DNS Consolidation

Description: This week we look at a couple of new and forthcoming Chrome features. I'll quickly run though some new and notable ransomware casualties, including a couple of follow-ups. We'll look at a critical flaw in the Drupal content management system, the big trouble with generic smart doorbells, an interesting attack on Tesla Model X key fobs, CA's adaptation to single-year browser certs, several instances of leaked credential archives, a critical RCE in a major MDM server, a bit about the Salvation Trilogy, and some extremely promising news about SpinRite's future. Then we'll wrap up by taking a look at the consequences of the increasing consolidation of DNS service providers. It's not good if staying on the Internet is important to you.

High quality (64 kbps) mp3 audio file URL: http://media.GRC.com/sn/SN-795.mp3
Quarter size (16 kbps) mp3 audio file URL: http://media.GRC.com/sn/sn-795-lq.mp3

SHOW TEASE: It's time for Security Now! with Steve Gibson. We've got Drupal issues - yes, again. We've got key fob hacks - yes, again. Also we'll talk a little bit about why you should probably not buy that cheap knockoff video doorbell. It's all coming up next on Security Now!.

Leo Laporte: This is Security Now! with Steve Gibson, Episode 795, recorded Tuesday, December 1st, 2020: DNS Consolidation.

It's time for Security Now!, the show that covers your privacy and security online. It does a little teaching, too. And have some fun while we're at it. And that's the guy in charge, right over my shoulder here, Mr. Steve Gibson of the GRC.com site.

Steve Gibson: I suppose that depends upon your definition of "fun." But, yeah.

Leo: Oh, well, if you listen to this show, you know what I mean.

Steve: Yeah.

Leo: I mean, if you think it's fun to write in assembly language, you've come to the right place.

Steve: So actually I have a little bit of news about that, too.

Leo: Oh, good.

Steve: Allyn Malventano and I have been exchanging some mail.

Leo: Oh, nice. He's the king.

Steve: With some potentially really interesting news about or consequences for SpinRite. But we'll get to that in a minute. We're going to end up talking about an interesting research paper which was recently presented at one of the many ACM (Association for Computing Machinery) conferences about DNS consolidation and what has happened over the last few years, what the stats are, and what it means for the Internet in general and for how important you feel it is to keep your website online, whether it's an enterprise or a personal site or whatever.

But lots of stuff has happened. We've got a couple of new and forthcoming Chrome features. I'm going to quickly run through some new and notable ransomware casualties, including a couple of follow-ups on previous events. We'll look at a critical flaw in Drupal's CMS.

Leo: Again?

Steve: Their Content Management System. Yeah, there's a critical need for anybody who hasn't updated recently. We've got the big trouble with generic smart doorbells. The NCC group took a look at 11 of them, and what they found was just chilling. But shouldn't surprise us. We have a little takeaway from that. We've also got the interesting attack, the third actually by this developer, on Tesla's Model X key fobs. We have a Certificate Authority's adaptation to single-year browser certs, which we were anticipating. I got actually a note this morning from DigiCert. Maybe it was yesterday morning.

Anyway, we've got several instances of leaked credential archives and what they mean, a critical remote code execution vulnerability in a major MDM server platform, you know, Mobile Device Management. I have a bit of commentary on the Salvation Trilogy, not any spoilers, of course, but I'm at 92% and, oh, Leo. Also some extremely promising news about SpinRite's future as a consequence of something that the benchmark has been revealing. And then we're going to wrap up by taking a look at the consequences of the increasing consolidation of DNS service providers and what it means. And of course we've got a fun Picture of the Week, as we always try to have.

Leo: You have been very busy, haven't you.

Steve: Yeah.

Leo: Working hard, as always, on Security Now!.

Steve: Yeah. So this Picture of the Week is a chart which shows - it's a two-axis chart where we have rows of time to crack a password, where we go from four characters up to 18 characters. And then in columns of each of those rows is the complexity of the password at that length. So a password that had numbers only; that had lowercase

letters only; upper and lowercase letters; numbers, upper and lowercase letters; or, finally, numbers, upper and lowercase letters, and symbols. And so five different classes of password complexity on one axis and password length on the other. So what you get is an interesting diagram such that obviously in the upper left where it was four numeric password only, that's going to be instantly cracked.

Leo: Even if it's a mix of upper and lowercase and numbers and symbols, four is just too few.

Steve: Exactly. And then in the exact opposite corner, I don't even know what "7qd" - is that quadrillion, probably - seven quadrillion years.

Leo: It's a long time.

Steve: Which, you know, is probably sufficient.

Leo: And that's not even that long a password. That's 18 characters. That's not even that long. Mine are much longer.

Steve: Yeah, with upper/lowercase. And we're assuming it's not words, you know, it's random. But anyway, so this is no revelation to any of our listeners. But it's such a good chart that it's already, since I posted the show notes up on Twitter, I've seen it being retweeted, that is, this particular chart. I went to its source, HowSecureIsMyPassword.net, and they've got something actually that kind of copies the Password Haystacks concept where, as you're putting in a password, it's showing you how long, although it doesn't show you all the fun stuff that Haystacks does.

Anyway, my point is that this chart would be good for our listeners to share with people who are unaware of, like, what it means because, you know, for example, you could, like, look up your password. Okay, mine is 10 characters, and I've got upper and lowercase characters only. Well, this says a month. Now, of course there's a whole bunch of assumptions that are required. We know all that. But just to get the concept through. I thought this was a really nice presentation. So I wanted to share it with our listeners.

Leo: How recent is this? Is this modern computers? And do we worry even about quantum computing, like suddenly all of this stuff takes seconds because of a quantum computer?

Steve: Well, so there's a perfect example of all the things that are unstated. For example, is this brute forcing over the Internet, trying to get into an account? Is it an offline hack where you're able to run, employ GPUs, or even more, bitcoin hardware acceleration in order to do SHA-256 at ridiculous speeds? I mean, so this is meant to convey the concept of both complexity and length are factors, and to sort of give people a rule of thumb. I mean, and frankly, you know, my Haystacks page is the same way. It just sort of is meant to give you a sense of, like, longer is better. More complicated is better. So do that.

Okay. So Chrome's Omnibox is becoming more "omni." We were talking just before you hit the Record button on the podcast, Leo, about something that required Chrome in

order to work. I'm a Chrome user. Like, for example, the show notes are composed every week in Google Docs. And I just figure, might as well do that in Chrome because it's going to definitely like itself.

Leo: Right.

Steve: So Firefox is open statically to my left. And that's where my hundreds of tabs tend to accrue over time. Chrome I sort of fire up just because it's convenient from time to time. But so there was an interesting little blurb that I saw that Chrome was adding, which I think would appeal to our users. And what that is, is the ability to put commands, like UI-related commands, into the omnibox, which is what they call the URL field, the point being you can put, like, search terms in, or URLs or whatever, and the browser will try, not always perfectly, by the way, sometimes it annoys me, but generally try to figure out what you're asking and do the right thing.

But how cool would it be if you could type in "clear cache" or "delete history" or "wipe cookies" or "update browser" or "incognito" or "edit passwords" or "edit credit card" or "translate this page." That's coming. It's not turned on yet. It's being rolled out slowly over time. But anybody who wants to can turn it on today. If you put in the omni box "chrome://flags," that will bring you to the big repository of switches. And you then need to search on that page.

This is a search field at the top. Put in "omnibox suggestion." That will reduce the huge number of options to three. The first two are what you need to enable: "Omnibox suggestion button row," enable that, and normally it's set to default, which means that Chrome or Google will eventually flip that on for you when they decide everybody wants this. And then the second one, I don't know why it's called "pedal," but it's "Omnibox pedal suggestions," P-E-D-A-L. You see this sometimes where programmers come up with some strange term, and it survives the UI, and it ends up surfacing for some reason.

Anyway, "Omnibox pedal suggestions" and "Omnibox suggestion button row." Enable them both. Then in the lower right you'll see that you've been presented with a Relaunch button. So click that. When it comes back up, you can type things into the omnibox command, like "wipe cookies," "delete history," "clear cache" and so forth. Which is just kind of a cool feature. It's worth noting that Chrome is not the first browser on the block to do this. In fact, Firefox has had this since the spring of this year. But it's not as obvious there. I think the Mozilla people, being a little more conservative, were afraid that if you put in "clear cache," that might not be what a Firefox user wanted. Google doesn't seem to worry about that. You have to say "clear Firefox cache" in order for it to present you with a button.

What happens when you issue a command, or in the case of Chrome something that it thinks is sufficiently clear that you're talking to it, is that the upper suggestion in the little dropdown list of things that you might be asking becomes a button, which you can then click in order to immediately have that action take effect. So anyway, just a cool thing. As I said, Firefox you've got to add the word "Firefox" to get its attention, but also it tends then to false-positive less often. So anyway, just a cool little feature.

Also, and this might be relevant to people who use Chrome with a gazillion tabs, you know, I don't know you can when tabs are like stuck in a horizontal row. I have a gazillion because tabs themselves are horizontal so stacking them vertically is the only thing that makes sense. And some day the world is going to figure that out, but that's okay. There is now another forthcoming feature for Chrome where you can enable a tab

search feature to have Chrome search for text, do an incremental search on the text of all the tabs you have open.

So I think this sort of signifies that we're reaching that point in browser development where we've run out of good ideas, or things that we really need, and so now they're sitting around, scratching their butts, thinking, hmm. Okay, you know, we don't want to lay anyone off. So what can we have them do? Anyway, so if you start Chrome with "--enable-features=TabSearch," then a little button appears to the right of your row of tabs in Chrome, which if you click it, or if you do Ctrl-Shift-A for activate tab search, you're able to do an incremental search across all of your open tabs. I never have more than five or six in Chrome because as I said, it's sort of - I just fire it up for the moment and then shut it down. But for people who are big Chrome users, who end up getting, like, I know there's a tab here somewhere, this may be the thing that you've been looking for.

In ransomware news, a few things. Delaware County, Pennsylvania has paid a half a million dollar ransom after their systems were hit by the DoppelPaymer ransomware last weekend. And of course being Pennsylvania, that's on a lot of our politically focused people's radar because it was one of the loudly and hotly contested states in the U.S.'s recent presidential election. So of course the first question anyone has is whether the ransomware attack, which was recent, may have had any effect upon the state's election networks.

So Delaware County was quick to state that the Bureau of Elections and the county's Emergency Services Department, neither of those were affected. They're on a different network than the one that was hacked. Sources said that the county's in the process of paying this half a million dollar ransom since it's insured for such attacks. So they figure, hey, what the heck, let's pay the ransom, get the key, and get our systems back up.

We're hearing more about DoppelPaymer, and so I think this is one that we're going to be talking about, much as we've been talking about Sodinokibi and Ryuk and so forth. It was derived from its predecessor, BitPaymer. And it shares a large body of its code. DoppelPaymer has been improved to add multithreaded encryption because of course that's what you want in your whole server encryption is speed. So anyway, it's faster now. And in an odd twist, the DoppelPaymer gang apparently advised Delaware County to change all their passwords and also modify their Windows domain configuration to include safeguards from the use of the Mimikatz program. Now, it's not clear what those safeguards would be. Maybe like explicitly look for Mimikatz.

Mimikatz, we've talked about it from time to time, it's an open source tool that's been around for about six years, since 2014. It's commonly used by ransomware gangs to harvest Windows domain credentials when they get into a compromised network. So it's one of those lateral movement tools. It doesn't qualify as "living off the land" because it's typically not present on systems. The ransomware needs to download a copy in order to use it. But it is on GitHub. And its author six years ago explained that he wrote it as a way to learn C and experiment with locating and extracting Windows credentials from the RAM of running systems.

So, yeah, it's very much like that Active Directory tool we were talking about a few weeks ago. It wasn't ever really written to be used for malicious purposes. But, boy, is it handy for those. It extracts things. It finds them and extracts them out of RAM. And DoppelPaymer and this gang are using Mimikatz.

We did have some follow-up on Canon's attack. Remember we talked about image.canon going down. And I was remarking, it's like, wow, they've got their own top-level domain, .canon. I guess if you have enough money you can say, hey, I want a TLD. So Canon finally publicly confirmed what we all pretty much knew based on the evidence, was that they did suffer a ransomware attack in August, and the hackers stole data from the

company's servers. At the time, their cloud photo and video storage service, as I was mentioning, image.canon, went down, and it caused some loss of user data.

But we noted when we covered this on the podcast there was a long list of domains that were of Canon-related services that were all suffering outages. Shortly after the attack, BleepingComputer obtained information showing that the outage had been caused by the Maze ransomware, M-A-Z-E. And Maze told BleepingComputer that they had stolen 10TB of data...

Leo: Ooph.

Steve: Yeah, including private databases, before they had triggered the file-encrypting malware on the 5th of August. And interestingly, it turns out the trouble with at least the image.canon site they claim was unrelated to anything they did. They confirmed that their actions did not extend to Canon's storage service. So they have no reason to lie, I would think. So just a coincidence.

Another recent victim was U.S. Fertility, which is the largest network of fertility centers in the U.S. They've got 55 locations across 10 states. And they were just recently hit by ransomware, encrypting a bunch of their systems. And so they were suffering some outage.

I got a kick out of the fact that Ritzau, which is Denmark's largest independent news agency, refused to pay any ransom. They were hit a week ago, last Tuesday. Their spokesman said the Ritzau news agency - oh, they were founded in 1866 by Erik Ritzau, so they've been around for a while. The spokesman said the news agency was subjected to an extensive hacker attack on Tuesday, and the hackers have subsequently demanded a ransom to release data. Ritzau has refused to pay money to the hackers.

So anyway, they're a big, sprawling organization. About one quarter of their more than 100 servers on their network were encrypted. So their IT department immediately set to work restoring the systems and expected to have them up in a couple days. So that's how you do it. You have your technology nailed down. If the worst happens - and, boy, we're going to be seeing later in just this podcast, we will be looking at one thing after another that would have allowed ransomware guys to get in. And what we know is there are as many ways in as there are employees, essentially. So the only solution at this point is to be in a position where you can recover, and these guys clearly are.

This one actually touches on Chromebooks, which I thought was interesting. Last Wednesday the Baltimore County - and there's a distinction here between Baltimore County and Baltimore City, as we'll see. The Baltimore County Public Schools posted the news: "BCPS can now confirm we were the victim of a ransomware attack that caused systemic interruption to network information systems. Our BCPS technology team is working to address the situation, and we will continue to provide updates as available. For now, please don't use BCPS services." And of course all of this hits, unfortunately, amid the COVID-19 remote learning phase, you know, virtual education, because we're seeing the expected winter increase in cases. So school's out, essentially.

The Baltimore City Public Schools district, apparently distinct from Baltimore County, also published an alert on its website urging students to only use school-issued devices for virtual learning. Baltimore City wrote: "Students participating in virtual learning should only use City Schools-issued laptops or devices. Do not use devices issued by Baltimore County schools, or your personal laptop or computer."

They said: "Students without access to a City Schools-issued device will be granted an excused absence." So reading between the lines, it sounds like there's some concern that the malware might crawl out onto devices connected to the County network, but not the City network. Or perhaps Baltimore City has additional device protection on their devices.

But here's what I thought was sort of interesting. Following last Wednesday's ransomware attack that hit the district's network, Baltimore County Public Schools has now urged students and staff to stop using their school-issued Windows computers and only use Chromebooks and Google accounts.

Leo: That tells you something.

Steve: Uh-huh, yeah. The update on their website says: "We now know that BCPS-issued Chromebooks were not impacted by the cyberattack." Okay, so that really does suggest that they detected that students' Windows-based computers may have been affected by this. So perhaps the attack went out and put malware onto student-issued computers. They said: "You may now safely use BCPS-issued Chromebooks and BCPS Google accounts for students and staff. Please do not use BCPS-issued Windows-based devices until further notice." Oh, and school is out yesterday and through today at least. Students are instructed to check in with the website to get an update and figure out whether they'll be able to safely reconnect. So, boy, this is creating a mess.

And lastly, just because it's big and significant, the French multinational production and distribution firm Banijay Group SAS, who we probably better know for the various brands they produce, which include "MasterChef," "Survivor," "Big Brother," "The Kardashians," "Mr. Bean," "Black Mirror," "Extreme Makeover: Home Edition," and "Deal or No Deal," among a great many others, was also another recent victim of the DoppelPaymer ransomware. Although they've only shared that they have suffered a cyberattack, and that some of their data may have been compromised, the DoppelPaymer ransomware gang is not only claiming responsibility, but proving their involvement by sharing several documents, presumably stolen from Banijay's systems. DoppelPaymer is also taunting the French production group by referencing GDPR compliance issues and leaking an internal GDPR compliance document, among others. So the crooks are having fun at these guys' expense.

Okay. In security news, Drupal. The Drupal security advisory is titled: "Drupal core - Critical - Arbitrary PHP code execution." Which is not good news. That was issued last Wednesday. And if any of our listeners are running Drupal-based systems and haven't yet updated, do it now. It is a sweeping vulnerability. Drupal 9.0 needs to be updated to 9.0.9; 8.9 needs to be updated to 8.9.10; 8.8 or earlier to 8.8.12; and Drupal 7 needs to be moved to 7.75.

So obviously this is a sweeping problem affecting a whole set of their various lines. The project uses the PEAR (P-E-A-R) Archive_Tar library. The PEAR Archive_Tar library has released a security update that impacts Drupal. So this is another case like we saw with Google where remember they had the zero-day that was discovered because they were using the FreeType library font interpreter. So this is another instance where the use of a third-party library has bit somebody using that library when a remotely exploitable flaw was discovered that can be referenced through the user of the library.

The advisory states that: "Multiple vulnerabilities are possible if Drupal is configured to allow .tar, .tar.gz, .bz2, or .tlz file uploads and processes them." They said: "To mitigate the issue before fixing it, prevent untrusted users from uploading any of those file types." So here, I mean, we know what this means. This means that there is a glitch in this PEAR Archive_Tar library such that interpreting, when it tries to interpret the content of any of

those files, it turns out it's possible to maliciously manipulate those files in order to get remote code execution.

So what makes this all the more urgent is that there are known exploits against these vulnerabilities, and some Drupal configurations are known to be vulnerable. Of course, as we know, Drupal is a popular content management system. As of Friday, over 944,000 websites, so just shy of a million, 944,000 websites are using vulnerable Drupal versions out of a total of 1.12 million, according to the official stats. But it turns out even those stats probably underestimate the scope and scale of the vulnerabilities because only Drupal sites which are using the Update Status module are included in the data. So that sounds like something that allows Drupal to keep track of sites. So many more may be at risk.

Drupal is currently in fourth place among CMS systems on the Internet. WordPress of course has the huge lead at 63.8% share. Second is Shopify at 5.1. Joomla is in third place at 3.6, and Drupal is at 2.5%. But this just goes to show, even a 2.5% share of content management system puts it at 1.12 million sites. And as I said, as of last Friday, just shy of a million of those, 944,000, are using vulnerable Drupal versions, and exploits are known. So, yikes. Update.

And unfortunately, I think this is another trend that we're seeing. I mean, we've sort of seen it before, even back in the Stagefright exploit against Android, lo those many years ago. That was a media codec exploit. So it wasn't formally part of Android. It was like, hey, let's use this codec pack in our OS, and got bitten by something that somebody else wrote. So, you know, in this day and age it's difficult to be absolutely responsible for all the code that is running. Even I, where I wrote all of my backend web server stuff, but I still have IIS as fielding the incoming inquiries. It's just it's very difficult to be an island and get anything useful done.

Leo: Okay. On we go.

Steve: So hopefully it won't come as a surprise to any of our listeners to learn that they don't want to have off-brand no-name IoT smart doorbells.

Leo: Yeah. Yeah, that sounds right, yeah.

Steve: The likes of which are sold on Amazon and eBay. You don't want those anywhere near your homes. Matt Lewis of the NCC Group, and their researchers, took a look at the operation of 11 off-brand el cheapo bargain smart doorbells and found their intelligence to be somewhat lacking, shall we say. Matt said: "Our findings could cause issues for consumers and are indicative of a wider culture that favors shortcuts over security in the manufacturing process." Okay, no big surprise there. He added: "However, we are hopeful that the much-anticipated IoT legislation will signal a watershed moment in IoT security. Until this comes to fruition, we must continue to work together to highlight the need for basic security by design principles, and educate consumers about the risks and what they can do to protect themselves."

So, okay. So first of all, this IoT legislation Matt's referring to, it's hopeful. It's been moving along since its introduction in 2017 by Senator Mark Warner. And it uses the typical "what can the government do" carrot-and-stick security requirements. So, you know, you need to do these things in order to get a government procurement contract. And it does include a bunch of stuff we need, so clearly some IoT security-aware people were involved.

The legislation, if it happens - and it's like it's still alive, and maybe it will. It requires that vendors, for example, commit that their IoT devices are patchable, so yay for that. Also that the devices don't contain known vulnerabilities. If a vendor identifies vulnerabilities, that vendor must disclose them to an agency with an explanation of why the device can be considered secure, notwithstanding the vulnerability, and a description of any compensating controls employed to limit the exploitability and impact of the vulnerability. And then based on that information, an agency's CIO could issue a waiver to request the ability to purchase the device.

The third requirement is that the devices rely on standard protocols, which only seems great. Let's not roll our own and say, oh, this is better than anything else. And then, fourth, the devices don't contain hard-coded passwords. So that's obviously a really good thing. It's going to be difficult, though, to do because, when you think about it, I mean, hard-coded passwords we know are bad. But not having them is going to require some interesting workarounds. So the bill is not yet law. And there are exemptions in there that are so large you could drive a truckload of dumb doorbells through them. So we'll see how this turns out. But it's certainly true that having any sort of legislation would be a lot better than just what we have now, which is this totally unregulated environment.

Okay. But specifically, of these 11 devices, two of the devices that these guys tested were manufactured by Victure (V-I-C-T-U-R-E) and Ctronics (C-T-R-O-N-I-C-S). They had critical vulnerabilities that could allow bad guys, not surprisingly, to steal the users' home network password. The flaws would also allow attackers to hack not only the doorbells, but also the residential router that the doorbell was connected to, and any other smart devices in the home, thermostat, other cameras, and even get into household computers.

And, for example, this Victure Smart Video Doorbell is its formal name, was found to be sending the customers' home WiFi network name and password, unencrypted, to servers in China. So, you know, there is absolutely no need for a locally connecting WiFi device to export its local network credentials to anywhere. But, you know, you buy some Victure smart doorbell from Guangdong, China and plug it in; and, I mean, you know, hey, look, it works. Well, yeah. But it's also sent your network credentials to China. Maybe that's not a big problem. But it doesn't have to do that. There's no conceivable use case for it doing that. So maybe that's not what you want.

And remember, Leo, the visual that we set up on a podcast a few months ago? Actually, we were talking about this after I had attached a few smart plugs to my network. I wanted some simple timers.

Leo: Oh, yeah, yeah, yeah.

Steve: And in fact I did add an IoT thermostat to my environment here, and I had also a humidity sensor and a separate logging thermostat. And I could just imagine a globe showing - and of course in movies like "War Games" we see all the little lines tracing an arc through the upper atmosphere of incoming missiles. Well, imagine all of the IoT devices in the U.S. with their connections back to servers in China. I mean, that's where they're connecting. That's where my doorbell - actually I don't have a doorbell. That's where my thermostat and my IoT plugs are connecting, which of course is why I would argue it is crucial that they be on an isolated network, as all of mine are. It's sort of, you know, when you picture that all of these tens, hundreds of millions of connections, we remember that story about the Trojan horse from olden times.

So anyway, Matt said: "If stolen, this data [obviously] could allow a hacker to access people's home WiFi, enabling them to target their private data, access any smart devices

they own," and so forth. The researchers found that another device, bought from eBay and Amazon without any clear brand associated to it, it was literally a no-name smart doorbell, was vulnerable to the KRACK exploit. That's the Key Reinstallation Attack that was discovered three years ago in 2017. So these devices don't have up-to-date WiFi stacks. Imagine that. On the other hand, why would they? Of course the KRACK attack opens any attached network to intrusion by allowing the network's WPA and WPA2 encryption to be cracked without much effort, given state-of-the-art cracking tools.

So of course, again, none of this comes as any great surprise. But I think it's nice to examine some specifics from time to time because it's too easy to sort of wave off generalities. The advice, of course, if you want to buy a smart doorbell, there is, I would say, very good reason, especially a smart doorbell. You've got a video camera; right? That anybody monitoring it can see what's going on out of your front door, can see when you all leave the house, can see when you come in, can watch what's going on around you.

Anyway, I would argue there's very good reason to stick with major brands. You're going to pay more. But, I mean, you have to care about security. And, I mean, unless you take personal responsibility for what one of these devices does, you have to isolate it on its own network. And I would argue, buy from a major brand. Even if there's a problem. And we know these things are going to have problems. The problems will make the headlines. The vulnerabilities will be found and cured responsibly and promptly as opposed to absolutely never. So anyway, I just thought it neat that these guys took the time to just sort of say, let's take a look at these doorbells and see what they're doing. And, yeah, to no one's surprise.

Now, the good news on the most recent of three Tesla key fob hacks is that it's not easy to do. But it also shows that even well-designed devices can get hacked. As Tesla found out for the third time, from this one researcher, and we've spoken of him before, Lennert Wouters (W-O-U-T-E-R-S), he is a Ph.D. student at the Computer Security and Industrial Cryptography group at the Catholic University of Leuven, that's KU Leuven - we've covered a bunch of the work coming out of there - located in Belgium. He came up with - the press said he "discovered" this. Well, I guess. But it's not like he tripped over it, as we'll see. But he worked out a method to overwrite and hijack the firmware of Tesla Model X - is that Model 10? Is that how we say it, Model 10?

Leo: Yeah.

Steve: Key fobs.

Leo: No, no, I'm sorry. Apple is 10. Tesla is X.

Steve: Okay. Model X.

Leo: It's very confusing.

Steve: Thank you. Because it's not Space 10; right? It's SpaceX.

Leo: Right, it's SpaceX, yeah. No, it's a Model X, yeah.

Steve: So, okay. So what this guy did, he figured out a way, a hack that would allow him to steal any car that isn't running the latest software update. His attack only takes a few minutes to execute and requires inexpensive hardware. But I'll explain it. It's not that easy to do. This guy, same guy, has previously produced successive successful attacks against Tesla's security in 2018 and 2019. And now he adds 2020.

Leo: He'd better get his Ph.D. I'm just saying. He deserves it. He's earned it.

Steve: Yeah, yeah. And you know that when he called the Tesla security people, they took his call.

Leo: Oh, yeah, yeah. And they fixed it, we should point out.

Steve: Yes. So he explained that his third attack works thanks to a flaw in the firmware update process of the Tesla Model X key fobs. It can be exploited using an electronic control unit, you know, the brains, the ECU, salvaged from an older Model X vehicle, which it turns out they've been around long enough, they can easily be acquired online, like from eBay or other stores or forums selling used Tesla car parts. Wouters said attackers can modify the older ECU to trick a victim's key fob into believing the ECU belonged to its paired vehicle. Right? So you're using this ECU that you bought used to trick the victim's key into believe it's pairing with its own car.

Once that's done, once you've pulled off this trick, then the ECU is able to push a malicious firmware update out to the key fob over Bluetooth. Since the key fob update mechanism was not being properly secured, they were then able to wirelessly compromise the key to take full control over it. And subsequently, they could obtain valid unlock messages to unlock the target car later.

Okay. So here's how the whole attack works, in practice. The attacker approaches the owner of a Tesla Model X vehicle. In this first phase of the attack, they must briefly get within about five meters of the victim. So at that point they're probably at Bluetooth Low Energy. Later it switches to full Bluetooth. But at this point they need to be within five meters of the victim to allow the older modified ECU, that is, this spoofing ECU, to wake up and ensnare the victim's key fob.

The attacker then pushes the malicious firmware update into the victim's key. Although this phase, after that initial contact, which can be brief, but it has to be close, now they're able to about to be about within 30 meters of the victim, and it takes about a minute and a half, about 90 seconds to execute the firmware push into the victim's key. But that at least allows the attacker to put some distance between themselves and the targeted Tesla owner, thus reducing suspicion. They don't have to hang around, like for a minute and a half.

Once the victim's key fob has been hacked, the attacker is then able to extract car unlock messages from the compromised fob. The attacker then uses these unlock messages to enter the victim's car. That gets them into the car. The attacker then connects the older ECU to the hacked Tesla's car diagnostics connector, which is normally used by Tesla technicians to service the car. The attacker uses this connector and another couple minutes to pair their own key fob to the car, which they are then later able to use to start the vehicle and drive away.

So essentially the attack is use an old ECU, get close to somebody with any not-yet-updated Tesla Model X for a total of about a minute and a half, most of which time you

can spend at a distance. That subverts the key in order to get it to release a bunch of unlock messages that it has stored. The bad guy then uses those to unlock the car to get into the car and trick the car then into pairing it with a new key, essentially adding the attacker's key to the permanent paired list. And then they've got a completely valid key that allows them to do anything that a valid owner of the car would do. So again, not easy, but clever.

The downside is the attack uses a relatively bulky rig. You know, this ECU is not small, so you've got to stick it in a backpack or a bag or maybe like in another car that's parked next to the victim or something. But still, the attack is feasible. And it's not expensive. You need a $35 Raspberry Pi, a $30 CAN bus shield on the Raspberry Pi in order to connect to the Tesla's CAN bus, a modified key fob, that older ECU from a salvaged vehicle - apparently those are 100 bucks now on eBay - and also about a $30 lithium polymer battery in order to power the whole portable rig.

And of course, being a responsible lad, after discovering the bug and developing the exploit earlier this summer, he reported it to Tesla's security team in mid-August. Only after Tesla began rolling out an over-the-air software update to all Model X cars did Lennert then publish his findings. And the software update containing the fix is 2020.48. So don't leave the garage without it.

Leo: That's an expensive vehicle. So it may be a lot of work, but it's also justified if you can get it. I mean, it's well over a $100,000 vehicle.

Steve: Yeah. And, you know, again, it's a rolling computer. It's funny, there's some commercial on some channel that I listen to where someone is saying, talking about someone's car, "Have you ever looked under the hood? It's like a computer on wheels." And I'm thinking, no, it's not "like" a computer on wheels.

Leo: It is.

Steve: It is a computer on wheels.

Leo: Yeah. My Tesla was a beta computer on wheels, that's for sure.

Steve: Uh-huh. And you were the beta tester.

Leo: I was the beta tester.

Steve: What was it, that like reverse went out or something?

Leo: Well, when we first got it, you would drive it thinking - I had it in reverse, I could have sworn. But it went forward.

Steve: Oh, that's what it was; right.

Leo: Yeah. And it dinged a car in front of us in a driveway. But then I called them, and they said, well, no, we were looking at the logs. You didn't have it in reverse. It's like, oh, I didn't realize you had logs. So they record everything. Geez, Louise. And then there were a few other things. The door kept closing on Lisa. She didn't like that. It's supposed to have all these sensors in it, but it turned out there really weren't that many.

Steve: That'll put you off a little bit.

Leo: Yeah. Finally at the point where she said, look, before you close the doors, because they're all automatic, you have to shout "Doors are closing" and wait for an acknowledgment that everybody's out of the reach of the door, especially those gull-wing doors because they clonk you right on the head. And then they had a - I can go on. You probably don't want me to. But it had a seat that tried to eat us once.

Steve: I did hear you say at some point, it wasn't on this podcast, that you were not planning to buy another Tesla.

Leo: No, I'm not. No, I'm not. Mostly it was - it wasn't the bad experience, although Lisa didn't like it. It was just that there were issues with parts and delays and stuff. And I just thought, I want to buy it from a company that actually makes cars for a living.

Steve: Well, and you were a pioneer.

Leo: I wanted to support Elon. I did, yeah.

Steve: You were right there, first in line.

Leo: And my next car is electric, but it's made by Ford. Perhaps you've heard of them. You know what, though? I would trust Elon with security more than I would trust Ford. Who knows? The new Ford uses an app on the phone to open the door.

Steve: The good news is, nobody now...

Leo: Wants a Ford. Oh.

Steve: ...can use the excuse of, oh, we really didn't consider security.

Leo: Right, exactly.

Steve: It's like, sorry, honey. That ship has sailed.

Leo: Yeah, ship has sailed.

Steve: So as we were expecting, certificate authorities have adapted to the new single-year certs. Traditional high-reputation non-automated certificate authorities are reacting to the browser industry's decision - which as we know was initially instigated by Apple, but then quickly adopted with some sigh of relief because Google was trying this and couldn't get it to happen. Apple just said, "We're doing it" - to enforce a maximum life on browser certs of 398 days. And as we know, this went into effect on September 1st of this year, affecting any certificates issued from September 1st on.

And this morning - oh, it was this morning - I received a note from my chosen and quite favorite certificate authority, DigiCert, explaining the way their new multiyear plans would function. The essence is what we expected. Despite the shortening of individual certificate lifetimes, that is, the cert itself cannot be valid for more than 398 days. It will now be possible to sign up in advance for a multiyear commitment. DigiCert provides up to six years. And as you'd expect, a longer commitment results in a lower cost per year, which seems fair since you are giving them your money upfront, and in return they're giving you a discount. And once that's set up, you're in control of your own certificate renewal, able to reissue certificates at any time that's needed.

And in DigiCert's case this is essentially an extension of the system they've had in place for years. I've mentioned on this podcast a number of times how convenient it has been to be able to issue a certificate at midnight on the weekend and receive it within minutes. Essentially, DigiCert decouples the authorization of proving I am who I say I am from the issuing of certificates so that they do periodically need me to reverify my identity, my corporate affiliation and so forth. They do all of that validation automatically every so often, which then frees me within those windows to issue certs whenever I want to.

So, you know, all certificate authorities are now needing to compete with ACME certificate automation, which of course was pioneered by Let's Encrypt. And as we know, I used to use EV certs and proudly had the Gibson Research Corporation in green in the browsers. But then the browsers made the decision across the board to deemphasize the display, any display that certs were Extended Validation. And also because my use of subdomains had been growing, with sqrl.grc.com, forums.grc.com, news.grc.com, and EV certs do not support wildcards by policy, it just made more sense to switch from EV to OV, Organization Validation certs, which is where I am now.

So anyway, I still think that given the decidedly mixed blessing of certificate automation - yes, it makes them free. It allows you to have encryption. It really doesn't do the job you could get about authentication, though, because we know that the fraudulent issuance rate is exceedingly high with automated certs. So there's a tradeoff. And I still think it makes sense to add an indicator in a cert about whether it was issued by automation or under human supervision of some kind. That, to me, that seems like a useful flag, but maybe that's just something, another thing users won't pay any attention to. I don't know.

Unfortunately, nearly 50,000 Fortinet VPN credentials have been posted online, through no fault of Fortinet's, I should add. Last year it was revealed and we talked about it at the time, that the FortiOS, which underlies the Fortinet VPN, was subject to a path traversal flaw. It was assigned a CVE of 2018-13379. And the NIST description of the vulnerability reads: "An Improper Limitation of a Pathname to a Restricted Directory" - then it has in parens (Path Traversal) - "in Fortinet OS" - and it's got a range of OSes - "under their SSL VPN web portal allows an unauthenticated attacker to download system files via special crafted HTTP resource requests."

On August 28th of last year, 2019, Fortinet posted: "At the recent Black Hat 2019 conference held in Las Vegas this past August 3-8, security researchers discussed their discovery of security vulnerabilities that impacted several security vendors, including Fortinet. All of the vulnerabilities impacting Fortinet were fixed in April and May of 2019." Meaning months before this disclosure, they got all this fixed. And of course, being a responsible company, Fortinet had also worked to notify all of their users to update their systems. And of course, despite their efforts to get their users to update, we know how that probably went.

The vulnerability allows Fortinet VPN system files - or allowed, I should say, in the past - to be obtained remotely and without any of that pesky authentication. This is in the news today because a large, nearly 50,000 records worth, of previous Fortinet VPN logon credential information has recently surfaced on the dark web and is now being passed around. It's a 7GB archive of individual SSL VPN web session files which contain session-related information including plaintext usernames and passwords and the IPs, both of Fortinet VPN users and the IPs they were connected to - banks, telecoms, government organizations from around the world. I mean, Fortinet is a real company, so they've got a big high-end install base.

Since the aggregated and leaked archive contains all the login information needed, patching any still vulnerable or even previously vulnerable Fortinet VPNs at this point won't prevent any of their accounts from still being used. So remediation, which Fortinet was very clear to explain in their disclosure, requires not only patching the VPN, but then also canceling and reissuing all of the previous VPN accounts because they may have been able to get loose. So basically what was obtained, obtainable back then at the time, known before the Black Hat conference and foreclosed; but again, this is the problem we have in our industry with patches not being applied. A bunch of people still got their logs exfiltrated from them. So anyway, what we really need somehow is a way of closing this loop, of getting these connected systems updated, or at least getting notices to the affected individuals out in time.

And speaking of accounts, more than 300,000 Spotify accounts have been hacked. The security industry has renamed what we once called a "brute force attack." Now we're referring to these as "credential stuffing" because that's essentially the process. You're stuffing credentials down into a form on a server or an API access point, trying to get a brute force success. And of course, as we know, it's typically performed by taking a list of previously used usernames and passwords and just pounding away at some poorly protected service, attempting to get a successful logon.

And I say "poorly protected" because today, any service worth its salt will observe that some attacker at some IP is attempting to brute force their way in. I mean, it only takes, what, five or six attempts with wildly different usernames and passwords to think, okay, this doesn't look legit. And yes, a large multi-IP botnet could also be used so that it's not just one IP or a few IPs pounding on the service. But okay, so it's a more diffuse attack. But it's still trivial, or it should be, to quickly blacklist even a large number of IPs which are making repeated unsuccessful attempts to log in. And when I say "repeated," the list it turns out was 380 million - 380 million usernames and passwords.

But Spotify doesn't do that. And of course they just allowed this attack to go on. Spotify provides no such protection, nor does Spotify provide multifactor authentication, which would have also successfully thwarted any sort of credential stuffing attacks. So it should come as no surprise that Spotify's history has been defined by years of their users complaining that their Spotify accounts were hacked. After passwords were changed, new playlists would appear in their profiles. Their family accounts had strangers added from other countries. That's what happens in today's world when authentication is weak.

VPNMentor produced a report titled "Spotify Targeted in Potential Fraud Scheme." And in the report they wrote: "We unearthed an Elasticsearch database containing over 380 million records, including login credentials and other user data being validated against the Spotify service." In other words, this database showed that it had successfully been used to log into 300,000, actually between 300,000 and 350,000 individual Spotify accounts. They said that the origins of the database and how the fraudsters were targeting Spotify were both unknown. The hackers were possibly using login credentials stolen from another platform, another app or website and using them to access the Spotify accounts.

They said: "Working with Spotify," they wrote, "we confirmed that the database belonged to a group or individual using it to defraud Spotify and its users. We also helped the company isolate the issue and ensure its customers were safe from the attack." So they started with about 380 million records, which allowed them to get into between 300 and 350,000 individual accounts. So that meant they had a hit rate of around 0.1%, or like one in a thousand. So certainly enough successes to be useful.

They notified, when they found this database on an Elasticsearch site, they contacted Spotify back in July of this year, informed them of the exposed database and the threat it would produce. Spotify reacted immediately, initiated a rolling reset of passwords for all the affected users, and this database showed where login had been successful, where this username and password had successfully logged in. It was flagged in the database as, yup, this is a good one. So they got the database, performed resets for all affected users, and resolved the problem.

So, you know, the lesson for of course end-users we know. Never reuse passwords; make them long and high-entropy; and if possible always add multifactor authentication when it's provided. The lesson for Spotify and other services: protect your users by spotting and proactively blocking clearly malicious authentication attempts. It's easy to do. Do it. And immediately offer and promote time-based multifactor authentication, like why wouldn't you? Crazy.

Leo: It does seem rare to see a company that doesn't have two-factor these days.

Steve: I know.

Leo: I had no idea Spotify didn't allow two-factor. That's crazy. I'm surprised.

Steve: Maybe they don't want to make it difficult to logon, don't want to bite the bullet.

Leo: Plus maybe they figure, oh, it's just a music service. Who cares?

Steve: Right, that may be. And in fact, when it talked about defrauding users, I thought, well, okay, it's not clear if you log in as a user, you know, normally you're not able to see your payment information. So they wouldn't be able to get that. That would be masked.

Leo: The defrauding is adding Yevgeny to your family account. Oh, my brother in Bratislava, yeah, of course. He is good, good guy. I love him. Strange taste in music. That's the worst thing that would happen. You all of a sudden get all this weird music out of your taste in your suggestions.

Steve: Spam in your Spotify account.

Leo: Why is he subscribing to all these criminal podcasts? Yevgeny.

Steve: So last piece of news is that Mobile Device Management (MDM), as we know, is a popular means for enterprises to manage the configurations of their mobile phone fleet. And because MDM servers, by nature of the way they work, must be publicly accessible to remotely manage those mobile devices. They are a natural target for bad guys. That means when a remotely exploitable code execution vulnerability is found, enterprises need to take heed.

One of the more popular MDM services platforms, MobileIron, is vulnerable to just such exploitation, which carries a CVSS score of 9.8, which while not 10, is up there. The flaw was reported to MobileIron by Orange Tsai from DEVCORE. It exists across various components of the platform. Actually, I don't know the nature of it. I didn't take any time to dig into it because it's been fixed, and everyone needs to update. But it's in the MobileIron Core, which is a component of the MobileIron platform that serves as the admin console; and in MobileIron Connector, a component that adds real-time connectivity to the back end.

Also impacted is Sentry, an inline gateway that manages, encrypts, and secures traffic between the mobile device and the backend enterprise systems; and the Monitor and Reporting Database, which provides comprehensive performance management functionality. So whatever this thing is, it's like a fundamental flaw in something about what MobileIron was doing. Sounds like a big mess. And it's certainly not one that any enterprise wants to have on the network.

MobileIron said in an update last week that it had been engaging in proactive outreach to help customers secure their systems. And they estimate that 90 to 95% of all devices under mobile management are now being managed on patched and updated versions of the MobileIron software. The company stated it will continue following up with the remaining customers where they can determine that they haven't yet patched the affected products. So again, mistakes happen. Maintaining tight lines of communication is the key. And we really do need automated updating, or at the very least automated update notifications. You know we're getting there. But, boy, the progress is slow.

So I did just want to say that it's been a long slow burn, and frankly even sometimes a little bit of a slog since Peter Hamilton packs his novels with seemingly endless detail.

Leo: Oh, man, it's long. It's a trilogy.

Steve: Oh, my lord. And sometimes you're like, why do I - do I really need to know about this? But it all comes together. And now I don't ever want it to end. As you said, I'm at 92% of the third book, and oh, my goodness. I will be rereading what I just read last night over again because it is so good. And the other thing, I don't know what Peter has in store, but I still have 8% left. And we've sort of wrapped up most of what I was expecting. There is a big loose end. But it feels like there may be a surprise because I will never forget the surprise, the breathtaking surprise at the end of "Fallen Dragon," where it's like this happens, and it's like, oh, I mean, and then suddenly everything that, I mean, you'd been set up for this surprise. And so anyway, we know he's fully capable of doing this. And anyway, I just wanted to say wow. You know, it was the - what was the first crazy series that he made really popular?

Leo: Was that "The Abyss" or the - oh, I love all of his stuff.

Steve: Yeah. It was the very early stuff.

Leo: Not the one with Al Capone. Oh, "Pandora's Star."

Steve: Yes, yes, that's what I was thinking of.

Leo: And then "Judas Unchained," the Commonwealth Saga.

Steve: Oh, my god. And then "Pandora's Star."

Leo: Yeah.

Steve: "Pandora's Star" and "Judas Unleashed."

Leo: Yeah, unbelievable stuff.

Steve: Or "Unchained" or something, anyway.

Leo: They call it the Commonwealth Saga, yeah. Really good. That's another trilogy.

Steve: That was just two.

Leo: The one that I didn't like was Al Capone was - oh, that was just two, you're right - was "Chronicle of the Fallers." I wasn't crazy about that.

Steve: Yeah. And in fact, yeah, the one with Al Capone, it kind of went off the reservation at some point.

Leo: Yeah, it got a little weird.

Steve: It was like, uh, what?

Leo: Literally, Al Capone is in it.

Steve: Yeah. Okay. In the "surprising news" category, the work on the ReadSpeed benchmark is tantalizingly close to completion. I finally published a first release candidate, which had the effect of soliciting some additional testing, and the result was

that we discovered a couple of final things that needed a little polish. And I'll get back to that this evening. But the benchmark has revealed something that's quite exciting. So I reached out to Allyn Malventano, whom you know, Leo.

Leo: Yeah, I'm trying to get you two together this week. But king of SSDs, a piece of perspective, and then he got a job at Intel doing SSDs over there. So he really knows his stuff, yeah.

Steve: Right. So I sent him an email titled "Interesting SSD timing reveals ... something." And I showed him what we were seeing. And you and, I think next week, you and I will be looking at the same file. So now he's over at Intel with a title of Storage Technical Analyst. And I'll go into more detail about this soon, as soon as I have something for everyone to play with. But it looks very much as though SpinRite will have an extremely bright future in SSD maintenance and repair. The benchmark provides read timing resolution with an uncertainty of less than 200 picoseconds in its ability to measure response time.

And it turns out that this could allow it to spot regions of SSD that are weakening long before they fail. In the time domain, it would be like having a microscope with an extremely high magnification. And what this reveals, it reveals the effects of mild regional slowdown, which it turns out is a natural consequence of SSD management through something known as the FTL, the Flash Translation Layer, which is the logic which manages the mapping of the underlying SSD media to its external presentation.

And it turns out that standard usage of many SSDs will result in a mild reduction in performance. Many of the early testers of the benchmark are noting that the front of their SSDs are a little slower in performing than later on because the benchmark tests five locations on each drive: the front of the drive at the 0%, the middle at 50%, and the end at 100. And then also the two quarter points at 25 and 75%. So you can sort of see, you know, one of the first things we saw was that the beginning of people's SSDs were slower, which was like, what?

Well, it turns out that usage causes fragmentation of the SSD, very much like, sort of akin to traditional hard drive fragmentation as a consequence of use. There it's at the file system level. Here it's actually below the logical access level. But we're also seeing something very different, which is what caused me to send this note to Allyn. It reveals when that FTL layer is having extreme difficulty reading a region of its media. On today's multilevel cells, you know how like we used to have single-level cells, where you'd store either like zero charge or full charge, and that would mean that you could store one binary bit. But now, of course, in this quest for greater density, if you were to store four levels of charge, then that would allow you to store two binary bits in a single cell. What they're often doing now is what they call "triple level," but it's actually eight levels of charge and three bits per cell.

And in fact, Leo, believe it or not, this technology has gotten so crazy that SSDs will advertise what their maximum capacity is. But they will, because it's more reliable, and you may never need maximum capacity. They're even able to start out using one bit per cell until the whole SSD fills up at the one bit per cell level, and then dynamically start allocating two or three bits as you continue to fill up the SSD.

Leo: Wow.

Steve: And in fact that may be, because SSDs are not allocating...

Leo: Capacity goes up as time goes by.

Steve: Well, actually the bit density of the actual SSD capacitor is increased.

Leo: Geez, Louise.

Steve: So that it goes from only storing a zero or a one; to storing a zero, one, two, or three; to storing a zero, one, two, three, four, five, six, seven as you actually need more of the capacity of the SSD.

Leo: And that's part of the wear leveling, too, I would imagine. It gives them - they can spread it out over the platter, whatever it is.

Steve: Yes, yes. But the multilevel cells are - they take longer to read. I mean, sorry, well, actually they do take longer to read. They also take longer to write because you're needing to write multiple levels into different cells. And, I mean, it's nuts.

Leo: It's amazing. It's amazing.

Steve: Oh, Leo, the technology that's hidden behind these things is crazy.

Leo: It's incredible, yeah, yeah.

Steve: So, and Allyn uses the term "cell drift" because that's another thing that can happen is that, you know, that's - in fact, we talked about this a long time ago where SSDs that were stored in a hot environment tended to lose their charge faster than those stored in a cold environment, which makes sense just in terms of electron activity. Anyway, it turns out that on today's multilevel cells, this Flash Translation Layer may need to tweak its cell voltage thresholds in order to deal with what Allyn calls "cell drift," and/or apply extensive levels of error correction in order to recover a specific region's original contents.

What we are seeing, and what the benchmark reveals, is that you'll be cruising along and suddenly come to a grinding halt while a particularly troublesome region is being read. And so what happens on lower end SSDs, they may not be using hardware acceleration in order to perform the error correction math on the fly. So they're having to do a firmware algorithm in order to recover the contents of a specific sector that you're asking to read.

Anyway, it may be that identifying and then applying careful selective block-aligned rewriting, a future SpinRite will be able to literally recharge and realign these drifting cells to bring a possibly endangered region back up to speed. And then, if an area cannot be repaired, then oddly enough, we may be back to the very early concept of marking a region as unreliable and taking it out of the file system. And when you think about it, with the crazy size of today's mass storage, it makes a huge amount of sense to remove a tiny fraction of possibly unreliable storage since everybody has way more than they will ever be able to use.

Leo: Right.

Steve: Anyway, when I suggested this in my email to Allyn, he replied. He said: "Actually, there's no need to map out clusters at the file system level, and this is one of the cases where it may be beneficial to go [and he put in quotes] 'old school SpinRite' and rewrite successfully slowly read sectors." He said: "Unlike hard drives, SSDs won't typically do any rewriting on their own, even if a sector was *very* [and he had in asterisks] difficult to successfully read. So these very slow reading areas can be remedied by rewriting them. You can minimize," he said, "an increase in media/FTL fragmentation by ensuring you do these operations in 4K or 8K aligned chunks and not just single sectors."

And of course, you know, we know that a future SpinRite would do both. It would characterize the overall performance of the media to learn how it performs. It would then identify any regions that are clearly operating far below average, and then first attempt to resolve the trouble with careful selective rewriting. But if the region did not increase its read speed, if it refused to be healed, then it would do what SpinRite originally did back in version 1.0, which is to take that region out of the file system and relocate any data that it has to safety, and then allow SpinRite to be used on SSDs as a really proactive preventive management tool.

Leo: That's really great news. It makes SpinRite even more useful on SSDs. That's great.

Steve: Yeah, well, and in fact, to our surprise a few years ago we started sharing testimonials from people whose SSDs and various types of thumb drives were being brought back to life by SpinRite. And it was like, what? Well, now we know how. Oh, and I mentioned that SpinRite will be fast. The benchmark is reading 544MB per second from a 500GB Samsung 860 EVO SSD when attached to a SATA III port. That means that SpinRite will be able to perform this sort of whole drive performance scan in 919 seconds, or fewer than 15.5 minutes.

Leo: Nice.

Steve: So it becomes practical and feasible again to use SpinRite for this.

Leo: Nice. It's exciting.

Steve: Yeah. It's very, very fun. And our listeners are all going to be able to find out how their drives, both spinning and solid-state, perform in this way. I would imagine next week we'll have this thing ready. So anyway, very cool.

Leo: Can't wait. Nice.

Steve: So DNS consolidation. Despite previous teachable moments, such as when DynDNS - we all remember that, Leo, we talked about it four years ago, it was in 2016 - when DynDNS was attacked, and the result was huge swaths of the web became

inaccessible. What happened was our dependence upon fewer and fewer large providers - in other words, DNS consolidation - was revealed. And since then it has only grown. Today, for example, the research that was just performed reveals that if Cloudflare, AWS, or GoDaddy were to go down, around 40% of Alexa's Top 100,000 websites would also go down, due to a failure of their now-consolidated DNS.

And for those who weren't around when we covered this four years ago in 2016, Dyn, which was subsequently purchased by Oracle, Dyn is a provider of managed DNS services. It was a victim of a massive DDoS attack by the Mirai botnet that crippled the company's operations and took down the DNS of more than 175,000 websites. And although some sites managed to remain accessible thanks to well-configured secondary DNS servers that had been wisely kept on different networks, most sites were not prepared and remained down for nearly a day as Dyn worked to remediate the attack.

So a team of researchers at Carnegie Mellon University have conducted a large-scale study of the top 100,000 websites on the Internet to see whether and how those responsible for website operations reacted to this attack, if at all, four years ago, and how many are still operating with a single DNS provider and no effective backup. Their 14-page research paper is titled "Analyzing Third-Party Service Dependencies in Modern WebServices: Have We Learned from the Mirai-Dyn Incident?" And for anyone who's interested, I have a link in the show notes to their 14-page paper. It was presented during the ACM's Internet Measurement Conference last month, and in it they show that today, in 2020, 89.2% of all websites use a third-party DNS provider rather than managing their own DNS server.

So, you know, that's not so bad; right? 89.2% are using a third-party provider. On the other hand, that's a lot of concentration because there aren't that many third-party providers. And if those third parties go down, that takes out everybody who's using them. So them needing to stay up is important. And, yes, a company like Cloudflare is super robust, is doing everything it can to stay up. But even mistakes happen, even if it's not a big attack. In fact, we talked about there was a Cloudflare mistake of this sort, I don't know, a year or so ago. And they fixed it quickly. And we talked about it in detail. It was a really interesting mistake in the management of their routing that caused everything, all their traffic to go to one place, which just collapsed it.

They also found, to make matters even more fragile than this concentration onto a few third-party DNS providers, that 84.8% of all analyzed websites relied upon a single DNS provider, without any backup redundancy to which they could switch in case of a failure or attack. Now, I mean, I get it. People don't understand, I guess, what that means. Or maybe it's just not really that important that they stay online. And so they're making a deliberate tradeoff of convenience versus reliability. But the awareness of the need for DNS redundancy dates all the way back to the birth of DNS.

We're all techies listening to this podcast. We've all seen at least two fields or a pair of DNS IPs appearing wherever configured DNS addresses are. I mean, you always have two DNS IPs. It's regarded as best practice to have redundant DNS servers on nonadjacent IPs, typically on differing Class C networks. And of course for true redundancy, the further apart they are, the better. It'd be great if they were actually on entirely separate providers. But if the configured servers are actually sitting next to one another in the same rack, that is, if they're just, you know, it's like, hey, yeah, they deliberately - I know that, for example, Cox, I have IPs, they look like they're on different C Class networks. But it's like, okay, what do you want to bet they're in the same room? Anyway, if that's the case, any benefit is illusory, of course.

So the CMU team says that the number of sites having no effective redundancy has increased by 4.7% since 2016. And they suggest that this demonstrates that the lessons website operators might have learned following the Dyn DDoS attack and outage were

instead lost, if they even ever were learned, and forgotten. They point out that while two of the top 100 sites - two of the top 100 sites - did add backup DNS servers since 2016, that means that 98 of the top 100 did not. They also noted that smaller websites continue to use a single DNS service provider without any backup, and in most cases the operators of these smaller sites chose a large, well-known provider, thus contributing to the long-observed tendency toward the consolidation among ISPs.

In the show notes I have a chart from their paper which just sort of shows, based on the Alexa rank, whether you're in the top 100, 1,000, 10,000, or 100,000, what is the amount of third-party dependency, critical dependency, and how much and how often you have redundancy. They say that the top three DNS providers - Cloudflare at 24%; AWS at half that, at 12%; and GoDaddy at a third of that, at 4% - are the single DNS providers of around 38% of the top 100,000 sites in the Alexa ranking. And in addition, four DNS providers are the lone critical providers for more than half of the Alexa Top 100. So four providers used by the Top 100. So any intentional attack, or perhaps the occasional accidental network hardware or software failure at one of these three providers, could bring down a large chunk of the Internet.

And the researchers also observed that when the much broader Alexa Top 100,000 sites are examined to reveal an apparently much broader base of 10,000 DNS providers - so naturally when you look across a much larger cross-section, then you're going to see 10,000 DNS providers. But most of those, 10,000 DNS providers, still have indirect dependencies back to only a handful of top tier providers: Cloudflare, AWS, GoDaddy, Namecheap, or Oracle - which as we know was formerly Dyn - and a few others. So for what it's worth, forewarned is forearmed.

And being aware of this is the point. And after that it's a judgment call. If anyone is in a truly mission-critical operation, I would argue that the added overhead of maintaining fully separate redundant DNS services is probably justified. Just if nothing else, choose two different big guys to host your common DNS. And you know, the amazing thing about the design of the DNS system, back from day one, is that it will automatically and transparently find and use a redundant DNS server. It's like, that part works really well. I've spent a lot of time looking at DNS. The whole DNS spoofability test basically induces the user's computer to send out lots of DNS queries. And so I've watched how the DNS server that was most recently used receives a single DNS query. If it doesn't respond within a relatively short time, all the DNS servers that are configured on a system are then sent a copy of that DNS query.

So, I mean, if there's like any stutter in the primary server, the secondary one, I mean, it responds, the system deals with it so quickly that the end-user never has any idea. So that's of course over on the user side. Over on the corporate side, you want your sites to be hosted by well-separated DNS so that an outage in either still allows the whole system to continue. And again, if it doesn't, it's like, if for you, if you were down during that DynDNS day, and all it meant was an unscheduled vacation, then okay.

Leo: You're not mission-critical, clearly.

Steve: Yeah. Maybe you don't care. But if you're having a boss who absolutely wants you never to be down, it's just not that big a deal to set up a second DNS really somewhere else because the system beautifully will really use it.

Leo: I'm pretty sure that's what we do here. But I'd have to look more deeply. We use DNS Simple, but I thing we also use AWS. So we get both.

Steve: I don't. Both of my servers are, you know, I'm at Level 3. And I don't know geographically if they're separate. But I'm using two Level 3 servers, so I'm not taking my own advice. On the other hand, if I had to have an unscheduled vacation day, okay.

Leo: So it was funny to hear you talk about Alexa as the website ranking company. I completely forgot about them.

Steve: Yeah, they're still around, yeah.

Leo: They're still around. And for a while I think I kind of didn't really trust their results because I think they were basing their measurements on you running an Alexa browser extension, that kind of thing. But they've, I think, come a long way since then.

Steve: I remember in the very early days it was Mark Thompson who put me onto them. And we used to compare our site rankings back in the day. Now it's like, oh, you know, whatever.

Leo: Who cares. Yeah, remember that was a big deal? And you'd get, you know, you'd have a little button you'd put, the Top 5% of the Web or whatever it was, on the bottom.

Steve: Yeah. And of course there were only, you know, 27 websites back then.

Leo: It was easy to do then, yeah.

Steve: So, you know, not a big deal.

Leo: Amazon bought them, and I guess that's where they got the name for their Echo Assistant. They figured, well, we already own it, so let's just use that. We don't have to worry about copyright. So yeah, it's Amazon now. Which means it probably is - they have a pretty good handle on web traffic, I would imagine.

Steve: Yeah, you know, all indications are Amazon is coming on really strong.

Leo: Yeah.

Steve: With their cloud offerings.

Leo: Yeah. They own the market; you know? I mean, Google and Microsoft and Apple and others are trying to get in that market, but there's nobody bigger than Amazon. They're really dominant.

Mr. Gibson, once again, a fascinating look at the way things work on this thing, this island we call technology.

Steve: And how they break when they don't.

Leo: And how they break, which is a big part of it.

Steve: Or they don't when they break or something.

Leo: Yeah. They don't work when they break, and they break when they don't work. You'll find Steve's work at GRC.com. That's his website, GRC, the Gibson Research Corporation. And he has lots of great stuff there, including his bread and butter, his one paid thing, which is SpinRite, the world's best hard drive maintenance and recovery utility. And now I'm going to say the world's best SSD maintenance and recovery utility because you got some benefit just by being a hard drive tool. But now with these new features, this is really exciting.

Steve: Yeah. You and I are going to look at the output of the benchmark.

Leo: Big breakthrough.

Steve: You're able to look at just the five numbers. Or you can ask for four levels of granularity where it breaks the - it does a 1GB transfer, and it breaks it down into individual transfers within the whole transfer. And it just, really, our listeners are going to go nuts.

Leo: I think you're going to have to change the name, though, to something like Doesn't SpinRite or something, because there's no spinning.

Steve: Yeah, SitRite.

Leo: SitRite.

Steve: SquatRite. I don't know.

Leo: SquatRite, no. I'll veto that one right off the bat.

Steve: No, it's not good.

Leo: It's still called SpinRite for historical reasons.

Steve: It's going to be SpinRite. It's just got too much established rep.

Leo: Oh, and but I think you want people to understand that now it does something for SSDs that you just couldn't do. I mean, this is really exciting. I'm thrilled. Find out more. Buy it. Why not? 6.0 is there.

Steve: It'll run on Macs, too, by the way.

Leo: You figured that one out.

Steve: Yeah.

Leo: Recently?

Steve: Well, it's got to be a Mac that still has the, what is it, the Boot Camp.

Leo: Oh, yeah, that's, yeah, so you...

Steve: And it's going to - oh, and also only Intel.

Leo: Yeah, you're going to have a whole new world of Macs out there that you're going to have to figure out.

Steve: Yeah.

Leo: Well, that's 6.3. Or 2. Or 7.

Steve: Yeah.

Leo: 6.0 is out now. You'll be getting 6.1 automatically. And you can participate in its development, as well: GRC.com. That's also where you'll find 16Kb audio versions of this show. You'll also find handwritten transcripts by Elaine Farris so you can read along as you listen. And he's got the 64Kb audio for those of you with sensitive ears. That's GRC.com.

We have 64Kb audio plus video at our website, TWiT.tv/sn. There's a YouTube channel devoted to Security Now!. You can watch there. Best thing, subscribe. That way you'll get it automatically, the minute it's available. You won't miss an episode. This is one show you do not want to miss an episode.

We will be back as we always are, Tuesday. Thanks to Jason Howell for filling in for me last week. He did a great job. I'll be back next week with Steve. We have our "Best Of" coming up later in the month. That's going to be a lot of fun.

Steve: And I'm going to be on the Christmas Special with you guys.

Leo: Oh, I forgot, yes. Yes, we decided to do OG Twits this year. You know, normally we would like to fly you out and all that. We obviously aren't going to be doing that. But it's going to be the OG Twits on our TWiT holiday special. That'll be a lot of fun. I can't wait for that.

Steve: That's going to be a great group - me, Paul, and Jeff.

Leo: Yeah, on the 20th. So make sure you make some time for that one. Okay. We do this show at 1:30 Pacific, 4:30 Eastern, 21:30 UTC every Tuesday. You can watch live, and I hope you will. Thank you, Steve.

Steve: Ciao.