The Top 10 Cybersecurity Misconfigurations

Description: How many people have downloaded GRC's latest freeware so far? Do we believe what 23andMe have told the world about the leak of their customers' personal and private data? What are the stats regarding all aspects of cyberattacks? How's the Brave Browser doing? Where and when is Google surreptitiously embedding tracking links into Google Docs exports? What high profile enterprise was also compromised by the Progress Software MOVEit SQL injection? What additional web browser just added and announced its support for Encrypted ClientHello? What change did Google just make with the release of their Pixel 8 family of smartphones? What cyber initiative did the U.S. Congress just overwhelming pass? What's "dwell time" and why do we care? And that's just the news. We'll also be entertaining many of our listeners' questions, then starting into the first part of our examination of a really terrific document that was just published by the NSA and CISA.

High quality (64 kbps) mp3 audio file URL: http://media.GRC.com/sn/SN-943.mp3
Quarter size (16 kbps) mp3 audio file URL: http://media.GRC.com/sn/sn-943-lq.mp3

SHOW TEASE: It's time for Security Now!. Steve Gibson is here. More victims of the MOVEit vulnerability. Would you believe it? We'll also talk about the future of the Brave browser, seems like things have gotten a little rocky. Steve is very suspicious of 23andMe's explanation about their breach. And then we're going to talk about CISA and their Top 10 Misconfiguration settings. Maybe something you want to think about going forward. It's all coming up next on Security Now!.

Leo Laporte: This is Security Now! with Steve Gibson, Episode 943, recorded Tuesday, October 10th, 2023: The Top 10 Cybersecurity Misconfigurations. This show is brought to you by members like you. Thanks.

It's time for Security Now!, the show where we cover the latest news in security with this guy right here, he's the king, the man, the voice. If you're not listening to Security Now! every week, you're missing it. Steve Gibson. Hello, Steve.

Steve Gibson: So is it the latest news in security or in insecurity?

Leo: In insecurity, yeah.

Steve: Yeah. One of my annoyances is when someone says you only have one choice. Well, isn't a choice, does that mean two, a choice between two things?

Leo: Yeah. Oh, that's - now it's going to annoy me.

Steve: Have one choice?

Leo: You're right. If you only have one choice, it's not a choice.

Steve: You're right, exactly. So you really have no choices if you have one. And if you have two choices, then that's only one choice.

Leo: So maybe that's what they mean. You have two choices, but there's only one choice.

Steve: Well, why don't they say what they mean, Leo? That's just no good.

Leo: I was recently commissioned by a woman you and I both know well to write a teaser for Security Now!. And we were doing these trailers for all the shows. I'm not sure where they show up. They show up somewhere important.

Steve: You don't know who sees it.

Leo: I don't know what they're for. But you know what? Lisa says record them, I record them. And as I'm writing, I'm thinking, so in each show I'm trying to write, you know, what I think makes this show an important show. And here's what I say for Mr. G: "When it comes to online security, zero-days, clever and not-so-clever hacks, Steve Gibson is the acknowledged expert. If it's your job to protect the network, you need, you must, you must listen to Security Now!." And I think that's accurate.

Steve: Well, actually that fits perfectly with today's topic. The NSA and CISA got together and produced a document based on the results of their extensive red and blue team operations.

Leo: Oh, that's got to be interesting.

Steve: Oh, it's really good. In fact, it's so good that already the title was the Top 10 Cybersecurity Misconfigurations. I did not have room to say Part 1, or the title would have fallen off the edges of the PDF, and it wouldn't fit in the lower third of the video, which, you know, is all important. But we're going to start into it this week and finish it up next week. But we've got a lot of other stuff to talk about for Security Now! Episode 943 for October 10th.

One question is, how many people have downloaded GRC's latest freeware so far? Do we believe what 23andMe have told the world about the leak of their customers' personal and private data? What are the stats regarding all aspects of cyberattacks? How's the Brave Browser doing? Where and when is Google surreptitiously embedding tracking links into Google Docs exports? What high-profile enterprise was also compromised by the

Progress Software MOVEit SQL injection? What additional web browser just added and announced its support for Encrypted ClientHello? What did Google just change with the release of their Pixel 8 family of smartphones? I heard you guys talking about it on MacBreak Weekly, and it's great news.

What cyber initiative did the U.S. Congress just overwhelming pass? And what's "dwell time," and why do we care? And that's just the news. Then we'll also be entertaining many of our listeners' questions, and then starting into, as I said, the first part of our examination of a really terrific document that was just published by the NSA and CISA. And of course we've got a great Picture of the Week. So I think another great podcast for our listeners.

Leo: You know, I was just thinking as you go through those questions, that would be a great security news quiz. If you've been following the week's security, a couple of them I went, oh, I read that. Oh, I knew that. But a lot more I did not.

Steve: Are you in the loop.

Leo: If you did not score 100% on our security quiz, stay tuned.

Steve: Stay tuned for the answers.

Leo: All your questions will be answered in this thrilling, gripping edition of Security Now!. I think we're ready for a picture, Steve.

Steve: We are ready for the Picture of the Week.

Leo: I'm scrolling up. I'm going to see it for the first time here. 4000 years ago man built the pyramids. Oh, my god. That's terrible. All right. Tell us what we're looking at here.

Steve: Okay. So the text that I put with this is "4000 years ago man built the pyramids. And it's been downhill ever since."

Leo: This is horrible.

Steve: So this is a picture, I don't know if this is the bottom of a stop sign or a parking meter...

Leo: A light pole or something, yeah.

Steve: It looks like it, yeah, it's something municipal. You kind of see the yellow curb running along the side. And so this is on the sidewalk. And it's really so sad. So this pole comes to a base, a square base, rounded rectangle or rounded square base with four

mounting holes in each of the four corners of the square. Unfortunately, the location of the bolts coming up...

Leo: They missed the holes. Well, they got - what's weird is they got one. Obviously the first one; right? They put that in first.

Steve: So I think maybe it's, you know, the other caption would have been, "Oh, you meant that in centimeters."

Leo: I think they drilled one hole. That one worked. But they forgot to drill the other ones? Or maybe they put them in the wrong spot.

Steve: Well, I think what actually happened is that someone stole the first one of these, and it had a larger base.

Leo: Oh, it had a larger base. Oh, I bet you're right.

Steve: Because you can see that each of the three that didn't make it, they're also in a square pattern, but they're too far away to make the other three holes compared to the one that actually has a bolt to it. And frankly, Leo, when you look at the way this is mounted, is it any surprise that the first one got stolen? I mean, you know, you just come along with a regular hex wrench and, you know, make off with whatever this was that was...

Leo: Well, why do you want it, is the question?

Steve: Oh, god.

Leo: Oh, that is so horrible.

Steve: Another picture of humanity at its best.

Leo: So that basically this heavy light standard is held down with one bolt and then washers on three corners.

Steve: And then washers, exactly, yeah.

Leo: Oh, that's terrible.

Steve: Yeah, not good. Okay. But what is good is that since I know many of our listeners have been waiting for it, I want to start off by noting that GRC's latest freeware utility, that Leo, you had a hand in incubating when you were horrified with the news of fraudulent drives...

Leo: Yeah.

Steve: It exists. About somewhere around 14,000 copies have been downloaded. About 4,000 a day at the moment. I put it online on Friday evening. And in the show notes these are actually the 12 USB drives I purchased from Amazon. Every single one of them is fraudulent.

Leo: What?

Steve: They're brand new, just purchased. On the ValiDrive page I provide the 12 URLs to Amazon to these drives on Amazon for just for people to see for themselves. I mean...

Leo: Here's a telling point. Like this SanDisk was supposed to be a SanDisk Extreme microSD card, SanDisk isn't there. It just says "Extreme." Right?

Steve: Correct. Correct.

Leo: So these are all like - and Amazon does nothing to stop this, no doubt.

Steve: No, nothing. But, I mean, even - some of them came in really nice-looking, you know, like they were trying to copy Apple's packaging. And I thought, oh, well, this'll be legitimate. Eh, no.

Leo: So these are all 1TB, and none of them actually were 1TB?

Steve: Not a single one. In the lower right you see a 2TB. Over in the lower left is a 256GB. Not a single one of them was what they claimed.

Leo: How big were they mostly?

Steve: They were all 32 or 64GB.

Leo: You know what, they're probably little microSD cards in there; right?

Steve: Well, exactly. So it's enough to take a FAT file system format. They all look legitimate. One, the fancy one, even had - it actually had a Lightning connector on one end and a USB on the other. And on the drive was stored a PDF showing, like, how to use it. I mean, it looked fantastic. And it only had one quarter of the storage that you thought you were getting. And the others are far less.

So anyway, I wanted to let everybody know that it exists. We have had some - I think I've seen two instances where Windows Defender said, you know, like said that it was a virus and quarantined it. But, you know, 14,000 downloads, and it's not affecting most

people. So as I said, this is - unfortunately we're in a land where it's going to take a while for the software to age enough so that basically its hash, its digital signature, acquires a reputation. And it's the reputation that protects software from these false positives now, and nothing else.

Oh, the other really interesting thing I found that I think some of our listeners will find interesting is that some drives test pretty quickly under ValiDrive. Other ones test, like, even legitimate drives very slowly. And ValiDrive shows you if it's waiting to read or waiting to write. That is, waiting for a read to return or a write request to return. And after you've processed a drive, it gives you a report with a lot of statistics about the read and write speed, the average, the median speed, the standard deviation, and also something known as the coefficient of variance, which is the standard deviation over the mean. So you actually get a sense for how much spread there is in the reading times and the writing times. But what's really interesting is that many of these drives are surprisingly slow.

Well, okay. The reason is, and I've mentioned this about the technology of NAND flash storage, is in order to write, to erase or to write data into NAND memory, you're having to push through a layer of insulation in order to inject or remove electrons. That requires a degeneration of a higher voltage than normal, typically around 20 volts is required. But USB devices only have access to five volts. So there is something known as a voltage pump or a charge pump inside all NAND devices. It has to be turned on, and then it has to basically pump up the five volts to 20 volts through a switched capacitor system in order to be able to start writing. That takes time.

And so what ValiDrive is showing people is the amount of time it takes before that drive is able to do any writing. SSDs typically scream along because they've got much more sophisticated electronics in them. But even some low end modest thumb drives will also run very quickly because they've just been engineered well. But there are definitely some thumb drives that just crawl along with ValiDrive because - so the reason this isn't normally a problem is that operating systems tend to write, you know, whole multi-megabyte files or multiple files at a time. If you're like copying a directory or a whole bunch of photos over to a thumb drive, it's all being written at once. So that charge pumping to get ready to write only has to happen once, and then that time is amortized out over all of the writing that you're doing.

But ValiDrive is like the worst case. It only reads and writes little 4K pieces at a time. And it's switching back and forth between reading and writing and reading and writing. In order to read, you have to dump that high voltage. So it gets dumped, then the reading happens. Now ValiDrive wants to write that same spot. So you have to wait again for the NAND to basically charge itself back up in order to be able to write. So anyway, as always seems to happen when I get into these things, as happened when I started working on SpinRite 6 and the ReadSpeed utility came out, and we found out that many SSDs were slower at the front of them because they were having more trouble reading data that hadn't been written for a long time because that's where the operating system was, we always end up finding some interesting new stuff.

Leo: Now I'm going to go - I have a drawer full of these suckers.

Steve: Yeah, it's interesting.

Leo: I'm going to download ValiDrive and go through all of them.

Steve: And of course it's 95K written in assembler.

Leo: Yeah, of course.

Steve: And you don't have to install it and blah blah blah.

Leo: Wow.

Steve: Anyway, neat new utility. And the people over in the newsgroup were saying to me when I said, "What do you think about this," they said, "Oh, yeah, yeah, you've got to do this, it's going to be very popular." I think they're probably right.

Leo: 14,000's a good number, and that's just probably people who are in the newsgroups or listen to this show. I'm sure that if the general public finds it we've got to put it up on Reddit or somewhere. It'll be [crosstalk].

Steve: I did tweet the news on Friday, and so...

Leo: Nobody's on Twitter. Come on. Just a bunch of nut jobs. Sorry.

Steve: Speaking of nut jobs.

Leo: Yes.

Steve: Last week, after internal private customer data was found circulating on hacker forums, the well-known, actually I think it's number one, DNA aggregation, analysis and profiling service 23andMe announced that the accounts of some of its users had been accessed through credential stuffing attacks. And of course we've talked about the nature of those before. So here's - because I think this is - I need to pick this apart a bit.

Here's the first three paragraphs of their announcement. They said: "We recently learned that certain 23andMe customer profile information that they opted into sharing through our DNA Relatives feature was compiled from individual 23andMe.com accounts without the account users' authorization. After learning of suspicious activity, we immediately began an investigation. While we're continuing to investigate this matter, we believe threat actors were able to access certain accounts in instances where users recycled their login credentials - that is, usernames and passwords that were used on 23andMe.com were the same as those used on other websites that had been previously hacked.

"We believe that the threat actor may have then, in violation of our Terms of Service, accessed 23andMe.com accounts without authorization and obtained information from certain accounts, including information about users' DNA Relatives profiles, to the extent a user opted into that service."

Leo: Yeah, now I'm just looking at my - because I am one of those people. I opted in. I've used 23andMe. They were a sponsor. I got my whole family to do it. And of

course we're all sharing our data with one another, so that means we qualify. My password is, I'm looking at it from one - probably from LastPass, but right now it's in Bitwarden. And it's long. And it's strong. So it's unlikely - and I'm sure I didn't use it anywhere else. So that means I'm probably okay; right? It's password reuse that's the problem.

Steve: I don't believe them.

Leo: Oh. No.

Steve: The more I've thought about this, the less it appears to pass the smell test.

Leo: Oh, no.

Steve: My problem is the sheer quantity of...

Leo: How would they get that many records?

Steve: Yes.

Leo: Yeah.

Steve: Yes. The quantity of customer records that were apparently retrieved. The attacker claims to have seven million records, which is half...

Leo: They'd have to try like 10 times that number of passwords; right?

Steve: Well, users. I mean, they're saying seven million customer records.

Leo: Oh, so well part of the reason they could have a huge number is because, for instance with me, my account, you get in my account, you're going to get all the relatives; right?

Steve: Right, right.

Leo: So it's a multiplier of some number.

Steve: Exactly. So but they're saying they had half of 23andMe's user base.

Leo: Yeah, no, that can't be credential stuffing. That can't be.

Steve: And also, and Leo, the nature of the data fields leaked looks very much like the internal raw database records. The initial data leak on hacker forums consisted of one million records which was data for Ashkenazi Jewish people.

Leo: Yeah, I'm one of those people, 4%.

Steve: And millions more are, we're told by the hackers, now available for purchase. So if we're to believe 23andMe - think about this. If we're to believe 23andMe, they're saying that all the attackers did was logon as some other valid customers, and that they were then able to obtain apparently complete records for millions of other 23andMe users? In other words, any 23andMe user can login as themselves, then do the same thing that the attackers did? Really? That's what they want us to believe?

Leo: Yeah. Well, now I'm concerned because the other thing they said is no genetic information was leaked. Now I'm really concerned.

Steve: We can't believe anything they've said because this just does not - and I noticed, if you read what they said, they were very careful to say we believe, we currently believe this.

Leo: Yeah.

Steve: This is what we think has happened.

Leo: We believe. We think. Those are waffle words.

Steve: Yes. So they left themselves an out. And again, they're asking us to believe that you could log in with your valid credentials and then do this. Because they're saying that's all the bad guys did. So, nah, I don't know. It's not my business to pursue this. But, you know, many people are quite sensitive to the disclosure of their very personal and private, especially their genetic information.

So I hope that someone in a position of authority digs a bit deeper and asks 23andMe to further explain what to me, from what we've seen so far, looks a lot more like public release or public response cover-up of a far larger problem. And, you know, these guys should be grownup enough to know better than to do that, if that's what happened. Again, don't know. But it just seems really suspicious that apparently seven million records were exfiltrated, apparently by just logging in as other people, as like regular customers. I mean, they could have created their own account and then logged in as themselves and then done this, we're being told. So I don't know. Something seems fishy to me.

Microsoft published their Digital Defense Report for 2023. And the numbers make for some interesting reading. It has a bunch of interesting facts worth sharing. I've just got the bullet points here. So, for example, of the 78% of IoT devices with known vulnerabilities on customer networks, okay, first of all 78% of IoT devices have known vulnerabilities on customer networks. 46% are not patchable. In other words, I mean, I guess it's good that half of them are. That seems like an optimistic number to me. But

still, half of them can't ever be fixed. They just don't offer patching as an option, even though they're IoT devices and 78% of known vulnerabilities. Whoops. Yeah.

Since 2019, attacks targeting open-source software have grown on average 742%. Since 2019. So in four years attacks targeting open-source software have grown on average 742%. Not surprising; right? We talked about it last week. You find out that with regard to the Exim mail server, there's an open source server. You learn that there's a vulnerability in the authentication somewhere. So hey, you've got the source, bad guys. Dig through it and figure out where it is. Much easier than reverse engineering it from the binary. Also fewer than 15% of non-governmental organizations (NGOs) have cybersecurity experts on staff. Fewer than 15%. What, one in eight? Wow.

Coin-mining activity was found in 4.2% of all incident response engagements. So not so high. You know, not like it's 100%. But still, you know, so, what, that's one in 25, so not much at all. 17% of intrusions involved known remote monitoring and management tools. In other words, so 17%, we know the way, you know, they're not having to hack things. They are getting in through remote monitoring and management, you know, like remote desktop protocol, unfortunately. But other things, too.

So-called adversary in the middle, this is now what we're calling - we used to call it man in the middle. Now it's adversary in the middle. So it's AITM instead of MITM. Adversary in the middle phishing domains grew from 2,000 active domains in June of last year to more than 9,000 by April of this year. So yes, you know, phishing is unfortunately proving to be a constant source of ways for the bad guys to get in.

156,000 daily business email compromise attempts were observed between April 2022 and the same time in 2023. 156,000 daily business email compromise. So, boy, that's another huge attack target. 41% of the threat notifications Microsoft sent to online services customers between July 2022 and June 2023, so basically summer to summer, one year, 41% threat notifications went to critical infrastructure organizations. In other words, those that you don't want to have hacked because they're like in charge of the electric grid or power generation or who knows. Hydroelectric dams.

The first quarter of 2023 saw a dramatic surge in password-based attacks against cloud identities. So again, not surprising, things are moving to the cloud. That's where the passwords are. Microsoft blocked on average 4,000 password attacks per second over the past year. So this is just - it's a constant barrage of credential stuffing. Approximately 6,000 multifactor authentication fatigue attempts were observed per day. So multifactor authentication is under attack, as well.

The number of token replay attacks has doubled since last year, with an average of 11 detections per 100,000 active users in Azure Active Directory Identity Protection. So doubled. DDoS attacks are on the rise, with around 1,700 attacks taking place per day, cumulating at up to 90 terabits of data per second. So huge attacks. I mean, you're just - you're gone off the 'Net. It's just, I mean, it takes, you know, these are just massive attacks, and 1,700 of them per day.

State-sponsored activity pivoted away from high-volume destructive attacks in favor of espionage campaigns. 50% of destructive Russian attacks observed against Ukrainian networks occurred in the first six weeks of the war. In other words, there was an initial surge of Russian attacks against Ukraine, but that has since diminished. Ghostwriter continues to conduct influence campaigns attempting to sow distrust between Ukrainian populations and Eastern partners who support Kyiv, both governmental and civilian. And finally, Iranian operations have expanded from Israel to the U.S. to target Western democracies and NATO.

And stepping back from all of that, the thing that struck me most is, unfortunately, this is not an industry taken as such in which we would like to see such growth. I mean, it's not like it's the same this year as it was last year. Or like anywhere you would call this explosive growth. Unfortunately, it's exploding in cybersecurity and, you know, and network attacks and intrusion attempts. I mean, it's a massive increase over the course of just one year. It would be, you know, way better if things were pretty much as bad this year as they were last year. But unfortunately that's not what these numbers suggest. We're seeing real growth in cybercrime activity, and unfortunately in its success, across the spectrum. Wow.

It's unclear what, if anything, this news might mean for the future of the Brave browser, but Brave Software confirmed that it has just laid off 9% of its staff across departments.

Leo: Yikes. That's not good.

Steve: So, yeah. And we don't have an absolute number. We don't know how large the staff is that 9% of it was let go. The company didn't indicate how many people were affected, but said its decision was driven by the tough economic climate, saying: "Brave eliminated some positions as part of our cost management in this challenging economic environment. Several departments were affected, amounting to 9% of our staff."

They had already been taking steps to bolster their revenue sources this year. In April, Brave Search dropped Bing's index to start relying on its own indexing solution. So, you know, it saved money there. In May, the company released its own search API for clients, with plans starting from $3 per 1,000 queries, that is, uses of its API. It also offers different plans for AI data model training, data with storage rights, spellcheck, and autosuggest. Last month, Brave introduced image, news, and video results as part of its Search API. So it's doing what it can. And Leo, you'll be honored to know that Brave has named their forthcoming AI assistant after you.

Leo: What, Leo?

Steve: It's called Leo.

Leo: Well, it's better than TWiT, I guess.

Steve: It is. While the plan is for Leo to be available to all users, Leo will also have a premium tier. Leo, you're getting a premium tier.

Leo: I think we need a premium tier, yes, indeed.

Steve: That'd be great.

Leo: Yes.

Steve: To offer features like higher rate limits and access to more conversation models. That paid premium tier will help pay for the cost of API access and hosting for everybody

else. So anyway, you know, Brave is making a, dare I say, brave attempt at cutting, you know, like forging their own path in an environment where that's not easy to do.

Leo: I support them.

Steve: Yes. We would like to not see a monoculture among browsers.

Leo: Well, that's why I use Firefox because...

Steve: Yes, yes. And I do, too.

Leo: Brave is still Chromium; right?

Steve: Yup, exactly. I do, too, as we know.

Leo: Yeah.

Steve: Someone named Joe posted on Fosstodon. He said: "Today I found out that Google Docs infects HTML exports with spyware, no scripts, but links in your document are replaced with invisible Google tracking redirects." He said: "I was using their software because a friend wanted me to work with him on a Google Doc. He's a pretty big fan of their software, but we were both absolutely shocked that they would go that far." Okay, now, I was curious to see, since the show notes are written in Google Docs.

Leo: We do, we use Google Docs, big-time.

Steve: Yeah. So I was curious. So I first exported these show notes as a PDF, which is what I normally do. And as far as I could tell, their embedded links were clean. But I wanted to see whether I could substantiate Joe's claim. So I then exported the show notes as HTML. And sure enough the embedded URLs all point to Google, with the original, visible URL as a parameter, along with a bunch of tracking gibberish. I have a picture of it in the show notes. So if you look at it, it's grc.sc/, you know, it's one of my shortcut URLs. But what's actually there is www.google.com/url?q=, then my URL, followed by &sa=D&source=editors&ust= and some gibberish serial number, and then usg and another gibberish ASCII thing. So, you know, that's nothing I did or wanted or created. And what that means is, if that HTML, anyone clicks on links in that HTML, Google is tracking them. I have no idea why they would care to, want to. I guess it's just because they can.

Leo: It probably has to do with editing permissions and things like that, too; right? I mean, they want to know if you - because you can save something that they can't edit, or they can edit, or comment on and all that. So I guess it...

Steve: So, yeah, I wouldn't go so far as to call this spyware.

Leo: It's functionality.

Steve: I wouldn't call it spyware. Well, it's not functionality because all it's doing is it's bouncing the person who clicks through Google and then to the destination URL. So, I mean, it's counting or tracking or who knows what-ing. Anyway, it's clearly deliberate, and it sure doesn't seem like it's any of Google's business what links are clicked in the future on an HTML export of something created by their Docs. I mean, they don't do it where they can't. It doesn't look like it's in docx for exports, or in PDFs. Anyway, I just thought everyone should know, and that Joe was right in his post about what was going on over on Fosstodon.

Our podcast 928 of June 20th, so that was, what, a couple months ago, June 20th, was titled "The Massive MOVEit Maelstrom." That's one that Jason co-hosted, Leo.

Leo: What a mess this has been. It's a gift that just keeps on giving.

Steve: Oh, my god. And just all it was in this day and age, as our listeners know, was an SQL injection. It was just a SQL injection.

Leo: Yeah.

Steve: Anyway, it just came to light as a result of a disclosure that Sony filed with its U.S. authorities that Sony Interactive Entertainment was among the now more than 2,300 companies - 2,300 - who were impacted by the exploitation of that SQL injection vulnerability. So anyway, Sony said that nearly 7,000 of their families and employees were affected as a consequence of that information leak. So...

Leo: And obviously they learned nothing from the Sony Pictures Entertainment leak of three years, I mean, this is just...

Steve: First thing that occurred to me, yeah.

Leo: God.

Steve: And while we were talking about Encrypted ClientHello last week, this was a week ago today, on that day Mozilla, our browser, yours and mine, Leo, was busy announcing that in Firefox 118, which I just checked is what I was sitting in front of when I was writing this last night, is now supporting Encrypted ClientHello. And I really like their page. I have - I grabbed two pictures from their announcement page because it has such a very clean and simple graphic to highlight the difference between non-Encrypted ClientHello and Encrypted ClientHello. It just shows, you know, here's your phone running Firefox. And with standard SNI, Server Name Identification, example.com is flying through the air toward the web server, and then the TLS channel brings it back. But with Encrypted ClientHello, it's encrypted in a channel, you know, in a little tunnel, going to the server, and encrypted coming back. So very clean.

Leo: Yes.

Steve: And they also summed up last week's deep dive, the one we did, quite succinctly. They simply wrote: "ECH uses a public key fetched over the Domain Name System (DNS) to encrypt the first message between a browser and a website, protecting the name of the visited website from prying eyes and dramatically improving user privacy." And then later they explain that ECH must be paired with DNS over HTTPS (DoH) to secure and hide that initial public key fetch, as well, over DNS.

Anyway, as we know, Cloudflare's web proxy frontend has switched on ECH support for all of their free tier client sites, which is a gazillion. Now we need the other standalone web servers to catch up and offer their support. An experimental implementation for OpenSSL now exists, so that will help to get Unix and Linux-based servers, which are using the OpenSSL library for their encryption connectivity, onboard and running with it. And in time it seems clear that all the rest will join, since user privacy is dependent upon the site they're connecting to, that is, once their browsers all support ECH, it's dependent upon the site they're connecting to, offering its support for the privacy of all incoming connections. You know, the sites need to publish a public key in their DNS, which the browsers are then able to use to encrypt the first packet to the server. So, you know, of the sites to do this, as soon as the software catches up, we'll be able to do that.

The Pixel 8 and the 8 Pro now offer two additional years of updates. Last Wednesday, Google announced that its newest phones, the Pixel 8 and the Pixel 8 Pro, will now receive seven years of software and security updates, a two-year bump from the previous five-year support that they were providing for the Pixel devices. And everyone listening understands the importance of long-term security updates for consumer devices. Updates really do need to extend throughout the useful life of consumer devices. You know, it doesn't do any good to say, oh, yeah, we'll support your phone for two years because we want you to buy another one.

Well, the phone still works, so you're going to give it to somebody who's going to have an unsupported phone. And smartphones, as we know, are currently among those devices needing the most protection. No other device is more prone to attack than smartphones, and few other devices are as complex or as exposed to the multiple channels of external material that smartphones are. So props to Google for stepping up and going from five years to seven years.

Something known as the MACE Act Passed. What's the MACE Act, you're likely wondering? Well, MACE stands for Nancy Mace, who was coincidentally one of the bill's co-sponsors. Nancy is a Republican representative from South Carolina. But the acronym MACE (M-A-C-E), aside from being Nancy's last name, which I guess is coincidental, stands for "Modernizing the Acquisition of Cybersecurity Experts." Actually she probably took her name and figured out how she could make it...

Leo: Yeah. I think she - unless she changed her name to match the act, which seems unlikely. Yeah.

Steve: That would be going a little too far.

Leo: Yeah.

Steve: So Modernizing the Acquisition of Cybersecurity Experts, MACE. And the U.S. House of representatives, which is presently having a difficult time agreeing on the time

of day, is apparently in wild agreement over this one, this idea, because the Act passed through the House on a vote of, get this, 394 to 1.

Leo: Well, I think it's a good idea, then. Who was the one?

Steve: I guess it must be. It does make me curious, though, who the holdout was.

Leo: Yeah. Probably Joe Manchin. He doesn't like anything.

Steve: Is Tommy Tuberville a senator or a representative?

Leo: He's a senator. Maybe it's Tommy. Good old Tommy probably voted against it.

Steve: Well, but this is in the House, not the Senate.

Leo: Oh, it's in the House, okay. Well, could be anybody in the House.

Steve: Yeah, exactly. The other cosponsor of this bipartisan bill was California Democrat Katie Porter.

Leo: Oh, and she's good. I like her. Yeah.

Steve: Oh, yeah. She's just about as far to the left as Nancy is to the right.

Leo: Mace is to the right, yeah, yeah, yeah. It's perfect.

Steve: But on this they agree.

Leo: It's bipartisan.

Steve: So the MACE Act is aimed at addressing shortages in federal cybersecurity positions by expanding the pool of eligible applicants by lowering the education requirements. Now, at first you might think, whoa, wait a minute, we don't want kindergartners setting up our firewalls. No, no. That's not what this does. Under the legislation, agencies would be allowed to consider an applicant's education only if their education directly reflects the competencies required for the position.

So apparently before this, you know, you had to have a master's degree in social sciences or political science or something, and then also have your cybersecurity certificate. Now, no. Only the education that matters. The bill would also require the Office of Personnel Management to publish annual reports detailing changes to minimum qualifications for cybersecurity positions and data on the education level of people in

those positions now. So anyway, the bill is now headed to the Senate, where it's expected to pass. And that will give Tommy Tuberville an opportunity to say no.

Leo: He can weigh in.

Steve: Yeah.

Leo: It seems like a good idea, I guess; right?

Steve: I think so.

Leo: It's an emergency.

Steve: Well, so the problem is, Leo, there is a serious shortage. I mean, there are, like, we need more cybersecurity people.

Leo: Yeah. Everybody should go to ACI Learning and study so that you can get that job.

Steve: Yes. Even if you didn't go to college, again, they're not going to care.

Leo: It shouldn't matter.

Steve: As long as you've got your certification level, you're good to go.

Leo: That's right. The skills are what count.

Steve: And again, I think that's great. So, dwell time has plummeted.

Leo: Oh, no.

Steve: Who'd have thought that?

Leo: What's that?

Steve: I know. Who wonders what that is? Those who track, monitor, and remediate the consequences of ransomware attacks use the term "dwell time" to reflect the time from the initial network intrusion until the ransomware encryption event is triggered. So those monitoring this dwell time are reporting that attackers are deploying ransomware on breached networks faster than ever before. In just 12 months, that is, the past year, the median dwell time of ransomware groups on hacked networks has fallen from 4.5 days to

less than one day. So which is to say a year ago your network would be breached, 4.5 days would go by until your network was encrypted. No longer. According to the security firm Secureworks, ransomware is now being deployed within one day of initial access in more than half of all engagements and in as few as five hours from initial network penetration in 10% of all cases.

Okay, now, one of the reasons for this is that the percentage - this is the other interesting thing. The percentage of human-driven attacks has risen dramatically. At the time, several years ago, we talked about how bad guys would have a pending inventory of victims that they would be getting around to when they had time, when they were able. I mean, there were just many available networks. This victim pool was coming into them from automated scanning and attack malware that would get into a host network, settle down, and then phone home, logging into its command and control network and saying, okay, boss, what do you want me to do now?

But that was then. These days, the majority of attacks are human driven in real time. It's not that there are fewer vulnerable systems, as we noted at the top of the podcast. There's like more than ever, unfortunately. It's that there's a lot more manpower available because this is where the money is. This stuff is paying off. So now a live human attacker is doing the penetrating. They quickly get the lay of the land, look around, see what's up. And if the network appears to be a useful victim, they will immediately set about exfiltrating all the data that they can in order to use it for blackmail purposes later, then find and prepare to encrypt all of the systems within reach. And as we saw, in 10% of the cases, in as few as five hours of initial penetration. So, you know, you just can't sit around on known vulnerabilities and think, yeah, I'll get to that after lunch.

Leo: I wonder because, you know...

Steve: They may be eating your lunch.

Leo: Dwell time was increasing also because they wanted to wander around, exfiltrate stuff. There were other things they could do.

Steve: Yup. Yup.

Leo: But I'm wondering now if it's shrinking because of better defenses, like if somebody gets in, they think, yeah, we'd better take advantage of this before they catch us.

Steve: Yes, I think - yes, that's a very good point, awareness has certainly skyrocketed, as well.

Leo: A lot of people with canaries in their enterprise; right? You know?

Steve: Yes, yes. And you can imagine that once upon a time the IT guy would hang his head low and walk into the CEO's office and say, "Uh, chief, we've been hit by ransomware."

Leo: Right.

Steve: And a few years ago, the CEO would say, "Ransom what?"

Leo: Uh-huh.

Steve: Now everybody knows what it is.

Leo: Now it's more likely the CEO that's with his hair on fire running to the IT department saying, "Do something."

Steve: Yeah. Exactly. And speaking of doing something, Leo.

Leo: Let's do something.

Steve: It's now your turn to do something.

Leo: Let's do an ad. I can do that. I can do that. Yeah, it's fascinating. I don't know if it's in your story rundown, but MGM...

Steve: Yes.

Leo: Is it in your rundown? We'll talk about it later.

Steve: No, no, no.

Leo: So they were, as we know, Caesars was hit by ransomware.

Steve: Yup.

Leo: Paid probably half the ransom, not the full amount.

Steve: Got themselves back online.

Leo: And got online. And MGM, they're saying they declined to pay. And they suffered from at least a week of disruptions.

Steve: And they're now saying it's going to cost them a total of $100 million.

Leo: Hundred million dollars.

Steve: For all the remediation. And that 10 million of that was just, you know, hiring consultants and figuring out was going on and like dealing with it. So, yeah.

Leo: But good for them, I guess, for not paying because that's what the FBI wants you to do. Because if you pay them it just encourages them. Yeah. I don't know. It's probably cheaper to pay.

Steve: And Caesars did pay, and unfortunately it's yes.

Leo: And they went to MGM.

Steve: There was a song once that I think the title was "It's Cheaper to Keep Her."

Leo: Okay. Just a little tip, don't sing that with Lorrie around. I'm just saying. Just saying.

Steve: So David Sherman, a listener, asked via Twitter: "Which is better, Passkeys via Bitwarden or Passkeys via browser?"

Leo: Or SQRL.

Steve: Well, there's a thorn. "How to sync passkeys among different browsers? Thanks."

Leo: Passkeys - by the way, could I just say this? We've been talking about this on the shows, not just this show - has really not turned out to be all things.

Steve: No. I agree. I think the best answer to his question is that Passkey management is still too new and too much in flux for any answer to necessarily hold for long. But there's an essential point I don't think I've made clear enough in the past. Sadly, Passkey management has been shrouded in hocus pocus because the purveyors of Passkeys have, from all appearances, desperately hoped to be able to use the mysticism surrounding this promising new technology to retain and corral their users into their own proprietary environments. The universally missing feature of simple Passkey export and import is astonishing to me.

We all know what a password is. We make them up, and we use password managers or our browsers to remember and regurgitate them on demand. But no one really knows what a Passkey is because no one has ever actually seen one. It's all just mysterious "Don't worry, we got this" hocus pocus behind the scenes, and we're told that it's all super wonderful and hacker proof and so forth. Okay. So here it is. A Passkey is nothing more than a private key. Period. And as such it's not some impossible-to-represent mystical token. It's just a relatively short blob of binary data. So it could easily be turned into a QR code, or into a short and manageable Base64 encoded string of text. And at that point it could be moved from place to place, printed out and stored somewhere for

safety. All of a user's current Passkeys could be exported as a simple CSV file for safekeeping.

But for some bizarre reason, end users are not allowed any of those freedoms today. Expert users want it, and everyone could have it. But no. However, I don't expect this mystical barrier to survive in the long term because it is trivial to export and import Passkeys for backup and cross-platform sharing. So someone will be the first to create a simple QR code, textual, or CSV file Passkey export and import. Then everyone will want it, and it will become a required feature for any Passkey system. But unfortunately we're certainly not there today because the system is still so new and Passkey support hasn't yet become the commodity that someday it presumably will. And Leo, I agree with you. I don't have a single one. I mean, not even one. It's just like...

Leo: Yeah. First of all, very few sites are using it. I created one, you may remember I was on the air with you doing it for Best Buy. And I only then found out, well, it's only on my iPhone. It's not - and this is - and I think this is vendors wanting you to lock in because, I mean, I have a Passkey for Best Buy. Why can't I use it on my Linux machine? Why can't I use it on this Windows machine? Well, you can't. It's inside your iPhone.

Steve: Yup. Yup. And as I said, there's nothing to prevent your iPhone from putting up a QR code, and you show it to your Linux webcam.

Leo: And then it would have it.

Steve: And now the Passkey is there.

Leo: And this was what you did with SQRL. You built this portability in. But, see, you didn't have the same economic incentives that Apple and all these other companies have.

Steve: Right. Right.

Leo: They want to keep you on their platform.

Steve: Right. And unfortunately what they've done is killing this thing.

Leo: Yeah. I think so.

Steve: I mean, basically nobody, it's like, I don't want this. I know what a password is. I don't know what a Passkey is. Just show them. I mean, again, it's like, it's not some, like, mystery. It's a little bit of binary. And you can put it in a QR code, print it out as text, I mean, and then you'd have way more flexibility and freedom. And you'd lose none of the power. It still uses public key crypto. The site sends your client a challenge, which you have to sign using your private key, so it's not subject to replay attacks. I mean, all the other good things that SQRL and Passkeys both have stays available. I think, you know, your argument is, well, users cannot be trusted with this magic stuff. So we're

going to hide it from them and do the "just trust us." But that just doesn't work. And as a consequence, this thing's sort of DOA, unfortunately.

Leo: Yeah. Yeah. I mean, when I go to Google, I can log in with my Passkey. And then it puts up a QR code at Google.com. And then I get it, I have to just aim my phone at it, and then the phone says okay.

Steve: And are you you.

Leo: And I'm doing the Face ID. And now I'm logging into Google. And so it kind of works.

Steve: Oh, yeah, I mean, it works. But, like, who wants it?

Leo: That's the problem. It's so much easier, especially if you're using a password manager, it's so much easier.

Steve: Yeah.

Leo: All right.

Steve: Henrik Lexow. He offers us a view from inside an ISP. He said: "Hi, Steve. After your deep dive into encrypting the ClientHello," and I should preface this by saying this is a little chilling. "After your deep dive into encrypting the ClientHello, an old internal dilemma has resurfaced. I've spent years," he said, "at a European ISP, where ideas about selling content-filtered access control or insights would occasionally come up. Some of the veteran business folks, who had experienced the unencrypted Internet era, demanded similar services here. I had to explain that HTTPS Everywhere was happening, but we often circled back to the possibility of inspecting TLS packets for user insights."

Leo: Oh, that's one way of putting it. User insights, eh?

Steve: User insights. That's what we're going to monetize. He said: "This usually boiled down to two 'customer value areas,' child content protection and enterprise content filtering, with the underlying goal of monetizing data from blanket Internet use inspection." He said: "(We eventually did not offer these services during my tenure.)" He said: "My ongoing struggle revolves around balancing parental concerns for children's online safety with the goal of Internet privacy. ECH, despite its complexity, is a step in the right direction, as I'm aware of the ISP efforts to tap into this revenue stream while publicly championing privacy. ECH can put an end to this hypocrisy. But, as is often asked in privacy, 'What about the children?'

"My question to you, Steve, relates to those two 'customer value areas' in the context of ECH becoming a reality. First, who should take responsibility for child protection services now that ECH is coming? Not all children have parents who can actively monitor their online experiences. And how could it be done correctly? And second, are there valid reasons for enterprises to engage in content filtering, be it for security or other

purposes? I personally struggle to see the value in this. In the case of spending time on non-work activities on the Internet, it seems more like a cultural issue within the company rather than an Internet access problem. Best regards, Henrik."

Okay. So in answer to his first question, "Who should take responsibility for children protection in a world where TLS packet filtering will finally be thwarted," the first thought that comes to mind is local DNS service filtering. I think that's the best solution by far. If a family's home is using, for example, the free OpenDNS Family Shield Service, which is as easy as configuring the residential router to use a specific pair of IP addresses - for anyone who's curious, I have them in the show notes, 208.67.222.123 and .220.123 - then there's no need to see into any aspect of the TLS connections once they've left the family's router since unwanted domains will never have their IP address resolved in the first place.

And Henrik's second question: "Are there valid reasons for enterprises to engage in content filtering, be it for security or other purposes?" I doubt that the addition of Encrypted ClientHello will change the enterprise environment much. As we've talked about in the past, an enterprise's network is owned by the enterprise, and all of its employees should be informed and aware and kept aware that the content of the enterprise's network is not private and that nothing they do over that network should be considered private. We've talked about placing a strip of paper to constantly remind everyone of that fact along the top of every company-owned computer monitor, and of having the Human Resources Department remind every employee of that fact every year during their annual review.

So that's the enterprise's policy view. The practical implementation of such technology within the enterprise is already well established. All non-proxied TLS connections will be blocked from ever leaving the enterprise's network, and any system wishing to connect to the external Internet will need to have the enterprise's proxy server's root certificate added into its root store. When that's been done, all outbound TLS connections will be intercepted and accepted by the enterprise proxy middlebox which will, in turn, connect on behalf of each user to the remote resource while being able to fully examine in detail and filter the contents, not only of where they're connecting to, but everything that goes on during that connection. So for the enterprise, the addition of Encrypted ClientHello won't change anything. The enterprise will need to update their middlebox's firmware to add support for ECH, but otherwise life goes on as it has before.

Leo: That's great to hear it from inside an ISP. And those silly rationalizations that they come up with. Oh, it's for the kids.

Steve: Yes, yes. Can't we, like, somehow make money from knowing what our clients are doing? You know?

Leo: Yeah, that's really it.

Steve: Is there some way we can do that?

Leo: Yeah, that's it.

Steve: Yeah, that's it. Matthew Cowgur, he tweeted: "Does having a very good password," he says, "mine is about 140 bits of entropy, have any effect on encryption

with a low number of iterations?" Now, it's interesting that this is not a question that we've directly addressed during our focus upon the question of iteration count. We've exclusively focused upon iteration count as the way to slow down any brute force password guessing attack. But the only reason such an attack needs to be slowed down is because we're assuming that the user may not have chosen a super-strong password in the first place. That means that given a sufficient number of guesses, the attacker will eventually obtain the not-very-strong password just by trying a whole bunch.

Wikipedia reminds us that in one study of half a million users, the average password entropy was estimated to be about 40, four zero, .54 bits. So a little over 40.5 bits of entropy. And the medium to strong password threshold is regarded as around 50 bits of entropy. If you've got 50 bits, that's considered to be the low end of a strong password. So 50 bits is 9.5 bits more than the study quoted as being typical.

So thanks to the power of powers of two, and we're using powers of two because they're binary bits, they can have two states, raising that, raising two to the power of 9.5, which is to say the number of more entropy bits for a strong password versus the average, so two raised to the 9.5 is 724. That tells us that the difference in brute force resistance between that study-average password strength and one that's on the lower border of being strong is 724 times the strength, going from 40.5 bits of entropy to 50.

So now let's look at Matthew's question. Matthew claims to be using what he calls a "very good password," having around 140 bits of entropy. Now that we've created some context, it should be clear that, if Matthew is correct, and his password truly contains around 140 bits of entropy, it's not a very good password, it's an insanely good password.

Leo: Ooh, you scared me. Good.

Steve: Insane.

Leo: It's great.

Steve: If 50 bits of entropy is the lower bound of a good password, then Matthew has added 90 bits to that 50 bits to get up to his 140. Once again, turning to the power of powers of two, 90 additional bits of entropy results in a password that is 1.238 times 10^27 stronger. Okay, now, to help us understand the size of that number, 27, that's the number of zeroes, that's 27 is three times nine. And nine zeros is a billion, which we have that three times. In other words, Matthew's 140-bit entropy password is 1.238 billion billion billion times stronger than a strong password.

Leo: Oh, that seems pretty good. That seems good.

Steve: And a password we'd considered to be [crosstalk].

Leo: I'll take it, yeah.

Steve: Yeah, wow. So Matthew's question was: "Does having a very good password, mine," he says, "is about 140 bits of entropy, have any effect on encryption with a low

number of iterations?" And the answer is a resounding hell, yes. If you have such a password, you would be fine with zero iterations.

Leo: Aha.

Steve: Since the only reason we iterate is to protect weak passwords. If you really have a strong password, and I'm talking to everybody now, you have nothing to worry about. When we changed the iteration count, back in the LastPass days, from 5,000 to 100,100, or now to 600,000, we've increased the attacking difficulty only by a factor of 120. But just adding seven binary bits of entropy, seven binary bits is 128 times because that's 2^7, 128. So a password having just seven additional bits of entropy would provide more cracking resistance, 128 times more, than jumping the iteration count from 5,000 to 600,000, which only increases it by 120 times.

So the key takeaway here is that increasing iteration counts is a linear increase, whereas adding bits of entropy is exponential. Every bit, every single bit of entropy added doubles the cracking difficulty. So when you've doubled it seven times, that's 128 times stronger that you've made it. Therefore anyone who has a really strong high-entropy password, even back then when their vaults were stolen from LastPass, even if it's not as ridiculously strong as Matthew's, has really nothing to worry about regarding PBKDF iterations. Increasing iteration counts is far weaker protection than using a truly strong password. In which case it doesn't even matter how many times you iterate. It could be zero.

Leo: So really these key derivative functions, whether it's PBKDF2 or Scrypt or Argon2, they're really kind of belt-and-suspenders to protect you if you don't have a great password.

Steve: Yes. They linearly slow down the attack. But adding bits of entropy exponentially slows down the attack.

Leo: Yes. So have a good, long, strong, truly random password, the kind the password manager generates for you.

Steve: Yup.

Leo: I guess you probably shouldn't keep - well, you could; right? You could keep your vault password in your password manager. It just wouldn't be any use if you didn't have an access to it. Okay.

Steve: Okay. So a listener said, "Hi, Steve. According to this TorrentFreak article, Cloudflare has enabled Encrypted ClientHello for all customers on free plans, which includes many pirate sites. The new privacy feature makes it impossible for Internet providers to track which websites subscribers visit. As a result, it also renders pirate site-blocking efforts useless, if both the site and the visitor have ECH enabled." And he provided the link to this TorrentFreak.com page.

So, okay. This is inevitable, and it's analogous to the encryption debate; right? We want to enhance privacy, but we're unable to enhance only the good guy's privacy. Everyone

gets their privacy enhanced, even those who will criminally abuse that privacy. The question the world has been struggling with for the last few years is whether our inability to restrict who gets more privacy means that no one should have any more. But it's looking like the world is going to agree that giving more to everyone is the best solution.

The TorrentFreak article that our listener linked to was interesting, and it shed a different light onto the emerging presence of Encrypted ClientHello connections. I've trimmed it down a bit to remove the stuff we already know, but here's what TorrentFreak observed. They said: "Cloudflare has enabled Encrypted ClientHello for all customers on free plans, which includes many pirate sites. The new privacy feature makes it impossible for Internet providers to track which websites subscribers visit. As a result, it also renders pirate site-blocking efforts useless, if both the site and the visitor have ECH enabled.

"Website blocking has become the go-to anti-piracy measure for the entertainment industries when tackling pirate sites on the Internet. The practice has been around for well over 15 years and has gradually expanded to more than 40 countries around the world. The actual blocking is done by Internet providers, often following a court order. These measures can range from simple DNS blocks to more elaborate schemes involving Server Name Indication (SNI) eavesdropping, or a combination of both. Thus far, the more thorough blocking efforts have worked relatively well. However, as privacy concerns grew, new interfering technologies have emerged. Encrypted DNS and SNI, for example, made blocking efforts much harder, though not impossible.

"A few days ago, Internet infrastructure company Cloudflare implemented widespread support for Encrypted ClientHello (ECH), a privacy technology that aims to render web traffic surveillance futile. This means that site blocking implemented by ISPs will be rendered useless in most, if not all cases. ECH is a newly proposed privacy standard that's been in the making for a few years. The goal is to increase privacy for Internet users; and it has already gained support from Chrome, Firefox, Edge, and other browsers. Users can enable it in the settings, which may still be experimental in some cases.

"The main barrier to widespread adoption is that this privacy technology requires support from both ends. Websites have to support it, as well. Cloudflare has made a huge leap forward on that front by enabling it by default on all free plans, which currently serve millions of sites. Other subscribers can apply to have it enabled. The push for increased privacy is well-intended, but for rights holders it represents a major drawback, too. When correctly configured, ECH defeats site-blocking efforts. Tests conducted by TorrentFreak show that ISP blocking measures in the UK, the Netherlands, and Spain were rendered ineffective.

"This doesn't automatically apply to all blocked sites, as the sites must have ECH enabled, too. We've seen mixed results for the Pirate Bay, perhaps because it has a paid Cloudflare plan, but other pirate sites are easily unblocked. This new privacy feature hasn't gone unnoticed by pirate site operators. The people behind the Spanish torrent site DonTorrent, which had dozens of domains blocked locally, are encouraging users to try ECH." Yeah, no kidding.

"DonTorrent notes: 'Before ECH, your online privacy was like a secret whispered in the wind, easily picked up by prying ears. But now, with ECH by your side, your data is like hidden treasure on a remote island, inaccessible to anyone trying to get there without the right key. This feature encrypts your data so that neither ISPs nor organizations like ACE and MPA can censor, persecute, and intimidate websites they consider illegal.'

"Cloudflare and other tech companies," writes TorrentFreak, "are not supporting ECH to make site-blocking efforts obsolete. However, this privacy progress likely won't be welcomed by rights holders, who've repeatedly criticized Cloudflare for hiding the hosting

locations of pirate sites. TorrentFreak reached out to a major anti-piracy organization for a comment on these new developments, but we have yet to receive an on-the-record response. It wouldn't be unthinkable, however, that we will see more blocking lawsuits against Cloudflare in the future."

So, and of course we touched upon briefly the content-filtering thwarting aspect of ECH last week, but we didn't explore the real-world consequence to those, like the Motion Pictures Association, who have been using legal means to force the blocking of sites containing pirated copyrighted content. Looks like that ability to do so is going to be short-lived at this point. As I said at the top of this, everyone getting their privacy increased means that the bad guys do, too. And the world has decided that's the way we're going to go. Which I think everyone agrees is the right direction.

Skynet tweeted: "This ECH stuff is going to mess with my public WiFi content filtering; isn't it."

Leo: Oh, yeah.

Steve: Yeah. He said: "I just got my vendor to figure out what was blocking patrons' eBooks from being downloaded onto their Kindles, and now this. Thanks, Steve. Thanks a lot." Anyway, so...

Leo: Think of the authors.

Steve: That's right. Think of the authors. This was, we all know, not my doing. I'm just reporting the facts <grin>. But as for it messing with someone's public WiFi content filtering, uh, yep, it's going to do that. I presume that ECH is somehow protecting itself from protocol downgrade attacks. We've talked a lot about them in the past because they can be tricky to prevent. The original downgrades were HTTPS to HTTP. On an initial HTTP connection that was attempting to switch itself to HTTPS, since the connection was not yet encrypted under HTTP, something like a script running would simply change all the HTTPS URLs into HTTP, keeping the connection and all of its components unencrypted, and leading each end to assume that the other end had a problem with encryption when that was not true.

So downgrade attacks have always been a problem. In the case of ECH, for example, as we know, it only works when both ends support it. And at this early point in time, support is probably more surprising than not. That is, it's like, hey, I can get ECH? Great. Doesn't happen that often. So if, for example, an outgoing or incoming initial TLS handshake packet were to be tweaked to show that ECH was not supported by that packet, when in fact it was, the other end would shrug and not be surprised. Meanwhile, the domain name would be exposed to anyone watching the traffic because it wouldn't have the advantage of Encrypted ClientHello encryption.

Hopefully - and I don't know. I did a little bit of poking around. I couldn't come up with an answer quickly. Hopefully ECH's designers were aware of this problem and did come up with some means of preventing middlemen from removing ECH support on the fly from connections before ECH has a chance to get started. Otherwise, its attempted presence would just be a temporary inconvenience. Anyway, in the case of our listener Skynet's public WiFi, any ECH-using clients would also be using DNS over HTTPS, and thus not his local access point's DNS. So unfortunately, filtering DNS wouldn't work, either. I mean, it actually is going to present a problem to local WiFi content filtering. That's true.

For this next question, I anonymized it myself. He didn't ask me to, but I feel as though this person's tweet should be anonymous. He said: "Hey, Steve. You mentioned in a recent episode that the Linux kernel has fixed the epoch time issue in kernel build 5.10. I feel like that might have put too many people at ease. Not only are old devices still running out-of-date kernels, but modern stuff does, too.

"As I have mentioned before, I work for Check Point, a company with firewalls protecting a massive portion of the Internet. We serve millions of businesses, including all of the Fortune 500 companies, which account for decent chunks of the packets moving across the Internet, all having to pass through one of our firewalls. After many years of fighting with R&D, Check Point finally upgraded its OS to move away from Linux Kernel 2.6.18 to 3.10 as of 2020. This is the latest anyone can run, 3.10, still with the epoch code issues.

"I shudder at the thought that we still have customers running code that has passed end of life over a decade ago. Even if R&D begins working on a migration to kernel 5.10 today, it would take years to finish, and decades to move everyone off the older releases. All of that on current, modern, state-of-the-art key infrastructure that the world relies on. I think people should still expect a huge mess at the end of epoch time. Thanks for the wonderful show. Looking forward to more episodes to 999 and beyond, and not having to use Twitter anymore." Well, maybe someday soon. "All the best."

So anyway, that note speaks for itself, and I thought it was a valuable look inside the reality of the commercial use of Linux-based appliances. There really is an "If it's not broken, let's not break it" mentality. And really, after Check Point built a robust firewall architecture on top of an old OS kernel, if it's working, why change it? If it's primarily being used to boot one's own code, and important things like the OpenSSL library can be kept current, then why mess with the boot loader? You know? Which is essentially what Unix or Linux has become there. The problem is, though, the earlier Linux file system timestamps are all using the signed 32-bit time, so any reliance upon file times is going to go berserk in 2038. On the other hand, that's still 15 years off. So we're going to be okay.

Leo: I guess. I think we'll still be doing this.

Steve: And Leo, you may be driving, we may both be driving new cars by then.

Leo: Maybe, yeah.

Steve: We'll probably be driving our last cars by then.

Leo: Fifteen years from now you and I, I don't know if we'll be driving, but we'll see.

Steve: Ah. Hopefully.

Leo: It's coming pretty quick. Hate to say it. And I hope my car is not running a Linux kernel pre-3.10. That's one thing, for sure.

Steve: Matt, tweeting as @Slater450413, said: "Hey, Steve. There's something that still bothers me about that recent Microsoft hack." He's talking about the capture of the key

in the crash dump. He said: "There were far too many coincidences where the attacker just 'happened to know' various flaws." Okay, now, so just to interject, Matt's referring to the fact that not only was the secret key resident in RAM and then captured by a system crash dump, but then there was a chain of no fewer than every single one of five flaws were required for that crash dump image to migrate itself all the way to where it was able to be reached by an attacker.

So anyway, Matt continues, writing: "Unlike regular flaws in a desktop OS where you can continually poke at it with a debugger attached to the OS, most of the surface of a cloud instance being attacked also happens to be blind to outcome observation, like the modern version of blind SQL injection. Yet pinpoint accuracy was achieved regardless of that, multiple times. It's almost as if the attacker had source code to examine, sort of like the access and download that was achieved during the SolarWinds attack two years ago."

He says: "I don't like conspiracy theories, and I'm aware we will never know, but this seems far more likely than an attacker guessing their way through this much obscure and highly technical knowledge. I remember back on Security Now! Episode 800 there was a passing comment Microsoft made publicly that they do not rely on source code being kept secret as a security feature. But I wonder if this accidental visibility may now be coming home to roost." And it is a good point. As Matt says, we'll never know. But it was, if nothing more, a surprisingly glaring sequence of successive failures all required in sequence that brought that key into the hands of someone who then also knew what to do about it.

Leo: My memory is not complete, but it seemed like it wasn't necessarily triggered by the bad guy.

Steve: No.

Leo: In other words, there was a key. It was in RAM. Due to a flaw in Microsoft, it was saved as a dump. The bad guy didn't trigger that.

Steve: No.

Leo: Then further mistakes were made kind of things where it was [crosstalk] public network.

Steve: Multiple stages where they had code in place that was supposed to scrub their crash dumps for keys.

Leo: But it didn't work. [Crosstalk] case Microsoft had [indiscernible]. But I don't think this was triggered by the bad guy. I think it was a crime of opportunity where that dump kind of got its way into the public, and then the bad guys know, one thing they do know, you look at crash dumps for private keys, and that's when they found it.

Steve: I think you're making a very good point, yeah.

Leo: It was really just bad coding on Microsoft's part. It didn't need to be discovered by anybody. It just happened.

Steve: Brings bad karma. Brought a whole new meaning to bad karma.

Leo: Yeah, yeah.

Steve: I do have one piece of errata, and then we're going to take our last break and then talk about, begin to talk about the Top 10 Cybersecurity Misconfigurations.

Leo: Oh, I can't wait. That's a great subject.

Steve: Oh, it is. Last week I mis-numbered the shortcut of the week 842 instead of 942.

Leo: Doh.

Steve: So, yeah, technically that would have made it a shortcut of the week from about two years ago, which was not what I intended. Elaine, who transcribed the podcast, of course, and several of our listeners picked up on my mistake. That shortcut, just to remind everyone, was to the very nice YouTube video explaining how to set up Syncthing. So for anyone who may have thought, "Hey, that's great, Steve created a shortcut for this week's podcast 942," and then found that it didn't work, okay, the link I created was grc.sc/842. Sorry for the confusion.

Leo: Can't make one at 942 now?

Steve: No. I figured I'd just tell everybody where it went.

Leo: You could probably google it, too, "Syncthing tutorial," something like that.

Steve: Well, yeah. When I was setting up this week's properly numbered shortcut of the week, I noticed that 555 of our listeners took me literally and followed the 842 link to learn more about Syncthing.

Leo: Good. They're smart.

Steve: So I'm glad for that. Or they're used to me, like Steve really means, oh, let's guess. Let's roll the dice.

Leo: Take a chance. All right, Steve. Let's take a break, and then I am fascinated to learn...

Steve: This is very meaty, very good piece.

Leo: Top 10. We should do like a Letterman countdown from 10 to 1, cybersecurity misconfigurations. Maybe we can get Tom Selleck or somebody in to read those.

Steve: Last Thursday, the U.S. National Security Agency, our NSA, and the awkwardly named Cybersecurity and Infrastructure Security Agency, our CISA, jointly published a cybersecurity advisory. Of course, everything has initials, so it's the "CSA," right, the Cybersecurity Advisory, the CSA. This advisory was the result of NSA and CISA red team and blue team activities, as well as the activities of both agencies' Hunt and Incident Response - that's of course HIR - teams. The advisory identifies and highlights the most common cybersecurity misconfigurations which they continually uncover within organizations. And the report details the tactics, techniques, and procedures - which of course is TTPs - which the bad guys use to exploit these misconfigurations.

Checklists, I think, such as these, and obviously you agree, Leo, can be very useful because over and over and over in this podcast we encounter the many consequences of the tyranny of the default. You add multifactor to your authentication. Great. Good move. But did you carefully read through all of the cautions that its publisher included? Has this new facility been fully configured correctly? Or did it take longer than you expected to get it going, and so now you're late for a meeting and had to run off leaving it - probably forever - the way it came out of the box?

Reading through this advisory, I feel as though this podcast has been serving its listeners well, since nothing here will actually surprise anyone who's been listening for long. These are the topics that we often focus on. In fact, number one on the list is "Default configurations of software and applications."

Okay. So today we're going to hit the high points on this advisory. But this work really drills down into very useful and actionable specifics. In fact, there's so much here, this is really only going to be part one of talking about this because there's so much. And their intention really was to get specific, not just to kind of produce a Top 10 list of - and you've got to figure out what that means. But they dig in. So that makes this thing too long, even for us to cover in two parts at the level of detail here that it offers.

But every listener who's responsible for their enterprise's network security really would do well to spend some time with it on their own. So it's available as both a web page and as a very nicely formatted 44-page - like I said, it goes into depth - 44-page PDF. I've made the PDF edition this week's carefully-numbered GRC shortcut of the week. So grc.sc/943 will redirect you to the PDF which resides at defense.gov. And I also have the link to the original web page up at the top of the topic of the show.

Okay, so without further ado, I'm going to quickly enumerate these Top 10 most common and most troublesome cybersecurity network-related misconfiguration issues. Then we'll begin digging into each one a bit deeper. So Top 10 are default configurations of software and applications. No surprise there. Improper separation of user/admin privileges. Yup. And one of our sponsors of TWiT will love this one: Insufficient internal network monitoring. That's clearly a biggie. Number four, lack of network segmentation. How many times have we talked about that, especially in a residential setting with untrustworthy IoT things. Number five, poor patch management. In other words, not keeping up with updates, or not prioritizing updates.

Number six, bypass of system access controls. That'll be interesting to see what they have to say about that one. Weak or misconfigured multifactor authentication methods. Again, okay, not quite clear what they mean there. Insufficient access control lists on

network shares and services. Okay. Now there we're talking about it works for everyone if we don't make the ACLs too tight. So let's just leave them the way they are. Bad idea. Number nine, poor credential hygiene. And number 10, unrestricted code execution. Again, that's one of those, well, if we start restricting things, things are going to break. So that's not good. Unfortunately, it can be very useful for the bad guys.

So the NSA and CISA elaborate, just before they get into the details on those Top 10. They explain: "These misconfigurations illustrate, first, a trend of systemic weaknesses in many large organizations, including those with otherwise mature cyber postures; and, two, the importance of software manufacturers embracing secure-by-design principles to reduce the burden on network defenders." In other words, they're saying both ends of this are at fault. And I think that's exactly right.

They said: "Properly trained, staffed, and funded network security teams can implement the known mitigations for these weaknesses." Meaning it is possible to secure things. And secondly: "Software manufacturers must reduce the prevalence of these misconfigurations, thus strengthening the security posture for all customers by incorporating secure-by-design and default principles and tactics into their software development practices."

They wrote: "NSA and CISA encourage network defenders to implement the recommendations found within the Mitigations section of this advisory, including the following, to reduce the risk of malicious actors exploiting the identified misconfigurations. So remove default credentials and harden configurations. Disable unused services and implement access controls. Update regularly and automate patching, prioritizing patching of known exploited vulnerabilities." Of course that's a big CISA issue. "Reduce, restrict, audit, and monitor administrative accounts and privileges."

They said: "NSA and CISA urge software manufacturers to take ownership of improving security outcomes of their customers by embracing secure-by-design and default tactics, including embedding security controls into product architecture from the start of development and throughout the entire software development lifecycle. Eliminating default passwords. Providing high-quality audit logs to customers at no extra charge." And we have seen a case where you had to pay more for the logs, and that didn't work very well. And finally: "Mandating multifactor authentication," that is, the software manufacturers should be mandating multifactor authentication. They said: "Ideally phishing-resistant for privileged users and making MFA a default rather than an opt-in feature."

In other words, both parties is what we're seeing here right off the bat. The software manufacturer and the software user have responsibilities. A perfect example that they mention is default passwords. The users of any system MUST change the default passwords when they're first setting up their software, but the creators of that software should also absolutely come up with some way to avoid ever having a default password in the first place. In other words, you know, yes, the people using it should know to change the default. But there shouldn't be a default for them to have to know to change.

So just how serious is this simple-seeming problem of default credentials? Actually, it's a bit shocking when you look at how many different ways these "defaults" can be abused. The document explains a few. They wrote: "Many software manufacturers release commercial off-the-shelf network devices which provide user access via applications or web portals, containing predefined default credentials for of course the built-in administrative accounts. Malicious actors and assessment teams regularly abuse default credentials." So again, this is coming from real-life experience with red team and blue team work and post-incidence response.

They're finding: "Malicious actors and assessment teams regularly abuse default credentials by finding credentials with a simple web search and using them to gain authenticated access to a device; resetting built-in administrative accounts via predictable forgotten passwords questions; leveraging default virtual private network credentials for internal network access; leveraging publicly available setup information to identify built-in administrative credentials for web applications and gaining access to the application and its underlying database; and leveraging default credentials on software deployment tools for code execution and lateral movement."

They said: "In addition to devices that provide network access, printers, scanners, security cameras, conference room AV equipment, Voice over Internet Protocol phones, and Internet of Things devices commonly contain default credentials that can be used for easy unauthorized access to these devices, as well. Further compounding this problem, printers and scanners may contain privileged domain accounts loaded so that users can easily scan documents and upload them to a shared drive, or email them. Malicious actors who gain access to a printer or scanner using default credentials can use the loaded privileged domain accounts to move laterally from the device to compromise the entire domain."

So, okay. My feeling is that the awareness of the danger posed by dangerous defaults of any kind has been very well known for decades. So at this point, any manufacturer who's still shipping products with dangerous default settings which they expect their customers to know to change, and frankly which must be changed in order to have any security, such a manufacturer at this point is beyond lazy. Many, if not most users, even obviously at the enterprise level, presume that the way things come from the factory are intended to be the way they should be, thus the tyranny of the default takes hold. But, you know, if this was not the case, if it was the case that defaults were secure, we wouldn't have the tyranny of the default, it would be the blessing of the default because these things would be ready to go, secure out of the box.

And, you know, there is, I think we know, the occasional sighting of a manufacturer who gets it, who requires their user to invent their own admin password right off the bat, you can't go any further until you do that, during the initial setup and configuration of the device. Then it'll make you log off and log back on using it to prove that you're able to, and then you'll be able to move forward, but not until then. And, you know, even sometimes that's annoying when you're in a hurry and just want to get something, a test setup, up and going and running.

But you've run across a device which was - whose design was done correctly. They've got no defaults, so they're inherently far more secure. But again, even today, even though we know how to do that, everyone knows how to do that, these sightings are still the exception rather than the rule. So the responsibility is still resting upon the rest of us, you know, those users who use these things.

The document notes also some interesting specifics. They said: "Certain services may have overly permissive access controls or vulnerable configurations by default. Additionally, even if the providers do not enable these services by default, malicious actors can easily abuse these services if users or administrators enable them." Or, I would argue, leave them enabled. Again, not secure by default. "Assessment teams regularly find the following: insecure active directory certificate services; insecure legacy protocols and services; insecure server message block (SMB) service."

Looking more closely at legacy protocol and services and insecure server message block services, they note: "Many vulnerable network services are enabled by default, and assessment teams have observed them enabled in production environments. Specifically, assessment teams have observed Link-Local Multicast Name Resolution (LLMNR) and NetBIOS Name Service (NBT-NS), which are Microsoft Windows components that serve

as alternate methods of host identification. If these services are enabled in a network, actors can use spoofing, poisoning, and relay techniques to obtain domain hashes, system access, and potential administrative system sessions.

"Malicious actors frequently exploit these protocols to compromise entire Windows environments." This is what's happening during that dwell time after someone gets in while they're busy digging deeper into the network. They said: "Malicious actors can spoof an authoritative source for name resolution on a target network by responding to passing traffic, effectively poisoning the service so that target computers will communicate with an actor-controlled system instead of the intended one. If the requested system requires identification/ authentication, the target computer will send the user's username and hash to the actor-controlled system. The actors then collect the hash and crack it offline to obtain the plaintext password.

"Server Message Block service is a Windows component primarily for file sharing. Its default condition, including in the latest version of Windows, does not require signing network messages to ensure authenticity and integrity." And we've touched on this not that long ago in the podcast. They said: "If SMB servers do not enforce SMB signing" - which again is not required by default because oh my god, oh heavens, it might break something, we'd have to go find out what and then fix it. They said: "Malicious actors can combine a lack of SMB signing with the name resolution poisoning issues above to gain access to remote systems without needing to capture and crack any hashes at all."

So as I'm reading that, another thing occurs to me. There's an aspect of asymmetrical warfare that applies here. These systems have grown to be insanely complex over time, and they're dragging along a growing encrustation of legacy protocol crap so that nothing from the past ever breaks, and everything that someone might have continues to work. Even if no one has anything, you know, if they plug it in, oh, it needs to work. And this is true, you know, even if the organization themselves doesn't have any of that stuff. It's all still there because it was on by default to make sure everything just works out of the box. But it is also horribly insecure.

So the beleaguered IT professional who just wants things to work doesn't mess with those things. Again, assuming that, if it came that way, it's supposed to be on. Sure, he or she also wants them to be secure. But first they have to work. But the bad guys have an entirely different agenda. And I understand that this is obvious, but I think it's still critically important. The bad guys are living off of this debris, off of all of this, you know, "What if maybe we'll need this someday" legacy stuff.

They've learned and know the ins and outs of how to abuse these retired or retiring systems that persist. And this is the asymmetric aspect. As we know, security is all about the weakest link. So it literally does no good to have super security on the latest spiffy new network layers if the ancient networking protocols are still left lying around, active and enabled. The enterprise may not be using them, but that doesn't mean the bad guys will not be abusing them.

Okay. So we're at page 18 in the show notes, which usually means we're out of time. And as I look at the clock, we're also at two hours. But there is still so much really good meat here to discuss: segmentation of user and admin privileges; lack of network monitoring and lack of network segmentation; poor patch management; bypassing of access controls; weak or misconfigured multifactor authentication; and more. So next week I plan to continue digging into some of the remaining high points of this very important document. So stay tuned.

Leo: Nice. It is, it's fascinating. And at some point the CISA people wrote us and said they wanted to get on the show. We should talk about the - maybe they wanted to talk about this. I don't know. We could ask them about it.

Steve: That's good. That's a good point.

Leo: Yeah, yeah.

Steve: We will do a show with some CISA guys.

Leo: I think it's a good idea, yeah. But really all we need is one guy right here, this guy, Steve Gibson.

Steve: Bye.