ProtonMail

Description: This week we discuss "Drupalgeddon2," Cloudflare's new DNS offering, a reminder about GRC's DNS Benchmark, Microsoft's Meltdown meltdown, the persistent iOS QR code flaw and its long-awaited v11.3 update, another VPN user IP leak, more bug bounty news, an ill-fated-seeming new email initiative, free electricity, a policy change at Google's Chrome Store, another "please change your passwords" after another website breach, a bit of miscellany, a heartwarming SpinRite report, some closing-the-loop feedback from our terrific listeners, and a closer look at the Swiss encrypted ProtonMail service.

High quality (64 kbps) mp3 audio file URL: http://media.GRC.com/sn/SN-657.mp3
Quarter size (16 kbps) mp3 audio file URL: http://media.GRC.com/sn/sn-657-lq.mp3

SHOW TEASE: It's time for Security Now!. Steve Gibson's here. We're going to talk about 1.1.1.1, Steve's DNS Benchmark, and which DNS servers are the fastest to use. He's also got a review of the encrypted email system from Switzerland, ProtonMail. It's all coming up next on Security Now!.

Leo Laporte: This is Security Now! with Steve Gibson, Episode 657, recorded April 3rd, 2018: ProtonMail.

It's time for Security Now!, the show where we cover your security and privacy online as we have done for more than a decade. Wow.

Steve Gibson: My god.

Leo: My god.

Steve: We're in our 12th year, in fact.

Leo: It's all thanks to this guy right here, Mr. Steve Gibson of GRC.com. Hi, Steve.

Steve: You know, for the first time ever I'm able to do my wave salute without worrying about bumping into the microphone.

Leo: You switched sides.

Steve: Well, so to speak. The landscapers decided now would be a great time to grind up some trees.

Leo: Oh, man.

Steve: So I listened to it. They went away for a while, and I thought, oh, maybe they're done. But I think that was just lunch break, and they're back with a vengeance. So the microphone is so directional that I thought, okay, fine. It's a pain in the butt to move it over; but, on the other hand, I can now do my salute without having to bump the microphone.

Leo: And I don't hear a thing, so good job.

Steve: Yeah, great. So ProtonMail came up a couple times. First of all, a lot of our listeners are wondering about secure email. Now, I've always regarded that phrase as an inherent oxymoron, and so I've never really just focused on it much. But I know it's what people want. And a question from a listener sort of reminded me of that question. It was about, if I were to leave Google, where would I go, you know, leave Gmail. And then, coincidentally, ProtonMail announced an upgrade to their mobile clients. And I thought, okay, fine. Let's talk about what they've done because they've arguably done the best job they can, given the inherent limitations of email. And they're a thousand meters below solid rock in Switzerland, so that's a story we have to tell.

Leo: That's the kind of thing, though, that you often would mock; right?

Steve: Yeah.

Leo: Like that doesn't have anything to do with security.

Steve: The backhoe can still cut the fibers.

Leo: Yeah, exactly.

Steve: Exactly. But we're going to talk about a bunch of things before we get to talk about ProtonMail. We have now the information that was pending last week on what has been named Drupalgeddon2, as in Armageddon, Drupalgeddon, which arrived the day after last week's podcast. So we have the information about that. We have a very interesting and I think significant new DNS offering from the oft-mentioned Cloudflare people, who are just doing a bang-up job with all kinds of stuff. I mean, they're really making a name for themselves. And GRC's own DNS Benchmark confirms their claims.

So later down in the show notes is a screenshot of my just having run GRC's Benchmark, and I'm going to show people - well, actually it's pretty simple, but there's a button you press if you want to add DNS servers to the Benchmark. So people are going to want to add 1.1.1.1, and not so much 1.0.0.1. We'll talk about that. Then we've got to talk about Microsoft's Meltdown meltdown, the persistent iOS QR code scanner flaw that persisted even across the much-anticipated version 11.3 update to iOS, which fixed a bunch of things.

There's another VPN user IP leak, some more bug bounty news which is kind of interesting in a different direction, a very ill-fated-seeming new email initiative that we need to talk about just because it may be coming to a client near you, the consequences of free electricity, a policy change at Google's Chrome Store, another "please change your passwords" after another website breach, a bit of miscellany, a heartwarming SpinRite report, some closing-the-loop feedback from our terrific listeners, and then we're going to talk about, brought to you from Switzerland, an encrypted email service that may be worth looking at if Gmail is chafing for one reason or another. Although I made a comment last week about them looking and reading everyone's email, and on a different podcast I realized, oh, that's old news.

Leo: Yeah, I heard you say that on Know How, and I meant to tell you that, yeah.

Steve: Ah, that's where it was. Right, right, right.

Leo: Yeah, they stopped doing that.

Steve: Well, good.

Leo: Not that they're not spying on you.

Steve: Not that there still might not be a reason to increase your security.

Leo: And your ISP's spying on you, for sure, and et cetera, et cetera, et cetera.

Steve: Yup.

Leo: All right. I can't wait. Actually, I've been looking at ProtonMail. My preference, my strong preference is just, if I want secure email, to use PGP or GNU Privacy Guard and encrypt the mail, and that way I don't have to worry, but metadata leaks. And as we know, metadata is valuable, too.

Steve: Yup.

Leo: So, good. I look forward to this. I have a picture.

Steve: So our Picture of the Week, having just jumped up and down and congratulated Cloudflare, I don't know where they get these numbers.

Leo: Fred Flintstone, obviously [referring to yabba-dabba-do in the background].

Steve: Yeah. So they're showing 1.1.1.1.

Leo: That's their new service, which they launched on April Fool's Day.

Steve: Their new service.

Leo: Which confused everybody.

Steve: It did, especially because it's 4/1, and it's four ones.

Leo: That's what CEO Matthew Prince said. This is exactly the perfect time to launch it, on 4/1; right? Even though...

Steve: Yeah. So the good news is it's not a joke. However, okay, so they're showing 14.8ms for their DNS lookup. They're showing OpenDNS at 20.6, Google's Public DNS at 34.7, and the average ISP - and I don't know what ISP is at 68.23. We'll get to this in a minute. But I have to say I'm really sort of tickled that, if you google for the term "DNS Benchmark" and look at images, it's just a screen full of screenshots of GRC's Benchmark. So I ran it a few minutes ago in order to make a screenshot for the show notes. And my ISP, right, it makes sense, is the closest DNS server to me. I mean, any packets going to any other DNS server have to go by my ISP first, sort of by definition.

Well, it was 8ms for four of my Cox servers. Cox is my ISP here in Southern California. 1.1.1.1, to their credit, and I fully intend to give them credit, was 1ms longer. It was the fastest non-Cox DNS server of all. I mean, and I've got OpenDNS, and I've got Google Public DNS, and a whole bunch of other ones are in there. So it is absolutely the fastest alternative to any DNS server that's not your ISP's. But I don't know whose ISP takes 68ms to respond. I just can't explain that. But not mine.

So the good news is, for Windows users - and it runs under Wine, I think, too, so non-Windows users who have access to Wine should be able to run GRC's DNS Benchmark and test this themselves. And I would suggest everyone does because, in my case, the performance penalty is, well, it is measurable, but it is arguably not worth sacrificing the benefits which we will be talking about when we talk about Cloudflare's offering. So anyway, the Picture of the Week was what is being plastered everywhere. And it's like, eh, okay.

The good news is no one has to believe this bar chart with numbers that are strange. I mean, Google's Public DNS wasn't that much slower, nor was OpenDNS. They were slower, but not, like, ouch. So anyway, people can find out for themselves, and everyone should because your mileage will vary. And, boy, if your ISP is taking 68ms to respond to a single UDP packet, if you have any choice, you might consider what else they're slow

at.

So, okay. Last Wednesday the big news that had had a week's notice given landed, which was there is a very bad remote code execution vulnerability in Drupal that dates all the way back to version 6.x branch, which was discontinued more than two years ago. So this is so bad, and apparently there are people still using something on the 6.x branch, that they offered updates even back there. So that suggests that this has been a very longstanding problem. And the other thing this suggests, unfortunately, is that there are going to be a lot of problems with Drupal sites that are not getting their email, that are not patching themselves, that have just been set up by some techie who wandered off, and people are using the site and don't realize that they're now a target because they can be found.

Last Wednesday, after the patches were announced, Drupal took their own site down after the release of the updates so that it, too, could be updated. They didn't want to do it beforehand, much as they may have wished to, because it may have been possible for people to figure out what was going on. And of course this is the problem is that this is a big target. As we know, they are the number two content management platform on the Internet, with a ton of installations.

So Drupalgeddon1 was a super severe bug. I think it was rated 25 out of 25. It was an SQL injection bug in 2014. So nothing really that bad has happened until now, thus giving this one Drupalgeddon2's name. It's an input validation issue where invalid query parameters can be passed into Drupal web pages. No proof-of-concept code was released because of course this isn't some third party that found it. The bug was found by Drupal's own security people. A guy named Greg Knaddison, who's a security team member, found it and explained that it has to do with the way Drupal interprets a value that begins with a hash as having a special meaning. And so they're trying to promote the idea that - they're working, of course, to downplay the severity of this, saying that hundreds of thousands of sites immediately patched within the first 12 hours of the release.

On the other hand, an independent firm, SiteLock, did some measurements of their own and determined that only 18% of Drupal websites were found to be running the latest core updates. So this suggests that, as we keep seeing with these kinds of problems, the vast majority of websites running Drupal are likely to be vulnerable to compromise because they're not being updated with the latest security patches. So I'm glad that everybody listening to this podcast knows; and, if you have any influence over local Drupal sites, that you immediately update it. We know, Leo, that your TWiT engineers were on top of it from the get-go.

Leo: Yup. Both sites patched like at noon, right after the patch came out, yeah.

Steve: Nice, nice. So I'm afraid that we will be covering exploitation of servers. And we know what's going to happen. They're not going to dig into anyone's network. They're just going to set up crypto miners because that's the thing now. Everyone wants to run cryptocurrency mining on servers, and I wouldn't be surprised if the hackers are rubbing their hands together because I think there are about a million Drupal sites. So they're probably rubbing their hands together saying, hey, there may be some three quarters of a million servers available for exploitation, if we can find them. And unfortunately, as a company like SiteLock demonstrates, it's easy to determine if a site is Drupal-based, based on the response to queries from the web. So I'm afraid we're going to be covering the downside of this.

Okay. So Cloudflare. The IP is wonderful, of course, 1.1.1.1. Hard to forget that one. They also offer 1.0.0.1, but everyone should take note that that is significantly slower. It's meant to be a backup, and so you want to make sure 1.0.0.1 is your secondary DNS IP. And at the moment you might even consider using your own ISP's DNS servers as backup, with 1.1.1.1 as your primary. And the logic that IP resolvers use is to first try the primary resolver that you've registered. In this case it'd be 1.1.1.1. And if it fails, typically then the rest of them that are set up as backups are simultaneously queried if the primary doesn't respond. GRC's DNS Benchmark also verifies reliability, and there wasn't a single query missed in the - I've forgotten now how many I do. Probably a hundred. So reliability was right up there.

So the news is, and we've sort of been stepping all over it already, is that Cloudflare has decided that they want to jump into sort of the growing DNS provider space. We've talked about others that have. Of course Google is, OpenDNS has been there, and Cisco purchased OpenDNS. And we actually talked about this before the Cloudflare announcement in the context of Mozilla last week because remember that I was talking about how the next version of Firefox - we're at 59 right now. When we get to 60, six zero, Firefox will be beta testing their connection with Cloudflare as the de facto DoH, D-O-H, DNS over HTTPS. And this is exciting because this gives us security and privacy for DNS, and probably no more speed than we get with UDP because you just can't get any faster than a single packet being sent and a packet coming back. TCP has to do the same thing.

However, DNS is not encrypted and is not authenticated. And so if we run DNS with the same single-packet query inside of a TLS encrypted and authenticated tunnel to a provider, then we get all of the benefits and no reduction in performance. So Cloudflare at this IP 1.1.1.1 - and anybody who's interested can actually go to that domain with a web browser, https://1.1.1.1. Put that IP in, and that will resolve to their announcement and configuration and setup instructions. And so it's meant to be user friendly. So you can actually put https://1.1.1.1 into your web browser and go take a look at the service.

So they have, as I mentioned, two resolver IPs - that one, the quad one, and also 1.0.0.1 - which when you put both of those in, add those to GRC's DNS Benchmark, it automatically gets the IPs that are configured for your system, presumably from your ISP or OpenDNS or whatever provider. So it adds those to its large list of DNS servers. You can then manually add these. And I'm thinking it's probably time for me to rev the software. I haven't touched it in eight years because it works. There's no bugs. So maybe it's time to add 1.1.1.1 and 1.0.0.1, just to have them built in.

But anyway, Olafur Gudmundsson, who's the Director of Engineering at Cloudflare, said of this effort: "Our goals with the public resolver are simple. Cloudflare wants to operate the fastest public resolver on the planet, while raising the standard of privacy protections for users." And I can assert that, yes, theirs is faster than any other non-ISP resolver in my own testing. And I did this several times over the course of the last several days.

He said: "We began talking with browser manufacturers about what they would want from a DNS resolver. One word kept coming up: privacy. Beyond just a commitment not to use browser data to help target ads, they wanted to make sure we would wipe all transaction logs within a week. That was an easy request. In fact, we knew we could go much further. We committed to never writing the query IP address to disk and to wiping all logs within 24 hours."

He says: "The DNS resolver 1.1.1.1 is also supporting privacy-enabled TLS queries on port 853" - the service is DNS over TLS - "so we can keep queries hidden from snooping networks. Furthermore, by offering the experimental DoH (DNS over HTTPS) protocol, we

improve upon privacy and a number of future speedups for end users as browsers and other applications can now mix DNS and HTTPS traffic into one single connection." Okay, now, I have no idea what that means. It's like, what? That needs to be clarified because I don't see how establishing a single connection to port 853, which is DNS over TLS, allows you to pipeline non-DNS traffic. So maybe they've got something more up their sleeve that isn't apparent.

He says also: "With aggressive negative caching" - and I'll explain that in a second - "as described in RFC 8198, we can further decrease the load on the global DNS system." Negative caching is the process of remembering DNS query failures in addition to DNS query successes. So it's obvious that DNS caching says, oh, look, here's something that people are asking for. Let's keep it locally in our cache after having resolved it once through what may have been a recursive DNS lookup. And as we know, DNS queries have an expiration as part of them, so that permits someone who receives it fresh from the authoritative server to hold it in their cache until its own self-stated expiration times out, in which case they need to go get it again.

Well, negative caching says, well, someone asked us for a domain name that doesn't exist. We've looked for it. Couldn't find it. So let's also remember that. That's negative caching, where you cache a negative outcome of a search - again, just to speed up the return of an NXDOMAIN, which says, sorry, that doesn't exist. And that's, you know, I'm sure that Cloudflare would never consider taking us to some advertising page, but they do properly return an NXDOMAIN. We'll note that some ISP DNS doesn't, that is, ISPs often use a failed lookup to take you to their own search page, where they do a little marketing, which really breaks the definition of DNS and has been controversial in the past.

So he says: "This technique first tries to use the existing resolver's negative cache, which keeps negative, non-existent information around for a period of time." He says: "For zones signed with DNSSEC" - and that's another thing that they are supporting which is very cool, I'll get to it in a second - "and from the NSEC records in cache, the resolver can figure out if the requested name does not exist without doing any further querying."

He also says: "We use DNSSEC validation when possible, as that allows us to be sure the answers are accurate and untampered with. The cost of signature verifications is low, and the potential savings we get from aggressive negative caching more than make up for that. We want our users to trust the answers we give out, and thus perform all possible checks to avoid giving bad answers to the clients."

He says: "However, DNSSEC is very unforgiving. Errors in DNSSEC configuration by authoritative DNS operators can make such misconfigured domains unresolvable. To work around this problem, Cloudflare will configure negative trust anchors on domains with detected and vetted DNSSEC errors and remove them once the configuration is rectified by the authoritative operators. This limits the impact of broken DNSSEC domains by temporarily disabling DNSSEC validation for a specific misconfigured domain, restoring access to end consumers."

So essentially what this means is they're doing all of the work and heavy lifting for us. Remember that, for the time being, until browsers start doing DNS over TLS, we're using UDP. So we have an unsecured, non-authenticated query that we're making, if you were to configure it to 1.1.1.1, to Cloudflare. But with Firefox 60 - and I meant to go look at Chrome to see whether Chrome can already have this turned on because certainly Google is offering this service, as I mentioned last week.

So I want to make it clear that, at the moment, some configuration is required in order

to switch from UDP to TLS. When you do that, then you're establishing an authenticated TLS connection to the DNS provider, in this case Cloudflare. So your queries are encrypted. You get privacy. You get the authentication of TLS. And Cloudflare is a DNSSEC-aware resolver so, to the degree that domains are signed, then they'll verify the signatures, making sure that what they're getting is not spoofed. And because you've got TLS for the so-called "last mile," your query to them, that's both private and unspoofable. So this is sort of a nice and unsuspected compromise on requiring all endpoints like us to do DNSSEC ourselves, at least for our browsers, and not yet for our operating systems, but that may be coming.

But our browsers can be smart enough to do their own resolution through an encrypted tunnel and really tighten up the DNS system which, as we know, as I said last week, is still sort of the one thing hanging out there. In the show notes here I have a screenshot of my having run the DNS Benchmark around 8:30 this morning. And the first four entries are for Cox Communications. And sure enough, they are the four fastest resolvers. The fifth entry says MegaPath Networks Inc., and it shows the IP 1.1.1.1.

So if anyone is interested, if you just google "DNS Benchmark," the first handful of links are to GRC. When you run the Benchmark, since it isn't yet - I haven't touched the code in eight years, since 2010 is when I finished the work on that. When you run the Benchmark, you'll need to add 1.1.1.1 and 1.0.0.1 to the existing list of DNS resolvers. Well, on the Nameservers tab on the upper left is an Add/Remove button for adding and removing nameservers. And so it's as simple as clicking that. Up pops a dialog that prompts you for the IPs of resolvers you want to add or remove, and you can do so right there. So add 1.1.1.1, 1.0.0.1, and then just click Run Benchmark. It'll start spinning and showing you results.

And what was interesting is, first of all, it's not clear, but there's an insider tip. You can left-click in the bar chart, and it will show you actual numerical results. I don't remember whether you can do it while it's running. You probably can. Everything works. I put a lot of time into this Benchmark. But certainly when it's done you can left-click on the bars and then cruise around, and a little floater that tracks where you left-click will show you the actual timings of all of those. That's how I know that the first four, the Cox Communications resolvers, were all responding in 8ms, and the next fastest one was 9ms.

What's interesting is that one, two, three, four, five, six, seven, eight, nine, 10, 11, 12, 13, 14, 15 entries down, below Quad 9, below both of Google's resolvers, below the OpenDNS resolvers, below DSL Extreme and a couple stray Cox Communications resolvers, like way further down the list is 1.0.0.1. So that one is, I mean, it's still quick. It's nothing like 68ms. I don't know where they got that. But it is a few milliseconds slower, and definitely slower than a whole bunch of others that we're familiar with, the OpenDNS resolvers, Google's, and so forth. So 1.1.1.1 is absolutely, for me, next fastest from all of my ISP's resolvers. But 1.0.0.1 is well down the list, which I imagine you're going to want to see for yourself, and the DNS Benchmark allows you to get those results. And if you poke around in the UI, there's all kinds of other little goodies in the DNS Benchmark.

Okay. One more story, and then we'll take our second break: the Windows Meltdown meltdown patch patch. A Swedish security researcher, Ulf Frisk (U-L-F Frisk), who's also the developer of a tool known as PCILeech, which is a kernel memory hacking tool that you can find on GitHub, discovered that Microsoft's January, that is, beginning of the year 2018 patch for the Meltdown fix for Windows 7 64-bit - that's all, the 64-bit version of Windows 7, not 8.1, not Windows 10, just that one - mistakenly flipped a permission bit in the page table management. It was supposed to be set to Supervisor Mode Only.

Somehow it got flipped to User Mode. As a consequence of the work he was doing on his PCILeech tool, he quickly discovered that, until last Friday, so end of March, so January-February-March, for three months Windows 7 64-bit memory, all of it, was completely readable and writable to all user-mode applications. Whoops.

So Microsoft quickly added, pushed out an out-of-cycle emergency update late last week to flip this wayward kernel memory access control permission bit back where it was supposed to be. The flaw was not remotely exploitable. It was just about internal OS memory management. So any attacker would have needed, first of all, to know about it, and then to have physical access and/or some malware installed. But it was devastating because, I mean, it completely blew away all of the OS protection, essentially making absolutely all of the system's virtual memory readable and writable to any user mode program running on that system.

So you want to make sure, now that I've said this, and now that everybody knows about this, I mean, this is, like, instantaneously exploitable. So if you're running Windows 7/64, and that's of course Service Pack 1, and Server 2008 R2 - which is the same OS basically, the server stuff is enabled in it - you want to absolutely make sure that you're all patched up to date, which will then include KB4100480. If you're curious, go into Updates and just make sure that you're seeing KB4100480 has been installed, and then you know you're okay.

Now, you can also test it, if you're interested. If you just Google PCILeech (L-E-E-C-H), you'll get taken to Ulf's page over on GitHub, where you can play with this tool he's got. He warns that some instability may be created in the system, which could bluescreen, in the case of your system being vulnerable. So save everything first. And if you're interested, you can run PCILeech just to verify that that system is in fact not vulnerable. And I would not recommend anyone do that on a live server because you could take it down by mistake. So anyway, for three months all of Windows 7 64-bit OS memory was readable/writable by all applications running on the system, and that got fixed late last week. Whew.

So speaking of bugs and patches, there is a much-covered iOS flaw in the QR code scanner which was added to the normal built-in iOS camera app. It's kind of a neat thing that we got with the jump to iOS 11, such that just using the camera app rather than needing a third-party QR code scanning app, if the camera sees a chunk of real estate which parses out to be a valid QR code, it says, "Oh, there's a QR code," and drops a little message down on the screen saying, you know, "Tap to go to this site."

Well, just before Christmas, on December 23rd of 2017, so just over three months ago, it came to a German security researcher's attention, a guy by the name of Roman Mueller, that there was a way to spoof these URLs. That is, you could create a QR code which showed one domain name on the screen, but took you to a different domain name, which is not good. And 90 days later this problem is still unpatched.

So Roman published his findings. He gave Apple three months to fix this. And I tried it yesterday on my newly updated version 11.3, and it still works, that is, it's still broken. Using a URL that is present in a lot of the coverage of this online, so anybody can find it, and I have the URL here in the show notes, when you scan this sample QR code, it says you're going to be going to Facebook.com, clear as it could be. And when you tap it, it takes you to infosec.rmit.de, which is Roman's demo page, saying nope.

Now, the problem, of course, is this could be a malicious page, spoofing Facebook, and no one would be the wiser. So I'm at a complete loss to understand how Apple could have let this be out there for 90 days. Roman kept quiet. He let everybody know after 90

days. And if you google "iOS QR code bug" there's a huge amount of coverage in the press. So let's hope that this shines a spotlight on it that apparently - I can't imagine what Apple is thinking that they haven't fixed this. This is a bad problem, and essentially it means that something is parsing the QR code to get this full URL, but the code that displays where you're going to go is different than the code which takes you there.

So I can't understand how Apple has not fixed this, but they haven't, and now the world knows about it. So now it can start getting exploited because there's no mystery anymore. This is trivial to do, and I wouldn't be surprised if bad guys are not thinking, how can we fool iOS users with this? So let's hope that Apple gets this thing fixed quickly now that they seem to have not given it the attention that I would argue it needs.

And speaking of 11.3, I've spoken of it, and I've been waiting for it for months because, as we've talked about, I was very disappointed when I think it was the very last version of 10 and then all of the 11s just completely brought my 6s to its knees, my iPhone 6s, because they adopted this ridiculous, oh, we're going to slow your phone down because we're worried that the battery's not strong enough. And of course I had alleged that I was taking really good care of my battery, and I'm sure it was in good shape.

Sure enough, what we have now in 11.3 is that the throttling has been removed until the phone actually experiences a problem with the battery, which seems like the right policy. But there's also a battery health meter. And as I expected, my very old iPhone 6s is 100% health of its battery because I take really good care of my battery. So it was always wrong that Apple, I mean, basically I got an iPhone 10 out of frustration because they had rendered my iPhone 6s unusable from a software update, which I think is unconscionable. So they got my money, and now I've got an iPhone 6s which runs just like it used to, just perfectly wonderful, with an even bigger screen than my 10.

So anyway, they did fix, with 11.3, 43 known security vulnerabilities, about half of them in WebKit. So it's definitely worth updating. I have maybe 10 iOS devices. I've got two phones, and I've got pads everywhere. Not a single one of them updated themselves or notified me of an update. It wasn't until I read that 11.3 was available that I then went into the control panel and, like, oh, yeah, there's my little red "1" saying you've got a notification. So anyway, now everything's up to date. But for what it's worth, for those who, like me, don't seem to have their devices automatically updating in any timely fashion - because this was pushed out, I think, the middle of last week - you may want to go check. You'll probably find that, if you're not at 11.3, your device is willing to go to 11.3. You've just got to give it a little bit of a nudge in order to get it unstuck.

We have another VPN client IP leak. There's a well-known leakage problem of IP addresses, and we've talked about this for years. The WebRTC function inherently leaks the IP of the client as part of the WebRTC functionality, which is normally not that big a problem because, after all, any site that you're going to knows your IP anyway. But in the case of a VPN user, you are often, as we were discussing last week, wanting your IP to be kept to yourself. That is, you want any site you're visiting to see the IP servers, or the IP endpoint IP, as a proxy for yours, but not yours. Well-designed VPNs have filtered and do filter the WebRTC IP leakage so that that doesn't happen.

But a researcher, Paolo Stagno, who goes by the handle "VoidSec," audited 83 VPN apps - although, having looked them over, the definition is a little loose because he's got various non-VPN proxies and things in there, too - but 83 things that can put themselves between your browser and you, and most of them are VPNs, 83 of them, to see whether they were properly anonymizing their users by filtering this longstanding WebRTC IP leak. He found that, of those 83, 17 VPNs were still leaking their users' IP addresses. And again, the definition here is a little loose for VPN. But I checked, I scanned through the

list to see if there were any important ones. Hotspot Shield, which we have spoken of often, does not leak. And I remember you chuckling over this one, Leo, when I mentioned it before, HideMyAss VPN does leak.

Leo: Oh, well, doesn't hide it that well, yeah.

Steve: So your ass is not as well hidden as the name of the VPN service would suggest. The cool thing is, because he doesn't have the budget or the logistics for checking all VPNs, what he did instead is to create a free service that allows all VPN users to figure out whether their particular VPN is blocking this WebRTC leakage. The site is https://ip.voidsec.com. So ip.voidsec.com. That will just present a nice little page showing you the IP address that you're connecting from. And this, as I was saying before, any website you go to has an IP from which you are connecting.

And, by the way, you can do this right now. You don't need - if you're just curious, don't use a VPN and so you can see what it looks like. You can see if your web browser supports WebRTC. When you go to ip.voidsec.com, you'll probably see a listing of several IPs which are being presented through the WebRTC interface. And then he does have, is maintaining - and Leo, you've got it on the screen right now - a Google doc showing all of the spreadsheets that he has tested, and many that other people are reporting yea and nay for, which is how I know that HideMyAss doesn't. So ip.voidsec.com, if you're a VPN user, absolutely go there. Make sure that the WebRTC interface is not leaking; and, if it is, go screaming to your VPN provider because that's, well, if you're using HideMyAss then we already know that you've got some leakage problem which really is not what you want.

Anyway, so more on the bug bounty front, this time the federal government. As I've said, as we've been talking about this recently, I think this is a new career opportunity for people who want to work on their own. And if you're good enough, these bounties are large enough that you can probably support yourself. So recently, as of Sunday, Easter Sunday, April 1st, the DoD, the U.S. Department of Defense has bellied up to the bar with their own bug bounty program.

Well, actually, they're offering a bug bounty program through HackerOne, which is an organization which is sort of the clearinghouse for enterprises that want to offer, but have somebody else manage, a bug bounty. And I would imagine they probably get a piece of the action. HackerOne is able to count among their customers General Motors, Lufthansa, Starbucks coffee, the EU, Spotify, Airbnb, Lending Club, Nintendo, WordPress, Twitter, Shopify, Ghostery, Google Play, and Cylance. So some big players. It's HackerOne.com.

Jack Messer, who's the project lead at Defense Manpower Data Center with the DoD, he said: "The Department of Defense enterprise system" - which I guess is their travel, I saw in some other coverage, it was like their own travel system, their travel agency. So I guess the DoD is naturally shipping people all over the place, and so they wanted to make sure that their publicly facing travel management system is not leaking too much.

So he said: "...is relied on by millions of employees for global operations. The DoD has seen tremendous success to date working with hackers to secure our vital systems," he wrote, "and we're looking forward to taking a page from their playbook. We're excited to be working with the global ethical hacker community and the diverse perspectives they bring to the table" - boy, this guy's a bureaucrat - "to continue to secure our critical systems." Okay. "To be eligible to participate in the DoD's bug bounty challenge,

individuals from the public must be United States taxpayers or a citizen of or eligible to work in the U.K., Canada, Australia, or New Zealand. U.S. government active military members and contractor personnel are also eligible to participate, but not eligible for financial rewards."

Okay. So I dug in a little bit, looking into the history of this, because I was curious. And it turns out that the first instance was two years ago with Hack the Pentagon, where over - and this is a little breathtaking - 3,000 vulnerabilities have been resolved since then. The first Hack the Air Force bug bounty challenge found 207 valid reports, and hackers earned more than $130,000 for their contributions. The second Hack the Air Force resulted in 106 valid vulnerabilities, and $103,883 paid to hackers. Hack the Army in December of 2016 found 118 valid vulnerabilities and paid out $100,000. Hack the Pentagon in May of 2016 resulted in 138 valid vulnerabilities, which were resolved, and they just wrote "tens of thousands" paid to ethical hackers for their efforts. And then Hack the Air Force 2.0, which was another round later, demonstrated continued momentum of the Hack the Pentagon program, which is sort of the overall umbrella, beyond just its first year, and further hardened the attack surface.

So if there are any listeners who have hacking skills, who like to dig into systems, you want to make sure that you do it ethically, that you're not crossing any lines. But it's very clear to me that this concept of bug bounties, where high-profile, well-moneyed interests really have to have their products secured, and there's this dawning awareness that their own engineers are not doing the job for whatever reason. And as I've said, I mean, I understand. It takes other people to find bugs in things. That's sort of the nature of the beast. There's a career opportunity here to make some good money. So something worth considering.

Okay. This just seems like a bad idea. There's something called BIMI, and I don't - I think you would pronounce it "beam me," as in "beam me up," BIMI, Brand Indicators for Message Identification. And there's a lot of smoke around this. There's a site, www.brandindicators.org. There's even something called the Authindicators Working Group. So the members of the Authindicators Working Group include an outfit called Agari that sort of seems to be spearheading this, Comcast, Google, LinkedIn, Microsoft, Oath as in the Verizon subsidiary that operates AOL and Yahoo, PayPal, Return Path, and ValiMail, those last two being email providers.

And so what this group is trying to do I'm worried about. They're trying to come up with a way to attach verified logos to email. The site asks the question rhetorically: "What if you could put your brand on every email, and users could trust it's you?" From the site they say: "Brand Indicators for Message Identification (BIMI) is an industry-wide standards effort that will use brand logos as indicators to help people avoid fraudulent email..."

Leo: Oh, this is absurd. God, this just frustrates me.

Steve: I know, "...while giving marketers a huge new opportunity to put their brands in front of consumers for free."

Leo: Because no phishing scam would ever use your logo.

Steve: I know, I know.

Leo: I mean, they wouldn't dare.

Steve: I know. And so they say: "If you use Yahoo Mail, you will see brand indicator logos from Groupon, Agari, and other large brands in the financial services, aviation, and technology industries." And so just to explain what they're...

Leo: Does it show up somewhere beside the email? I mean, like on the side?

Steve: Yeah. Yeah, the idea would be that - and if you go to the site, Leo, if you go to BrandIndicators.org, you can see the way they're proposing. It would be like, in the same way that our messages show, like, the people's faces, like in iMessage, the idea would be there would be a circle with a company logo next to their email in our inbox. And so they say, under Brand Impression: "Every time you send an email, you are guaranteed your logo will be displayed in the preview pane and near the From line."

Under Trusted Communications, they say: "When users see your logo, they'll trust the email and be more likely to respond to your email and do business with you." And then, under Stop Phishing, they said: "Brand indicators will authenticate the sender of the email using an existing standard, DMARC, before displaying your logo." Except, as you immediately picked up on, Leo, none of that is true.

Leo: Okay.

Steve: So I went over to GitHub to...

Leo: So, wait, it's a win-win situation.

Steve: It's a win-win. Authindicators.github.io. And I'm thinking, oh, there's an RFC. There's like a specification. There's all this thing going on. What's it about? So I scrolled down, after all getting through the boilerplate of how wonderful this is going to be. And under Section 3.2 Security, it says: "Brand indicators are a potential vector for abuse."

Leo: Oh.

Steve: Yeah, no kidding. "BIMI creates a relationship between sending organization and email receiver so that the receiver can display appropriately designated indicators if the sending domain is verified and has meaningful reputation with the receiver. Without verification and reputation, there is no way to prevent a bad actor [and then gives] exxample.com from using example.com's brand indicators and behaving in a malicious manner." Yeah, exactly the problem.

Leo: Well, good. This really worked.

Steve: And then, get this.

Leo: So really this isn't about verification. This is about putting your logo on the email is what this is.

Steve: Yes. And the last line: "This document does not cover these verification and reputation mechanisms, but BIMI requires them to control abuse." In other words, we're going to - won't this be fabulous to have logos in email that cannot be spoofed. We don't know how to do that. But, boy, we've got everybody all revved up.

Leo: By the way, if you get an email from me, it will be signed by me, either using an S/MIME cert or a PGP signature that is verifiably mine, and you will know it's me. And that exists and has existed for more than a decade. Oh, well.

Steve: Yeah.

Leo: It doesn't have the logo in it, though. That's nice.

Steve: I know. I saw some reference to them having approached the CAB Forum, the CA Browser Forum. And I thought, oh, that's kind of interesting. Imagine if the certificate authorities could sign a logo. In that case you could use the same certificate authority system to assert that this logo is associated with this domain. So maybe. But anyway, none of it's there yet. And unfortunately, let's hope none of this happens. I mean, they're already launching this. They're like, oh, they've got specs and protocols, and Yahoo email is doing it now, even though it's completely - none of the antispoofing protection is there. And of course the moment that PoPal shows the PayPal logo in email, it's like, okay, no one can trust that anymore. So, bad idea.

Leo: Bad dog. Bad dog.

Steve: I don't know what they're thinking. Anyway, free electricity. What could possibly go wrong if you have free electricity? Well, cryptocurrency mining might be expected to pop up anywhere you have tech-savvy, well, university students who are not paying their own electric bills. And sure enough, it turns out it's becoming a big problem. I got a kick out of an online posting which was titled "University Found Out I'm Mining Bitcoin in My Dorm." And the posting reads:

"I have free electricity here at ASU, and I decided to use Nice Hash to mine some bitcoin. I've been mining for a month with no problems with a pair of GTX 1080 Ti's," but, you know, some big GPUs. "But I just received an email today that I need to uninstall my mining hardware." And this person writes: "Now, I'm pretty sure this is not based off my electric bill, as I mined with my laptop for a bit at the same time, and they said I had two devices mining." Meaning they probably saw two connections from his dorm room, two MAC addresses with mining protocol. He says: "Now I'm wondering how I can get around this problem. Should I use a VPN, or is there another way to disguise my mining?"

So this was part of a larger story that I got a kick out of that was basically saying that,

unfortunately, universities' security is generally not tiptop, and that university servers are being besieged with cryptocurrency miners, as we might expect. But there's that problem, and then there's a problem that students, who do not have to pay their own electric bill because it's just part of their tuition or whatever, I mean, your individual room is not typically charged for electricity. They're saying, hey, free electricity, let's fire up some GPUs, and we'll be able to heat the room and get some cryptocurrency at the same time. So, yeah, of course. I imagine we will see universities become a little savvy, as apparently this ASU has, and start detecting and probably blocking cryptocurrency mining where possible.

Leo: Does it use a known port? Would you have to do deep packet inspection?

Steve: Yeah, you would need to look at the packets in order to see what was going on. But I imagine universities are doing some good firewalling anyway, just in order to deal with all of the antics.

Leo: Yeah. Can you imagine?

Steve: Tor, oh, my god. Torrents and who knows what. Yeah, it's like, yikes. So meanwhile, Bleeping Computer reported that Google has announced that, effective yesterday, April 2nd, the Chrome Web Store review staff has stopped accepting new extensions, new Chrome browser extensions, that perform cryptocurrency mining, and that existing Chrome extensions that perform cryptocurrency mining will be delisted sometime in late June. So it turns out that, until yesterday, Google was explicitly allowing deliberately clear mining with Chrome, if mining was the extension's sole purpose, so it was registered in the Chrome Store as a currency miner, and if the user was informed in advance that their computer's resources were going to be used for this CPU, basically, hardware-intensive computer-draining task.

James Wagner, who's the Extensions Platform Product Manager for Google, said that, unfortunately, about 90% of all extensions which do mining are attempting to slip that by their user. And they're not complying in one way or another with those policies. So they're not doing user-mediated mining, they're actually doing cryptojacking on the sly. So Google is going to start tracking those things down, and they're not going to accept any more that they know of. And that would allow them, the moment they see that the one got by, to just yank it without - by formally changing their policy.

And if anyone's wondering, remember that Chrome includes its own Task Manager under the More Tools menu item that will show the percentage of CPU being consumed per tab. So Chrome breaks it out by tab. And if something is using up a lot of your system, if Chrome itself is consuming 90% of your system's reserves, you can determine which tab that you've got running is actually doing that. And it may be that you've visited a site that is doing some mining on your browser without your permission.

The Under Armour folks who publish MyFitnessPal, a very popular fitness app that has about 150 million users, lost control of their login information. All of their 150 million users' username, email, and hashed passwords were stolen. Everyone affected has been notified by email. There's no information, however, on how the passwords were hashed. They did say that no credit card information or other information appears to have been accessed, since that's stored in a nonbreached database.

So the protocol is what we've become accustomed to after such breeches. We don't know how strongly protected the hashes were, so we don't know how difficult it will be to reverse the hashes to determine what the passwords were. But certainly, if you are a MyFitnessPal user, and you did not get email from them, change your password. And if it happens that you were not using a unique password at MyFitnessPal, you will want to make sure that, because a bad guy could reverse the password potentially and determine what your plaintext password was and would have your username and email, they might go poking around trying to see where else they can log in with the same information. You want to make sure that, if you've reused that information anywhere else, you get that changed.

So Leo? "Star Trek: Discovery."

Leo: Oh, yeah, you've binged it.

Steve: I did. Lorrie and I...

Leo: Trying to save a buck.

Steve: Lorrie and I, we loved it.

Leo: Oh, good.

Steve: So I just wanted to follow up, for what it's worth. It's on CBS All Access. You can get it for seven days free trial. It's 15 hours, 15 episodes, and absolutely worthwhile. It is, as I mentioned when I talked about it before, it is not Jean-Luc's Prime Directive Federation. It's gritty and sort of adult and really, really good. So I just wanted to follow up and say that we're really happy that we watched the first season, and we'll wait for another season to get done and probably, I would say, this time pay for a month in order to watch it over the course of a few nights. It was really good.

Leo: Good. That's good to know.

Steve: I got really one of those classic perfect notes from a SpinRite user, Dan Martins in Fairbanks. And I don't know if there's any Fairbanks other than Alaska. I was just assuming Alaska. But the subject was "SpinRite did it again." And he said: "Dear Steve, I'm writing to pass along my mother's ecstatic joy and thanks for what SpinRite did for her. I've been listening to you and Leo for years, and I purchased SpinRite mostly to support you and everything you do for us. I've used it from time to time to check on the drives in my systems, and it may have [I think he meant] been helping to keep everything working without any trouble, as we know it can.

"However, my mother's computer did not have SpinRite's maintenance help, and it died without warning. Well, really there was warning, but she did not know to read the signs. She explained that it had been taking longer and longer to start up in the morning, but she heard that computers slowed down, so she figured that was normal. And naturally she had not backed up about three months' worth of work on the sequel to her first

novel. The copy she had on a thumb drive would not read back, and it looked like a complete loss.

"Since I'm my family's go-to computer tech, she called in a panic and explained that she and her husband had tried everything, and they feared the worst. But I knew one thing they had not tried. Her computer has a DVD, so I created and sent her a bootable SpinRite CD. After it arrived, I helped them start SpinRite running and told them to call me back when it was finished. Since I've been listening to you and Leo for years, we all know how this story ends. When she rebooted her computer, everything was back to full speed like it was new; and, more importantly, her nearly finished second novel was completely readable."

Leo: Ooh. Oh, wow. That's huge.

Steve: Yes. "She couldn't believe it, but I knew to expect it. Having had a 'near death' experience, she will be backing up to multiple drives from now on. So a happy Easter was had, and 'thank you' hardly seems like enough. Your avid listener and supporter, Dan." So Dan, wow, thank you so much. And I'm glad SpinRite was able to come to the rescue.

Okay. A little bit of closing the loop. I got a kick out of this one Doug J sent with the subject "COBOL" because remember, what was that, that was the second most dreaded language or something? First was JavaScript, believe it or not, and COBOL was second worst. And so he said: "Hello, Steve. Unfortunately, COBOL is still the old and crumbling foundation for PeopleSoft. I dread it because," he says, "once a year I have to look at it and hope I do not have to make some change, fearing unintended consequences."

And I'd heard of PeopleSoft before, but I forgot what it was. Well, it turns out that it's one of those things Oracle purchased. PeopleSoft, Inc. was a company that provided human resource management systems, financial management solutions, supply chain management, customer relationship management, you know, all those HRMS, FMS, SCM, and CRM junk that you hear about. Oh, and enterprise performance management software. Perfect. I mean, that's what you want to write in COBOL are those sorts of things. "As well as software for manufacturing and student administration to large corporations, governments, and organizations. It was an independent corporation until Oracle purchased it in 2005." So I guess they're going to milk it for whatever revenue it can still generate with its COBOL code, and then maybe just quietly put it out to pasture, who knows.

Anyway, I got a kick out of the fact that my mention last week of COBOL stirred up one of our listeners. And Greg V said, his subject was "Cloudflare's Quad 1s and Privacy." And he asked: "If DNS UDP traffic passes through your ISP unencrypted, how can it be private? Isn't it a simple task for the ISP to track the queries?" Then he signs off, "SpinRite Owner Greg." And of course, as we know, yes, as long as you're using UDP, no privacy, no authentication, man-in-the-middleable, and subject to all kinds of exploitation.

Where we are headed, however, at least initially with our browsers, will be to establish, I imagine, when we launch the browser, the browser will come up and reach out to its preconfigured DNS over TLS provider like Cloudflare or like Google, bring up a tunnel, and then all of the browser's DNS resolution will then go through that tunnel in order to give us all the benefits of security and authentication and the benefit, for example, in the case of Cloudflare, of extremely good performance. So for now we don't have that yet. Quad 1 from Cloudflare over UDP gives us very good speed. And of course, as I covered

before, you can use GRC's Benchmark to find out how it compares.

And, by the way, I didn't say, I would love to hear back from our listeners what other people find using the Benchmark. I'll be happy to share that with our broader listener base if people want to send me or tweet their results from running the GRC Benchmark, having added 1.1.1.1 and 1.0.0.1, and then do the Benchmark.

And, finally, Scott T on the East Coast. His subject was "Secure Email / Transition Away from Gmail." He says: "I'm looking to move away from Google Mail and was wondering what options you recommend besides spinning up one's own server. Or, if you do recommend spinning up your own server, what's the best way to make it as maintenance-free as possible?"

Leo: If you have to add, do not.

Steve: Yes, Leo. Yeah, thank you, perfect.

Leo: Can I recommend FastMail?

Steve: Please.

Leo: FastMail.com is awesome.

Steve: Yes.

Leo: They are very active in the IMAP community. It's the best IMAP implementation I've ever run into. Lots of nice features. It's not free, but I took my email off of Gmail, and I use FastMail entirely. I used to use Gmail for spam filtering, but I don't even feel I need that anymore. They have very good spam filtering. So highly recommend FastMail.com. They're really good. But I'm going to listen to what you say about ProtonMail.

Steve: Well, so, yes. So FastMail.com, it's absolutely something to consider. And I was hoping you were going to chime in and tell us who your provider was.

Leo: Yeah, I love them. Love them.

Steve: So ProtonMail has been sort of floating around. It was, first of all, I get the feeling across the board that the founders' hearts are in the right place. It is, I would say, to every degree possible, encrypted email service. It was founded four years ago, in 2014, by some guys at CERN, the well-known research facility in Switzerland. On the desktop, ProtonMail offers web browser, JavaScript-driven, client-side encryption. So it's browser based on the desktop. And there's JavaScript running there which performs encryption of the contents in a way that they don't get the key. So this is essentially TNO email service. Trust No One email. But of course doing this is not simple because you still

need to be able to transact email with other people, with other ProtonMail users and so forth.

So let me sort of get into some of these details. It's also worth noting that they offer iOS and Android apps. And it was a recent, like last week, update in their mobile offerings that brought them back onto my radar. On the web side, since November of last year they've been encrypting the user's contacts. So the user's key that they don't have is used to encrypt and store the user's contacts on their server. Last week, that feature was added to ProtonMail's iOS and Android apps so that that's available for the first time there, as well.

So ProtonMail is operated by Proton Technologies AG, which is a company based in Geneva with servers in two locations in Switzerland, which of course places both sets of servers outside of U.S. and EU jurisdiction, which may be important for some users. The default account setup is free, and the service is sustained by optional paid or just voluntarily. You can pay for services, or you can just send them some money to support them. And it's been a success. As of a year ago, January 2017, they had over two million users. It was initially a crowdfunded startup. I think it was Indiegogo that launched them four years ago, and they were looking for $100,000, and they instantly got $500,000 and some odd dollars. So it was clearly something that was interesting a lot of users.

So they explain on their site, under Anonymous Email and protecting their users' privacy, they say: "No personal information is required to create your secure email account. By default," they write, "we do not keep any IP logs which can be linked to your anonymous email account. Your privacy comes first." Under Open Source / Free Secure Email they say: "We believe email privacy should be available to all. That's why our code is open source, and basic ProtonMail accounts are always free. You can support the project by donating or upgrading to a paid account." And Easy to Use / Security without the hassle: "ProtonMail can be used on any device without software installed, meaning web-based. ProtonMail's secure email accounts are fully compatible with other email providers. You can send and receive emails normally."

Now, here's where it gets interesting because they've done a lot. And I did get a kick out of, as I mentioned at the top of the show, under Physical Security they mentioned: "ProtonMail's infrastructure resides in Europe's most secure datacenter underneath 1,000 meters of solid rock." So, okay. But what matters to us, of course, is the technology. So ProtonMail uses a combination of public key crypto and symmetric encryption protocols to offer end-to-end encryption. When a user creates a ProtonMail account, their browser generates a public key pair. So of course that's the public key and its matching private key, and those are RSA keys.

It'd be nice, maybe at some point they will evolve to elliptic curve keys, since that would make them much smaller and handier. But for now, RSA keys. And actually they're 4096-bit RSA keys, so they're not messing around. The public key is used to encrypt the user's emails and other user data. The private key capable of decrypting the user's data is symmetrically encrypted with the user's mailbox password. This symmetric encryption happens in the user's browser using AES-256, so as good an encryption algorithm as there is.

Upon account registration, the user is asked to provide a login password for their account, and ProtonMail also offers users to log in with two-password mode that requires a login password and a mailbox password. The login password in that case is used for authentication. The mailbox password encrypts the user's mailbox that contains received emails, contacts, and user information, as well as that private encryption key. So that allows them to store the key for you. But then, for example, if you went to a browser

where you hadn't been before, and you wanted to have full access, you would use both of these authentications, the second one doing a local decryption of your matching private key. So they never have it, but you're able to essentially retrieve it in encrypted form from them.

Then they say: "Upon logging in, the user has to provide both passwords. This is to access the account and the encrypted mailbox and its private encryption key. The decryption takes place client-side, either in a web browser or in one of the mobile apps. The public key and the encrypted private key are both stored on ProtonMail servers. Thus ProtonMail stores decryption keys only in their encrypted form, so ProtonMail developers" - that is, the ProtonMail people - "are unable to retrieve user emails nor reset user mailbox passwords."

So again, with true security comes responsibility. It is TNO. It is Trust No One. So they can't perform recovery for you. You've got to make sure you store these things safely. And so this system, as they wrote it, absolves ProtonMail from storing either the unencrypted data or the mailbox password, divulging the contents of past emails, and decrypting the mailbox if requested or compelled by a court order. In other words, they can't do any of that. They don't have the keys.

ProtonMail exclusively supports HTTPS and uses TLS, and they've got a great TLS. Ephemeral key exchange to encrypt all Internet traffic between users and the ProtonMail servers. 4096-bit RSA SSL certificate was signed by QuoVadis Trust/Link and supports Extended Validation, Certificate Transparency, Public Key Pinning, and Strict Transport Security. So they did all of the right things, checked all the boxes. In fact, they hold an A+ rating from Ivan Ristic's Qualys SSL Labs for their servers. Then a couple years ago they added native support to the web interface and their mobile apps for PGP. So they support Pretty Good Privacy. This allows a user to export their ProtonMail PGP-encoded public key to others outside of ProtonMail, enabling them to use the key for encryption, for mail coming back to those users. And they said that the ProtonMail team plans to support PGP encryption from ProtonMail to outside users.

And then the last bit of coolness that I got a kick out of - because, again, you've got security between your client and them. What about outbound from them? And that's of course the bugaboo with email, unless you know that your intended destination is a PGP user. So they said: "An email sent from one ProtonMail account to another is automatically encrypted with the public key of the recipient."

Remember, they have the public keys. They don't have the private keys. And that's cool because that allows them - so any email incoming is immediately encrypted with the recipient's public key, and they can't decrypt it from that point. And inter-ProtonMail traffic never leaves their servers, and essentially it comes from the user, and it's immediately encrypted with using the public key of its ProtonMail recipient and then waits there for that recipient to pick it up. So, for example, it would be easy if you wanted to set up a little network among a group to all use a free ProtonMail account, and all of your email stays on their servers and is always kept encrypted, and only the recipient is able to decrypt it after it's been transferred into their inbox.

They say: "Once encrypted, only the private key of the recipient can decrypt the email. When the recipient logs in, their mailbox password decrypts their private key to unlock their inbox. And now email sent from ProtonMail to non-ProtonMail email addresses may optionally be sent in plaintext or with end-to-end encryption. With encryption, the email is encrypted with AES under a user-supplied password. The recipient receives a link to the ProtonMail website on which they can enter the password and read the decrypted email. ProtonMail assumes that the sender and the recipient will have previously

exchanged this password through a back channel. Such emails can be set to self-destruct after a period of time."

So that's a really cool solution for wanting to encrypt something to somebody who doesn't themselves - who is not tech savvy, doesn't support, doesn't know how to do PGP or perform their own decryption. So the idea would be you would come up with a password, get it to them through some other channel, fax it to them, carefully dictate it over the phone, whatever, so that they have that. You then are able to send them email which you encrypt with that password. What they get is a web link, a clickable link that takes them back to ProtonMail over HTTPS with TLS, and so complete security. And they enter the password, which on their browser performs symmetric decryption so that even ProtonMail never sees the decrypted email. It's decrypted on the recipient's browser so that they're then able to receive the mail.

So as I said, they thought through essentially every possible scenario to extend the email encryption envelope as far as possible. And I just, you know, given the technology is open source, is auditable, clearly these guys have their hearts in the right places. They don't want to be able to respond to court orders, and they've designed a system whereby they can't. You as the user need to make sure you don't lose your passwords because inherently they are unable to provide them to you.

But in return for that, you get a very nice, Trust No One, free or optionally paid or donation-supported email system based in Switzerland that does allow you to get encrypted content to people who otherwise don't know how to do anything with encryption, with the obligation being to somehow get the decryption password to them so that you're then able to exchange mail. So a very cool solution, and one worth looking at if somebody wants to move away from Gmail. And of course, Leo, your FastMail is just sort of a straightforward email system.

Leo: It's, yeah, not encrypted.

Steve: If somebody doesn't want all of that rigmarole.

Leo: Encryption's up to you. It's sad that, and I understand why, but that GNU Privacy Guard or PGP never really caught on because it solves the problem. It doesn't solve the metadata problem, but it solves the problem of encrypted email, Trust No One and all of that stuff.

Steve: Yup, it does.

Leo: But it's too hard for people to use, and just nobody uses it. S/MIME certificates are easier, but nobody uses that, either.

Steve: Yeah. In fact, I was talking with the Padre on Know How last week, and we talked about that. The way I look at it is if the only reason we have moved from that land where Firesheep was able to acquire people's browser sessions and impersonate them, is that it was all done for the end user. That is, users don't have to do anything with encryption. All they do is, like, oh, look, I have a padlock now, or my key's not broken any longer. Or, oh, look, the URL is in green. I guess that's good. Green means good.

The point is that it was servers that did this, and browsers that did this. End users had to do nothing. And even getting websites to do it was like pulling teeth. We had to make it free so that it didn't cost anything, and it was all automatic, and you just, like, hook up. And, oh, look, your server automatically does Let's Encrypt and off you go. I mean, what we've seen is that we have to drop the bar so low in order to get people to do anything more, that finally we've got that with HTTPS. So that suggests that there's just no way that that was going to happen with email unless, again, unless it was just automatically somehow magically done, and no one had to lift a finger. So what we ended up with is, where it was important, the tools are there to do it and to make it pretty easy.

Leo: It's just it's a measure of how people don't really - as long as people understand, they don't really care. This is a postcard, they don't seem to really care. So the real issue is the key exchange is a difficult thing, and that's something, by the way, PGP has solved beautifully. There are these public key servers. Any PGP or GNU Privacy Guard implementation will have a lookup feature that looks it up on the servers, an ability to post yourself on the servers so there's a directory. If people want to have my PGP key, for instance, they could search for my email address or my name on the servers, download it, and add me. And it's very simple, I mean, it's a system that really works. And I really think that's the problem is just nobody really cares.

Steve: Yeah.

Leo: Yeah. The nice thing about that is, I mean, ProtonMail is not impervious to a warrant from the Swiss authorities. And with the CLOUD Act, it's probably not impervious to U.S. authorities, either, in the long run. But PGP, only you have the key. They'd have to come to you and say unlock it. And at least you'd know.

Steve: True, true.

Leo: At least you'd know. I just don't, you know, I'm glad ProtonMail exists, and I've had an account since 2014. I mean, I signed up early on.

Steve: Well, and for what it's worth, remember, as with between PGP users, between ProtonMail users it's the same. I mean, it's encrypting before it leaves the browser and decrypting only when it is received by the other end. So ProtonMail does offer the same end-to-end encryption, and no attack on their servers legally or physically can reveal the contents.

Leo: A Gmail to Gmail email is encrypted in place and in flight. Only Gmail to Gmail. And it's unclear whether Google has access to the mail. I think they do because they have antispam tools.

Steve: Well, and Gmail is only encrypted over HTTPS in flight.

Leo: Yes, that's right.

Steve: So instead of using a local private key.

Leo: Right. I use a 4096-bit key in PGP. But, I mean, it's not that - there's no point. I don't have, I mean, you know.

Steve: Yeah. In fact, even Padre, who like you has been a PGP user from the beginning, he said, "I think I have maybe three people. I've got, like, three."

Leo: I have a long list of names in my key list. But the entire conversation I've ever had with any of these is, "Uh, is this set up correctly?" "Yes, it is." And that's it.

Steve: Hello, can you hear me?

Leo: It works.

Steve: Yes, I hear you.

Leo: And I do, every month I get a couple of emails from people saying, well, you're the only one I know who uses it, so is it working? And I'll respond, encrypted if I can, back to them, and that's about it. I had one guy, and he stopped, used to send me news articles encrypted, like suggestions for stories to cover, which was fine. I mean...

Steve: Yeah. And I said when I was talking to Padre, it's like, I mean, I just have no need for it. I've never bothered. If I ever needed to send something encrypted, I would write it in Word. I would encrypt it myself.

Leo: Right. You'd use miniLock or something like that.

Steve: Exactly, and then attach it to an email and say, here. Here's a blob. Decrypt it using our prearranged solution.

Leo: I've had my miniLock key on my website for several years, and I've never received anything. But what are you going to send me? I mean, so anyway, there is a very clever password manager that uses PGP. It actually is, I think, extremely - if I were going to design a personal - but you have to maintain it. It does a tree, a folder tree of all your passwords in PGP encrypted. And then I guess it must have an in-the-clear name, so that leaks a little bit, but only locally. It's called Pass, if anybody wants to look at it. It's the only application I know of. And then there's Keybase,

which I use, and that's another way people can keep track of your key.

I do sign my mail. And I think that that's important. I think that that is, back to this BIMI story, if people might impersonate me, I do think it's good that, if you get an email - people impersonate me all the time - if you get an email from me that's not signed, not to trust it.

Steve: Yup.

Leo: Steve Gibson's at GRC.com. That's the Gibson Research Corporation. And you know what you'll find there? Many, many things. First and foremost, of course, SpinRite, the world's best hard drive maintenance and recovery utility.

Steve: Yabba-dabba-doo.

Leo: We hear another great story about SpinRite. You'll also find SQRL, Perfect Paper Passwords, ShieldsUP!, the DNS...

Steve: The DNS Benchmark.

Leo: ...Benchmark, wow, must have. InSpectre, still relevant. Lots of stuff. That's all free because Steve does it all out of the goodness of his heart, including this show, which you could find there. He has audio and very nicely crafted, handwritten, well, I'm sure she types, but human-written transcripts. That's using her hands. Elaine Farris does those each week. It takes a few days to get them out, but that's a great way to search the site for stuff, and it's a really nice thing to have.

We also have audio and video on our website, TWiT.tv/sn. In fact, if you're there, subscribe. You might as well get every episode. You want to have the collection, get all 657. It's a wonderful thing to have. And you can watch it, too, if you have a hankering to see how Steve does his Vulcan signoff there. And what else? Oh, we do it every Tuesday at 1:30 Pacific, 4:30 Eastern, 20:30 UTC, if you want to watch live at TWiT.tv/live and join us in the chatroom at irc.twit.tv.

Steve: Yay.

Leo: Yay. Thank you, Steve. We'll see you next time on Security Now!.

Steve: Okay, buddy. Thanks.

Leo: Bye-bye.

Steve: Bye.