's deceptive.

Steve: Even though it says it is. So presumably they, Mozilla, have done some testing of this, and they've decided that no website has a legitimate need for more than one or two, so they added some margin and set it at five before they surface a dialog to the user. Now, what? What is the user going to do about this? Like, wait, what? Do I say yes or no? You know? So, and then they also look at the number of unique sites visited during a session of a browser use, then take the larger of either five or 1% of the unique sites count during that session. So that means that the breakeven point would be after visiting 500 different sites. Right? Then once you've visited another 100, then 1% of 600 sites total would be six requests that are allowed.

Leo: Can't allow that.

Steve: So I suppose this means that if anyone is using their browser like crazy, the site they're visiting might need a bit more leeway. But a bit more is all they're going to get. You know, if all of this sounds like a horrendous mess to you, our listeners, then I would say you've been paying attention because this is a horrendous mess. This sort of heuristic is not the way the Internet and the web were designed.

The Internet is as robust as it is because it doesn't have any sort of this nonsense. You know, it was designed with clear, clean, and simple rules, which is, I would argue, why it's done as well as it has. You know, but this allow sites to request an exception, but not too many times, you know, it's the very definition of a kludge. I'm surprised they're not just tossing a coin. Heads, you can use a cross-site cookie; tails, uh, sorry, you know, better luck next time.

Okay. Anyway, they continue by writing - this is Mozilla. "Third parties will only be able to prompt you if you interact with the website you are on." Okay. "For example, if you visit Dogs.com and select the payment field, Amazon Pay cross-site cookies may be allowed to facilitate that transaction. After that, Firefox will ask if you want to keep allowing them." Oh, and they say: "You can view the Permissions panel to see if a website has been allowed or denied permission to use cookies, by clicking the permissions icon in the address bar."

So now there's going to be a new permissions icon in the address bar. If you deny the request, the third party will not be able to use cross-site cookies during that session. But if you refresh or reload the page, the third party may prompt you again. From the Permissions panel for a site, you can click the X or revoke previously allowed access to cookies."

Wow. So website visitors, who have no idea what's going on behind the scenes, are being expected to make on-the-fly allow/deny decisions about whether or not the third party appearing at any given site should be allowed to use cross-site cookies. This is going to be interesting.

They said: "Firefox automatically allows third-party websites to use cross-site cookies on the first five or so - or so - websites you visit. For example, Amazon Pay would be able to use cookies on Old Navy, Blick, Dog.com, and a handful of other sites without asking you for permission." But, oh, I guess that means if you encounter Amazon.pay a sixth time, during something, a session, then, whoa, wait a minute, do you want to allow this because you've been using it a lot. Wow.

"If a third-party continues," they write, "if a third party continues to use cross-site cookies across multiple sites, this becomes a signal to Firefox that the third party might be a tracker. At that point, the third party would have to prompt you to ask for permission to use cross-site cookies."

They said, and they actually used the word "heuristics": "There are other rules," they said, "('heuristics') that will make Firefox temporarily grant access to cross-site cookies to certain websites. These rules are designed to enable special use cases such as single sign-on services and usually require some special interaction such as a top-level redirect or a user interaction, making it difficult for trackers to exploit them."

Okay. So, boy. It is a tangled mess. Presumably these "heuristics" that Mozilla is talking about for Firefox are things like Firefox knowing in advance the domain names of large single sign-on providers so they automatically get special permission and can use cross-domain cookies without limit because that's their legitimate business purpose. So now

that puts Mozilla in the position of deciding whose business model will and will not receive more permissive treatment under Firefox. I mean, we're losing the browser that we have always loved.

Leo: Thank god for Gorhill. I don't know what we will do. He'd better not die or anything.

Steve: Well, he did post recently that he's about to be in trouble with uBlock Origin, but over on Chrome.

Leo: Yeah. It won't work with Manifest v3.

Steve: That's correct. So, you know, at least we do still have Gorhill here on Firefox. And I'll be talking about that next week. So what's happening is that the industry is now tying itself in knots because cross-site information sharing we want and cross-site information sharing we don't want are using the same mechanisms.

Mozilla's Third-Party Tracker explanation begs the question: In light of this, how could Chrome's Privacy Sandbox designers have possibly imagined the elimination of all cross-site third-party cookies? Mozilla's just finished telling us all the reasons that you can't eliminate them because they're having to poke a whole bunch of holes in this in order to keep things working. So annoying though it may be to purists, Mozilla highlights some compelling use-cases which have arisen for needing cross-site information sharing in some form, sometimes.

Okay. So the W3C finally weighs in. Amid all this, the World Wide Web Consortium, the W3C, has finally weighed-in on third-party Cookies. So just to remind everyone and update everyone on the W3C, Wikipedia's first sentence or two says: "The World Wide Web Consortium (W3C) is the main international standards organization for the World Wide Web. Founded in 1994 and led by Tim Berners-Lee" - and this is after he left CERN - "the consortium is made up of member organizations that maintain full-time staff working together on the development of standards for the World Wide Web. As of March 5th, 2023, the W3C had 462 members. The W3C also engages in education and outreach, develops software, and serves as an open forum for discussion about the web." So these are the guys that are defining the APIs that our browsers support, and the standards that make the World Wide Web go.

"About a week and a half ago, on July 26th, the W3C's official statement was titled: '"Third-Party Cookies Must Be Removed."' Okay, that's never been said before. Third-party cookies must be removed. The article's Abstract says: "Third-party, aka cross-site cookies, are harmful to the web, and must be removed from the web platform. This finding explains why they must be removed, and examines the challenges in removing them." So a finding is what they're calling this thing that they published on July 26th. It examines the challenges in removing them.

"We highlight some use cases that depend on third-party cookies and offer some examples of designed-for-purpose technologies that can replace them." In other words, instead of you, like, overloading in an object-oriented sense, in like using cookies for a purpose they were never intended, the web is going to come up with designed for purpose technologies, replacement technologies, that don't track, don't allow tracking, cannot be abused, and will allow third-party cookies to be terminated."

So, they write: "Specification authors are expected to ensure they do not undermine the benefits of removing third-party cookies when proposing new web platform technologies." And their introduction is interesting because they call out Chrome. They write: "We consider privacy to be a core design principle and differentiator for the web platform. Many browsers have restricted third-party cookies." They said: "(See WebKit and Mozilla." And they said: "Unfortunately, not all browsers have followed suit." And at that point they had a link to the article that the Privacy Sandbox Group had published four days before this which acknowledged that third-party cookies would not be removed from Chrome.

Okay. So TAG is their abbreviation for their Technical Architecture Group, TAG. So the piece continues: "TAG calls for all browsers to drop support for third-party cookies, as this provides an opportunity to further improve the privacy preserving features of the web platform. Removing third-party cookies from the web platform is not without complications. There are use cases for third-party cookies that need to be preserved, and pitfalls we need to be careful to avoid in doing so. This document sets out some aspects that specification editors and implementers should be aware of in order to make sure we ultimately leave the web better than we found it after third-party cookies are removed."

Okay. So this all sounds wonderful. This might be significant. If nothing else, it likely provides some cover for any browsers that wish to move in this direction. There's now an official position from the Internet's major web standards body unequivocally calling for the end of third-party cookies. Their "Leaving the web better than we found it" section says: "We support removing third-party cookies from the web platform, and we embrace the opportunity to improve the privacy features of the web. When we review new technologies to replace third-party cookies, we need to ensure that the replacements do not recreate the same pitfalls to privacy.

"The TAG considers each new technology proposal both individually and as they fit together with the web platform as a whole. The web must be cross-platform, so multi-implementer/multi-browser support and developer support for privacy-related specifications is essential if they're going to achieve the goal of increasing privacy on the web. When we consider whether something makes the web platform better, we should be explicit about what that baseline for comparison is. Is a proposal better for privacy when compared to usage of third-party cookies? Or when compared with a web free from third-party cookies altogether? What about when some user agents restrict third-party cookies, but others do not?

"We want to emphasize that as any replacement proposals progress, implementations should have a strong commitment toward, and reasonable time frame for, removing third-party cookies. We are also wary of new mechanisms being introduced that could be abused together with cookies, fingerprinting surface, or other tools, for greater privacy invasion. Given this context, we see an urgency to have a strict timeline for the removal of third-party cookies. We are strongly in favor of innovations to build sustainable business models on the web platform, but an in-depth discussion of the various possibilities are outside the scope of this document. From an architectural standpoint, web standards should avoid encoding particular business models" - which unfortunately Mozilla has just done - that are available to authors, publishers, and web content creators.

"In conclusion, when accommodating changes caused by the removal of third-party cookies, we should avoid introducing new technologies that, when deployed either individually or in combination, effectively preserve the status quo of harmful tracking and surveillance on the web."

Okay. So I could certainly be reading too much into this. I'm not nearly enough of an insider to be able to venture a guess about what this might actually mean for changes in

the future. But, boy, does it sound good. They're using all the right words. And I think what this is saying is that those organizations, those entities like Amazon Pay, that are aware they are dependent upon third-party cookies, need to get serious about working with the emerging standards. And the W3C did mention a handful of them. Among them was Chrome's Privacy Sandbox. But it turns out there are others. There is other work in progress, not from a single company, but from multiple standards bodies that will provide a solution.

So what we could see happening would be that, you know, Amazon Pay and Facebook Logon and so forth would switch to using a non-third-party cookie standard, thus protecting themselves from the end of third-party cookies, which the W3C is now calling for. So this lays out a template for the shape of the future approaches that would have a chance to succeed. You know, they said: "We are strongly in favor of innovations to build sustainable business models on the web platform." And of course we know what that means; right? That means that sustainable business models, on one hand Amazon Pay and using the web and third-party authentication for logon, probably also means that, like Google, they understand that advertising powers much of the Internet, and that targeting ads makes those ads enough more effective that it can, as we've seen, double a site's revenue from advertising. At the same time, I lost count of the number of times they used the word "privacy." So it's clearly not their intent for anything like the status quo to remain. But will anything change?

And it's definitely worth repeating what I noted last week, which is that there are significant enterprises employing many people whose entire purpose is to deliberately and surreptitiously violate the privacy of everyone who ventures out onto the Internet with their computer. You know?

Leo: It's true, yes.

Steve: That's exactly what it is; right? And these days, who doesn't venture out onto the Internet with their computer? And equally unfortunate is the fact that these enterprises have paying customers who have some use for the aggregated private data of individuals - which is to say, all of us. So this W3C action of very clearly working toward putting an end to third-party cookies can be expected to see more opposition, much as it appears happened with Google in Europe.

Leo: What is the - they said in there there are uses of third-party cookies that are legit. I'm wracking my brain.

Steve: So apparently, and I didn't dig into it, but apparently Amazon Pay is using third-party cookies in order to link itself.

Leo: Ah. So this is the case, and you do this, too, it's not unusual for a site to have images from another server, to have data from multiple servers. So those servers would look like third parties when you're on a first-party page.

Steve: By definition, they are third parties, yes.

Leo: They are literally, yeah.

Steve: Yes.

Leo: So that makes sense. So that's probably some sort of related thing where, well, I need to set a cookie from the image server so that I don't send you that image again or whatever. But it's not going to come from this page because I'm a first party.

Steve: Right.

Leo: So I can see that, yeah.

Steve: Right. There are non-tracking, currently legitimate applications. Now, and so their point is that cookies have been used legitimately because they're there. Right? I mean, again, third-party cookies are being used to create these inter-domain connections, just because they're convenient. They're there. But it would be possible for a site like Amazon Pay to use a designed-for-purpose, you know, a non-third-party cookie solution.

Leo: Yeah, that makes sense.

Steve: And so that's what the W3C is calling for is an alternative technology that everybody who currently is using third-party cookies for good will be able to switch to. So then third-party cookies can be turned off. And the problem is, lots of companies are using them for privacy-invading purposes, and they're going to fight back.

Leo: Right. And the EU's listening because they say, well...

Steve: Yes.

Leo: These are journalistic enterprise support.

Steve: Legitimate business. They say they're legitimate businesses, yeah.

Leo: For advertising, yeah.

Steve: They have, you know, little babies they have to feed. Okay.

Leo: So really it's a conundrum. It really is. But I can see now, yeah, there are lots of servers, many servers you go to now who have multiple endpoints. And those are all third-party endpoints except for that one main server. So that makes sense. I can see that. That's those JavaScript things that everybody loads, things like that.

Steve: Right.

Leo: Yeah.

Steve: Time to take a break, and then we're going to update on what has been - what has happened to CrowdStrike?

Leo: What the heck?

Steve: As a consequence of that little, that little problem a couple weeks ago.

Leo: I'm curious.

Steve: Think anybody's upset, Leo? Do you think anybody called their attorneys?

Leo: We should preface that, as you did a couple of weeks ago when we talked about this, with the fact that up till then CrowdStrike was a very well-known, very well-respected security firm; that they did a lot of important research on the Internet.

Steve: Leo, believe it or not, one group of I don't know what it is going on, they're actually suing because their reputation was so good. They're saying, "You misled us."

Leo: Yes.

Steve: You know, you made all that money for us because you were just pulling the wool over our eyes.

Leo: Yeah. You were so good until you weren't.

Steve: Wow.

Leo: That does happen. All right, Steve. Let's talk about the sad tale of CrowdStrike.

Steve: Well, predictably, yeah, predictably the lawsuits and class actions have started up in the wake of CrowdStrike's little bitty problem. Did anybody notice that? I think that some details of them would be of interest to our listeners. The first is a shareholder lawsuit blaming CrowdStrike for the entirely predictable significant drop in the company's stock price. Apparently, the Plymouth County Retirement Association of Plymouth, Massachusetts, now regrets having invested so heavily in CrowdStrike.

The coverage of this, after removing the redundant stuff that we already know, it says: "Austin-based cybersecurity firm CrowdStrike is facing a class action lawsuit from shareholders who claim the company defrauded them by concealing how its inadequate software testing could cause a global computer outage, resulting in a big hit to the share price and overall market value. According to the July 30th complaint filed in the United

States District Court in the Western District of Texas, CrowdStrike's Chief Executive George Kurtz characterized CrowdStrike's Falcon software as 'validated, tested, and certified during a conference call on March 5th.'

"The plaintiffs say these statements were 'false and misleading' because CrowdStrike allegedly failed to properly test and update" - actually, they did update, and that was the problem - "allegedly failed to properly test and update its Falcon software before rolling it out to customers. The complaint alleges CrowdStrike 'instituted deficient controls in its procedure for updating Falcon, and was not properly testing updates to Falcon before rolling them out to customers.'" Okay, I think everybody would agree with that. "CrowdStrike did not disclose that 'this inadequate software testing created a substantial risk that an update to Falcon could cause major outages for a significant number of the Company's customers.'

"The complaint alleges: 'Such outages could pose, and in fact ultimately created, substantial reputational harm and legal risk to CrowdStrike. As a result of these materially false and misleading statements and omissions, CrowdStrike stock traded at artificially high prices during the Class Period."

Okay. So they're upset over the terrific management, operation, and reputation of CrowdStrike before the event, that had presumably made them so much money. And now they're suing because what was once overly inflated is apparently no longer. The article notes that: "Following the outage, the legal complaint says CrowdStrike's share price fell 32% percent over the next 12 days, wiping out $25 billion of market value. As of July 31st, CrowdStrike shares are worth $231.96. They closed at $343.05 on the day before the outage.

"The lawsuit, led by the Plymouth County Retirement Association of Plymouth, Massachusetts, seeks unspecified damages for holders of CrowdStrike Class A shares between November 29th of 2023 and July 29, 2024. The class action also alleges that Delta Air Lines' hiring of an attorney to represent them in seeking damages from the company, along with Kurtz's being called to testify before the U.S. Congress over the incident, caused CrowdStrike's share price to fall." So they're also unhappy that other things that allegedly hurt CrowdStrike's reputation also hurt the stock price. Okay.

The article says: "CrowdStrike said, in a statement provided to media outlets: 'We believe this case lacks merit, and we'll vigorously defend the company.' Speaking at the time of the outage, CrowdStrike Chief Executive George Kurtz said: 'We identified this very quickly and remediated the issue.'" Period. He added that its systems were constantly being updated to ward off "adversaries that are out there."

Okay. So that's the first of the entirely predictable legal actions that are underway. The second one surrounds the excessive degree of damage caused to Delta Airlines. As we know, whatever it was that happened to Delta kept many of their flights on the ground far longer than any of their other competitors who, like most of the rest of the planet, removed the bad file, rebooted, and resumed operations. But not Delta. Something happened at Delta.

Just yesterday, on Monday, the publication CIO Dive's headline read "CrowdStrike rebukes Delta's negligence claims in a fiery letter." And the subhead of their piece was "After the airline said it was considering legal action, CrowdStrike said Delta's contract capped the cybersecurity provider's liability to 'single-digit millions.'" They weren't any more specific. But sorry, folks, the $500 million you're claiming in damages, we're not going to get there with our contract.

Okay. So the article says: "CrowdStrike struck back forcibly against Delta Air Lines' claims of negligence and misconduct in a letter sent Sunday" - just last Sunday - "to the

firm representing Delta Air Lines, signed by attorney Michael Carlinsky." And I should mention, Leo, that he addressed the letter to David Boies. So, you know, some high-powered legal talent there. "So this is the latest in what has become a public dispute" - that is, between Delta and CrowdStrike - "following recovery from the global CrowdStrike outage, which was caused by a faulty software update," as we know, "pushed to Windows servers on the 19th of July.

"Delta was the hardest hit major airline carrier. Its disruption lasted longer and reached further than what United Airlines, American Airlines, and others experienced. As the airline grappled with the scale and length of the outage, it moved to shift some of the blame publicly against the cybersecurity provider. Delta's CEO Ed Bastian told CNBC last week the airline was considering legal action, seeking compensation for the $500 million in costs that the airline had endured." Yes, we're grounded, and it's hurting. "He said: 'We're looking to make certain that we get compensated, however they decide to, for what they cost us,' Bastian said."

Okay. So CrowdStrike pushed the recovery responsibility back on Delta, interestingly. The airline declined CrowdStrike's help with systems recovery, according to the letter, which was shared with CIO Dive.

Okay, now, no one disputes CrowdStrike's ultimate culpability here. Even they don't. So what appears to be at issue is matters of degree. I tracked down the text of CrowdStrike's Sunday letter to Delta. So this is CrowdStrike saying: "Dear David," as in David Boies, you know, superpower well-known attorney. "I am writing on behalf of my client CrowdStrike, Inc. in response to your letter dated July 29th, 2024, in which Delta Air Lines, Inc. raises issues and threatens CrowdStrike with legal claims related to the July 19th, 2024 content configuration update impacting the Falcon sensor and the Windows Operating System (the 'Channel File 291 incident')."

He writes: "CrowdStrike reiterates its apology to Delta, its employees, and its customers, and is empathetic to the circumstances they faced. However, CrowdStrike is highly disappointed by Delta's suggestion that CrowdStrike acted inappropriately, and strongly rejects any allegation that it was grossly negligent or committed willful misconduct with respect to the Channel File 291 incident. Your suggestion that CrowdStrike failed to do testing and validation is contradicted by the very information on which you rely from CrowdStrike's Preliminary Post Incident Review.

"CrowdStrike worked tirelessly to help its customers restore impacted systems and resume services to their customers. Within hours of the incident, CrowdStrike reached out to Delta to offer assistance and ensure Delta was aware of an available remediation. Additionally, CrowdStrike's CEO personally reached out to Delta's CEO to offer onsite assistance, but received no response. CrowdStrike followed up with Delta on the offer for onsite support and was told that the onsite resources were not needed. To this day, CrowdStrike continues to work closely and professionally with the Delta information security team.

"Delta's public threat of litigation distracts from this work and has contributed to a misleading narrative that CrowdStrike is responsible for Delta's IT decisions and response to the outage. Should Delta pursue this path, Delta will have to explain to the public, its shareholders, and ultimately a jury why CrowdStrike took responsibility for its actions swiftly, transparently, and constructively, while Delta did not.

"Among other things, Delta will need to explain why Delta's competitors, facing similar challenges, all restored operations much faster; why Delta turned down free onsite help from CrowdStrike professionals who assisted many other customers to restore operations much more quickly than Delta; that any liability by CrowdStrike is contractually capped at an amount in the single-digit millions; every action, or failure to act, by Delta or its

third-party service providers, related to the Channel File 291 incident; and the design and operational resiliency capabilities of Delta's IT infrastructure, including decisions by Delta with respect to system upgrades, and all other contributory factors that relate in any way to the damage Delta allegedly suffered." Oh, and yes.

"In light of Delta's July 29 letter, CrowdStrike must also demand that Delta preserve all documents, records, and communications of any kind - including emails, text messages, and other communications - in the possession, custody, or control of Delta, its officers and directors and employees concerning, but not limited to, the items listed below. As I am sure you can appreciate, while litigation would be unfortunate, CrowdStrike will respond aggressively, if forced to do so, in order to protect its shareholders, employees, and other stakeholders. CrowdStrike's focus remains on its customers, including Delta. CrowdStrike hopes Delta reconsiders its approach and agrees to work cooperatively with CrowdStrike going forward, as the two sides historically have done."

So, okay. You want to blame us? Let's figure out, Delta, why you stood alone as an outlier in the amount of damage that you are alleging that you took relative to all of your peer airlines. Industry estimates are that the CrowdStrike outage to the various Fortune 500 airlines will be around a total of $860 million or so, and apparently that's on average about $143 million per airline. But it appears that there is in fact a contractual upper limit that does cap CrowdStrike's liability. And that's not in dispute.

The article then offers some interesting background about this, quoting a guy named Scott Bickley, the advisory practice lead at Info-Tech Research. Scott explained, he said: "The standard limitation of liability (which is interestingly known as the LOL clause, limitation of liability) for most SaaS (software as a service) agreements caps liability at the actual funds spent on the subscription over a set period of time, usually the previous twelve months." That's what I said before, that often these things just say that you're entitled to getting your money back, essentially, is what they're saying.

"Many enterprises will negotiate a multiple of this amount," he says, "or a set capped amount." Bickley said that CrowdStrike's liability is likely to match annual spend or a multiple of annual spend if the clause was negotiated. He said: "Many large enterprises surprisingly do not negotiate these terms and default to using the language in the vendors' agreements, which of course benefits the vendor." He said: "Delta is likely going to pursue damages outside of the LOL cap and may rely on other legal arguments to bring the claim to a third-party dispute mediation or litigation."

So this is exactly what's expected. One last note that is, sadly, happening, individual travelers who were inconvenienced by this are also suing, even though travel delays and flight cancellations and rerouting is quite common. Yesterday Reuters reported: "CrowdStrike's legal troubles from last month's massive global computer outage deepened Monday" - that's just yesterday - "as the cybersecurity company was sued by air travelers themselves whose flights were delayed or canceled.

"In a proposed class action filed in the Austin, Texas federal court, three fliers blamed CrowdStrike's negligence in testing and deploying its software for the outage, which also disrupted banks, hospitals, and emergency lines around the world. The plaintiffs said that as fliers scrambled to get to their destinations, many spent hundreds of dollars on lodging, meals and alternative travel, while others missed work or suffered health problems from having to sleep on the airport floor." And yes, we did see the carpet on the airport floor, and that's not where you want to sleep.

"They said CrowdStrike should pay compensatory and punitive damages to anyone whose flight was disrupted, after technology-related flight groundings for Southwest Airlines and other carriers back in 2023 made the outage 'entirely foreseeable.'" So they're saying, okay, these things happen, so we want to be compensated. And, you know, we know

what the effect will be; right? After last years' troubles, I would be surprised to learn that the fine print in all passenger agreements doesn't already include a blanket liability waiver for any failure to fly caused by any foreseeable or unforeseeable events. And if it already doesn't, it certainly will in the future.

You know, this is just the way the world is now. When you sign forms for a relatively minor procedure in a hospital, the fine print explains that "We're really going to do the best we can to keep you alive; but, you know, stuff happens. So if something happens, we'll be really sorry. Sign here, and good luck." You know, it's always a bit unnerving, but what are you going to do?

So anyway, just as I was finishing up this podcast I received email from a listener, Rob Woodruff - and actually since then a bunch more - informing me that as a CrowdStrike customer he had just received email from CrowdStrike's CEO, and that their much anticipated "Root Cause Analysis" had finally been completed. I had no time to dig into it for today, and I do not plan to spend much time on it next week since we've already given this lots of time and coverage. But if we learn anything new, I will briefly share what we learned from it.

And Leo, let's share a break, our last break.

Leo: Yes.

Steve: Then I'm going to talk a little bit about GRC's email and how that's going, and we're going to take a look at how DigiCert handled a certificate revocation. I guess it's an emergency, but...

Leo: We've talked about this before and how hard it is to do. You know, it's not...

Steve: Well, Entrust doesn't do it.

Leo: Yeah.

Steve: You know? It's so hard that Entrust just says, uh, no.

Leo: Nah, can't do it. We thought about it. We decided not to. Now back to Mr. Steve "Tiberius" Gibson and "How Revoking!" Hmm.

Steve: Well, first I just wanted to mention that today's podcast filled up before I could share any of the terrific feedback that I've been receiving, in great volume, from our listeners. The incoming listener feedback system has been an utter success, and I've been drowning in thoughts, notes, pointers to news, and other great content. I just checked my "securitynow" inbox, and we're currently at 1,512 pieces of email received. I guess it's, like, been about four weeks now.

I'm mentioning this because I'm torn by my inability to reply to this much incoming email. Initially, I was trying to. But then I looked at how much time was going into creating replies and saying even not much more than "Thanks so much for sending that." So I need to explain that everyone should know that even in the absence of any reply

from me, if the email did not bounce back to you, I have it. I've read it, and I'm sure I appreciated it, really. When I feel that I have to reply, I tend not to read them since my seeing the feedback creates the feeling of an obligation to reply that I'm just not able to service. That's the way I am.

So I want to sincerely, really sincerely thank everyone here collectively who has written and who will write. I really value this feedback, and I am seeing what you have sent. You'll probably often sense it in the effect that you have on the podcast afterwards. Bits of obscure news that I may have missed if you hadn't told me about it. So please no one ever imagine that because you don't receive a reply, that means I never saw what you took the time to send. If you took the time to send it, I will have taken, at least taken the time to read it, even if I was unable to say so explicitly. So really, thank you.

One last point is that I've received a number of complaints from listeners who were forced to write to Sue or Greg at our Sales or Support email because they were unable to find the special "securitynow" email address anywhere on the site. The reason for that is it's not meant to be public, and it's not meant for GRC's non-podcast visitors. It's just for us, and I want to keep it between us. And the email address should not be too difficult to remember. It's "securitynow@grc.com."

And to make it even easier, I've set the "reply-to" address of every one of our weekly podcast mailings to that "securitynow@grc.com" email address. So anyone who receives the weekly email can simply reply to any email they receive, and many of our listeners just do that. So anyway, again, I've been feeling like the mail's piling up. I just wanted to tell everybody how much I appreciate it. The system's working. I am seeing everything. I've been feeling guilty that I can't thank everybody. So here's a collective thank you.

Okay. Now, revocation. Just as last week's podcast was happening, my favorite certificate authority, DigiCert, announced their discovery of a mistake their systems had made during the process of Domain Control Validation, or DCV. I talked about this last week with regard to Entrust, noting that SSL.com, the Certificate Authority from whom Entrust plans to purchase interim certificates on behalf of their customers, will still need to have their own customers, Entrust's customers, prove their control over their pending certificate's domain name directly to SSL.com. And I noted at the time that there are various ways to get that done. The weakest of these is to use email received by and sent to the domain in question. Better ways involve adding DNS records to the domain, which the Certificate Authority can then remotely pull and verify.

As we know, on the one hand we have Entrust, who has for years effectively refused to revoke certificates which had been shown to be mistakenly issued, out-of-compliance certificates mistakenly issued to their customers. And as we'll see today, on the other hand, we have DigiCert, who takes the CA/Browser Forum requirements to heart, and immediately jumped into action when a truly insignificant mistake was discovered. But as this shows, a rule is a rule. Many of our listeners understood this difference and sent notes to me with variations on the subject, "This is the way it's supposed to be done." You know, congratulating DigiCert, essentially. Consequently we have, you know, "How Revoking!" for today's podcast title.

The Hacker News explained the whole event, complete with updates as of Sunday, and there was one. Their headline was "DigiCert to Revoke 83,000+ SSL Certificates Due to Domain Validation Oversight." But just wait till you hear what it was. I've mixed their reporting with my own clarifications and some small additions.

So here's what we have: "Certificate authority DigiCert has warned that it will be revoking a subset of SSL/TLS certificates within 24 hours, due to an oversight in how it verified if a digital certificate is issued to the rightful owner of a domain." So they wrote:

"The company said it will be taking the step of revoking certificates that do not have proper Domain Control Validation (DCV)."

DigiCert said: "Before issuing a certificate to a customer, DigiCert validates the customer's control or ownership over the domain name for which they are requesting a certificate using one of several methods approved by the CA/Browser Forum. One of the ways this is done hinges on the customer setting up a DNS CNAME record containing a random value provided to them by DigiCert." And this is a 32-character, right, it's a massive gibberish random number, random value. "DigiCert then performs a DNS lookup for the domain in question to make sure that the random values are the same." So they tell the customer, put this in. Here. Here's some gibberish as a DNS CNAME record. Add this to your DNS. We'll query your DNS once you say you have. Only you who control that domain could have done that. So we're good to go.

"The provided CNAME record is for a subdomain of the user's domain, and the random value from DigiCert is prefixed with an underscore character" - so it begins with underscore - "so as to prevent a possible collision with an actual randomly named subdomain that might use the same random value. Note that domain names are not allowed to begin with an underscore, thus the collision prevention." In other words, DigiCert's, or the CA/Browser Forum specs say thou shalt begin a name with an underscore because it is an illegal domain name, but it's good for this purpose.

So what DigiCert found was that it had failed to include the underscore prefix before the random value used in some CNAME-based validation cases. Okay. So whether or not that matters is really an edge case. There is an instance where it can matter, which I'll explain in a second. But it is a minor problem. But this demonstrates that in the certificate authority space a rule is a rule, and we certainly beat up on Entrust over their flagrant violation of the rules. This wasn't flagrant by any similar means, but okay.

Okay. So the Hacker News continues: "DigiCert's mistake has its roots in a series of changes DigiCert enacted starting in 2019 to revamp their underlying architecture, as part of which the code to add an underscore prefix was removed and subsequently 'added to some of the paths in the update system,' but not to one path that neither added it nor automatically checked if the random value had a pre-appended underscore." Apparently there was some instance where the user was expected to do that, but they didn't verify that the user had.

"So DigiCert said: 'The omission of an automatic underscore prefix was not caught during the cross-functional team reviews that occurred before deployment of the updated system. While we had regression testing in place, those tests failed to alert us" - boy, this sounds a little familiar - "to the change in functionality because the regression tests were scoped to workflows and functionality instead of the content/structure of the random value. Unfortunately, no reviews were done to compare the legacy random value implementations with the random value implementations in the new system in every scenario. Had we conducted those evaluations, we would have learned earlier that the system was not automatically adding the underscore prefix to the random value where needed.'"

Okay. So the Hacker News said: "Subsequently, on June 11th of this year, DigiCert said it revamped the random value generation process, eliminating the manual addition of the underscore prefix within the confines of a user-experience enhancement project, but acknowledged it again failed to 'compare this UX change'" - you know, the user experience change - "'against the underscore flow in the legacy system.'

"The company said it didn't discover the non-compliance issue until 'several weeks ago,' when an unnamed customer reached out regarding the random values" - in other words, somebody noticed, you know, one of their customers out in the field - "used in validation,

prompting a deeper review." Yeah, I'll bet. "DigiCert noted that the incident impacts approximately 0.4% of the applicable domain validations which, according to an update on the related Bugzilla report, affects 83,267 certificates issued across 6,807 customers.

"Notified customers are recommended to replace their certificates as soon as possible by signing into their DigiCert accounts, generating a Certificate Signing Request, and reissuing them after passing Domain Control Validation. The development prompted CISA to publish an alert - interestingly, not over any danger that this incredibly minor and inconsequential mistake may have had, but stating that 'revocation of these certificates themselves may cause temporary disruptions to websites, services, and applications relying on these certificates for secure communication.'

"In a later update, DigiCert said: 'DigiCert continues to actively engage with customers impacted by this incident, and many of them have been able to replace their certificates. Some customers have applied for a delayed revocation due to exceptional circumstances'" - like they're on the International Space Station or something, I don't know - "'and we are working with them on their individual situations. We're no longer accepting any applications for delayed revocation.'"

And so Hacker News finishes: "These include customers who are operating critical infrastructure, who it said 'are not in a position to have all their certificates reissued and deployed in time without critical service interruptions.' And DigiCert further noted that all impacted certificates'" - yes, all 83,267 of them - "'regardless of circumstances, had been revoked as of August 3rd, 2024, at 7:30 p.m. UTC.'"

So here we have an example of a company that's doing it right. Even in the face of a quite significant pain being caused to their own customers by the sudden and essentially emergency requirement to revoke and then replace, well, hopefully replace before they're revoked, their certificates. And what's more bracing is that there was nothing whatsoever wrong with those certificates. You know, they weren't formed incorrectly. They didn't have bad fields or anything. It was the fact that the domain validation missed something which was meant to prevent a collision with real certificate names.

So this lack of a leading underscore in no way allowed anyone to nefariously obtain a certificate fraudulently. Now, I should say I wrote that, and since I wrote it, one of our listeners who is on the emailing list got the show notes and read this, said, "Uh, Steve, actually there is an instance where there is a collision with a non-underscore name." So for there to be a collision, you would need to have a situation where somebody was creating a CNAME record under a domain they don't control. And it turns out that DynDNS works this way. Right? Dynamic DNS allows people - you normally have, like, you're able to choose from a number of top-level domain names, and then you choose your own subdomain under that domain name.

Well, if that subdomain was deliberately set up to collide with a CNAME record for the same domain name - well, okay. So what this means is that a user of DynDNS could deliberately create a Dynamic DNS subdomain matching what DigiCert had given them, lacking an underscore, that would then allow them to get a certificate for that DynDNS domain. Now, that's not good news. And in fact it turns out that the DynDNS people are deliberately preventing the use of an underscore, whereas the certificate authorities are going to be deliberately requiring the use. In other words, this is all, once again, kind of a big mess.

Anyway, the point is, DigiCert jumped on this, fixed the problem, sent the painful email to all, what did I say it was, 68,000, no, 6,807 customers, immediately actually revoked the certificates, got it done in some cases within a day, and where customers needed longer within five, which is the requirement in the CA/Browser Forums, got it done, you

know, took responsibility, took the hit, informed their customers, and did it immediately. So, yes, they did it right.

I should note that, since this issue with the DynDNS possibility occurring, history has been checked. This never happened. So there was never any abuse of it. And now the DynDNS people will be disallowing the use of a leading underscore, whereas all of the use of CNAME records will require one, and this will keep those two uses from colliding.

Finally, I'm out of time today. But I want to note that, as many of this podcast's longtime listeners know, and as Leo will remember well, the topic of web server certificate revocation has been another one of those long-term hobby horses of mine, much like third-party cookies, because the entire web browser revocation system is a total and utter joke. It is completely broken, it has always been broken, and it has never worked. Despite some half-hearted attempts from time to time to fix it, it is broken, and it never got fixed. Web browsers do not know when the certificates they're receiving from web servers have been revoked.

Now, if we have some time next week we're going to update ourselves on this space, on this issue, because the truth is there may have been some use cases where certificates were being used where something actually would have known if they'd been revoked, and that would have caused a problem. But mostly no. Not a single browser.

In the meantime, try aiming your browser, whatever browser you have, at revoked.grc.com. I brought that interesting and revealing test back to life last week in anticipation of this topic. So revoked.grc.com. Go there with your browser. If it tells you, sorry, this site is untrusted, well, that's amazing. That used to happen in some cases with Firefox if you forced it to do online certificate revocation checking and told it not to fail open and a bunch of other things. The revoked.grc.com site provides the browser with a certificate I revoked on purpose last week. No browser that I have tested works right. Even Safari, that used to, no longer does.

Leo: [Indiscernible] doesn't either.

Steve: So if you see that page, sorry, folks, you shouldn't.

Leo: What should I be seeing? Like you can't go here?

Steve: A big warning. A big, scary...

Leo: Should I see that? Is that what I should see? There you go.

Steve: No. Whoops.

Leo: That's - I shouldn't be seeing that.

Steve: No. That means that your browser is showing you a certificate that has been revoked.

Leo: It's based on Chromium. Let me try Safari.

Steve: Safari used to work. On iOS, that was the one that I remember, like, saying wow, that's pretty good. I tried it last week after I brought the site back to life and issued a new - this certificate is only a week old. But I revoked it immediately after DigiCert made it for me.

Leo: Oh, there you go, Safari. Nice job. It says, if you can see this, and apparently you can - show this one. That's Safari. Let me see Firefox. This is terrible. This is terrible, Steve.

Steve: I know, Leo.

Leo: This is terrible.

Steve: It doesn't actually - certificate revocation does not work.

Leo: Apparently, because it crashed Safari, as well. Let's see here. Not now. Revoked.grc.com. Oops, I said erevoked. That failed. Oh, no. You've got to show these. Oh, no. They're all the same. Everybody. That's amazing.

Steve: Yup.

Leo: Everybody.

Steve: And if you go to ssllabs.com and have it check the server, revoked.grc.com, because SSL Labs, that's Ivan Ristic's really terrific site. Ssllabs.com. And then tell it to check revoked.grc.com. You will see what it thinks because it actually checks to see what's going on.

Leo: You're supposed to see an error; right?

Steve: No. It's actually doing a deep test of, like...

Leo: Testing all the different TLS systems?

Steve: It tests, yeah, different protocols. It checks the certificate chain. It checks which protocols you accept. And I think if you scroll down you'll see some red there.

Leo: Oh, yeah. Insecure. Revoked. Do not trust. Do not use this in your code.

Steve: And every one of our browsers ignores that. They do not know that certificates have been revoked.

Leo: It's broken.

Steve: It's always been broken.

Leo: Broken.

Steve: You remember when I went off on a binge about this years ago.

Leo: Yeah. But also remember that Google said, yeah, we're not even going to try to support this because it doesn't work.

Steve: Actually, it embarrassed them, and so they added my certificate as a special case.

Leo: Oh. Oh. That's not the way to do it.

Steve: No. I just changed the certificate, and it came back, and then they didn't bother doing it again because they knew.

Leo: That's hysterical. So they wrote an exception just for you.

Steve: Yes, they did.

Leo: Ugh. Shame, shame, shame. So what makes this revoked? Just that you - it's not that it's invalid, it's just that you specifically said to DigiCert "I revoke this."

Steve: Well, so like for example, if the certificate got loose and malware was using it, then you would want to revoke it so that malware wasn't able to use it to impersonate your site.

Leo: Can't, apparently.

Steve: Or DigiCert just revoked, you know, for what good it did, they just revoked 83 thousand, what was it...

Leo: 8,300.

Steve: No, it's 6,800 customers 83,267 certificates. And it didn't matter.

Leo: Didn't matter.

Steve: They didn't have to revoke them because...

Leo: Because the browser doesn't know.

Steve: Browser doesn't care. It doesn't know.

Leo: Unbelievable.

Steve: Yup, doesn't work.

Leo: Right now it's testing Bleichenbacher. So we're still testing. What Bleichenbacher is, I'm glad you tested for Bleichenbacher. I don't - this is SL - you're right. This SSL Labs is pretty thorough.

Steve: Yeah, it's really terrific.

Leo: Yeah, yeah. I mean, it's taking a long time to go through this. Wow.

Steve: Yeah, because it turns out it's going back and forth handshakes with the server.

Leo: Right.

Steve: Those are all exploits against TLS that it's testing.

Leo: Right. Very nice. But the main test is, is this certificate revoked? Yes. It is.

Steve: It is revoked. And the browser doesn't know.

Leo: Unbelievable. Wow. That is really - so how long has this page been up? I know this is a new cert, but you've had this page before; right?

Steve: Oh, yeah, yeah. We did multiple podcasts about it back in the day.

Leo: I remember, yeah.

Steve: And so that page will show that I just edited it because I did in order to bring it back to life.

Leo: Right. If you're reading this, good luck.

Steve: You may not be where you think you are.

Leo: Good luck.

Steve: What that means is you may not be where you think you are.

Leo: Yeah, yeah. Because the certificate's meaningless.

Steve: Yup.

Leo: Wow. What a world. See, what people count on is that you don't exist, that people don't listen to this show, that they don't know this stuff, and they're just kind of blithely going along, well, yeah, we have certificate revocation. Of course we do. It's in the spec. We must have it. Wow. No, we don't.

Steve: No.

Leo: This is why you listen to the show. This is why Steve is a national treasure. This is why you should join Club TWiT to support Steve and his work. I don't know what else to say. You could buy ads on the show, too. GRC.com, that's his website, the Gibson Research Corporation. If you want to be on the mailing list or send an email, same page, GRC.com/email. He'll validate your address. Then you can or cannot subscribe to the mailing list, which you probably want to do. But you can also now email him because your address is good, verified.

Steve: Yeah, we have a really neat one-click unsubscribe. What happens is I'm also getting subscribers, just random GRC visitors. And they go, oh, yeah, I want to get email from GRC. And so they subscribe to all three. Then on a Tuesday morning, when they actually get the email, they look at it and go, I don't want this. And so there is a single-click, an instant unsubscribe. And so I think I got, like, six unsubscribes after doing the emailing because these are people who had joined the list during the past week and thought, no, this is dumb. I just want to get, you know, I want to find out, well, actually I'll do my first mailing as soon as the IsBootSecure? app is ready.

Leo: We should, by the way, there are a number of people saying, oh, my god, this means the whole certificate thing is bogus. No. Certificates still are valuable for encrypting your traffic between a site, to verify the site is who they say they are. It's just it's that this revocation thing does not work. So you can't revoke a site that has - or a cert that has leaked out or is somehow malicious. You can't take it back. But I guess it does mean you shouldn't necessarily trust the certificate on any site.

Steve: That's the whole point. That's why we have revocation. Where we're headed is, and Google has mentioned this before because they're the brokest of all. The Chrome has

never even, I mean it was the pretense of what Chrome was saying that upset me so much back when we were talking about this because they said, oh, we have CRL set and, you know, we're managing this ourselves. I said, no, you aren't. And I created this to demonstrate it. So where they're headed is 90-day certs like Let's Encrypt uses.

Leo: So they just expire.

Steve: Exactly. You can't revoke them, but you don't have to wait very long for them to die a natural death.

Leo: But that gives a cert as much as 90 days to be wrong.

Steve: Bogus, yes. But it's better than nothing, which is what we have now. And unfortunately it does mean that all certificate management has to be automated. You know, right now they...

Leo: Oh, because of the 90 days, yeah, yeah, yeah.

Steve: It used to be five years, then three years, then - now we're one year and one month, you know, 13 months. And so eventually we will probably get down to 90 days. And no one's going to want to do that by hand. They're just going to want to use the ACME protocol to do that.

Leo: Right. If you want to share that last little bit about certs with other people, you can go to YouTube.com/securitynow and just clip it. That's a great way to do that. Steve has copies of the audio of the show, 64Kb audio as well as 16Kb audio, which is of course much smaller, for the bandwidth impaired. He also has transcripts, if you like to read along. And he has the full show notes, which is what I show from time to time on the screen. All of that's at GRC.com. While you're there, pick up a copy of SpinRite, the world's finest mass storage performance enhancer, recovery utility, and maintenance utility. Did I get it all?

Steve: That's good, yeah.

Leo: 6.1's the current version. You can get it right now at GRC.com. And it's his bread and butter, Steve's bread and butter. So if you don't have a copy, everybody who has storage of any kind should have it. Steve's got a lot of free stuff there, like ValiDrive and DCOMbobulator. I don't know anybody who really needs DCOMbobulator anymore, but...

Steve: No.

Leo: No. You don't get a lot of downloads for that anymore. ShieldsUP!, though, every time I get a new router I test it with ShieldsUP!. That's a must. And on and on and on. He's really the king of freebies, including this new one. What do you call it again?

Steve: IsBootSecure?

Leo: IsBootSecure? IsBootSecure? It sounds like Boris Badenov version of utility.