Vault 7

Description: This week Steve and Leo discuss March's long-awaited Patch Tuesday, the release deployment of Google Invisible reCAPTCHA, getting more than you bargained for with a new Android smartphone, the new "Find my iPhone" phishing campaign, the failure of WiFi anti-tracking, a nasty and significant new hard-to-fix web server zero-day vulnerability, what if your ISP decides to unilaterally block a service you depend upon?, shining some much-needed light onto a poorly conceived end-to-end messaging application, two quick takes, a bit of errata and miscellany - and a look into what WikiLeaks revealed about the CIA's data collection capabilities and practices.

High quality (64 kbps) mp3 audio file URL: http://media.GRC.com/sn/SN-603.mp3
Quarter size (16 kbps) mp3 audio file URL: http://media.GRC.com/sn/sn-603-lq.mp3

SHOW TEASE: It's time for Security Now!. Steve Gibson is here. We've got a big, jam-packed show, all the security news. Some wild stuff, too. He's found another weird Game of Life creation, the Game of Life in the Game of Life. You'll have to see it or hear it to believe it. And then we're going to analyze that big dump of CIA hacking tools. They called it "Vault 7." Steve's take, coming up next on Security Now!.

Leo Laporte: This is Security Now! with Steve Gibson, Episode 603, recorded Tuesday, Pi Day, March 14, 2017: Vault 7.

It's time for Security Now!, the show where we talk about your safety, your security, your privacy, all about technology security with this guy right here. He's looking at his hand because he's got to make sure he can do the live long...

Steve Gibson: I've got to get the thumb out.

Leo: ...and prosper.

Steve: Thumb out.

Leo: Steve Gibson. Steven "Nimoy" Gibson from the GRC, the Gibson Research

Corporation. He's one of the, like, greatest guys ever, and my close personal friend, but also a true security guru and a computer guru going back to the earliest days.

Steve: Love the technology, Leo, whether it's health or computers or bits or a little bit of science. I just love technology.

Leo: You have, I think, if I were to say what your signal ability is, you have a deep ability to understand highly technical topics and then distill it down in a way that people can readily understand it. And I think that's the thing that people love this show most for. Not that it's not a challenging show. But, well, for instance, we've been waiting all week. The Vault 7 release, purported to be CIA hacks from WikiLeaks, came out right when we began the show last week.

Steve: Yeah, it was Tuesday morning last week.

Leo: And you said, as you often do, I want to have some time to digest it before I - unlike most of the rest of us, who will spout off just with the slightest notice, you like to actually think about these things.

Steve: Well, yeah. So that's our main topic for the week is Vault 7. We'll wrap up the podcast talking about that. And again, as always, bring the Security Now!, I don't know, background and wisdom to a lot of what the press showed and how various companies have reacted and so forth. But we've got, I mean, this is March 14, finally the long-awaited Patch Tuesday for March, for Windows. We'll talk about that. We have the release deployment of Google's Invisible CAPTCHA. Getting more than you bargained for with a new Android smartphone - actually many, it's like 38 or 36, I think, is the count. A new "Find my iPhone" phishing campaign I think I heard you talking about on one of the previous podcasts, might have been over the weekend.

An interesting bit of research in the failure of WiFi anti-tracking. We talked about the anti-tracking of WiFi a couple years ago. Turns out that it wasn't done right, unfortunately. There's a nasty and significant new, very difficult to fix web server zero-day vulnerability which is affecting many major websites. The question about what if your ISP decides to unilaterally block a service that you depend upon. Shining some much needed light into a poorly conceived, end-to-end messaging application that I know I've heard you talk about recently, Leo. A couple quick takes, a bit of errata, some miscellany, and then we're going to take a look at what we learned from Vault 7, and a different takeaway than I've seen anybody else talk about. So another great podcast for our listeners.

Leo: Good. That'll be very interesting. And of course, as soon as you said Patch Tuesday, I immediately opened my Windows laptop. You know, it's the funniest thing. Besides getting the regular Microsoft patches, Microsoft's now handling the Adobe Flash patches for Windows 10. I guess they decided maybe let us do that, shall we? Should we update Adobe Flash now for you? So I'm going to get a bunch of updates, and I'm looking forward to hearing what have I gotten.

Steve: So our Picture of the Week is something that a number of our listeners sent me photos of, and then it got picked up by ExtremeTech, and I grabbed the shot off of their web page. Their title is: "Microsoft now puts ads in Windows 10 File Explorer, because of course."

Leo: Oh, so frustrating. So frustrating.

Steve: Because they can.

Leo: Because they can.

Steve: And no one's going to tell them no because, after all, people have Windows 10. So, yeah, this is just - I just look at my - I just sort of shake my head. It's like, yeah, well, okay. We are the customer, so...

Leo: We talked about this with Paul Thurrott on Windows Weekly last Wednesday. And there's a way to turn it off, but it's completely obscure. It doesn't say "turn off ads." It says something like "turn off sync tool" or something. It's ridiculous. And I don't understand, well, I guess the operating system is free now for many people. But it just...

Steve: Well, actually they're just paying for it in a different fashion. And, you know. So for those who aren't seeing the video, in your file browser this says, "Get the best deal on your cloud storage with OneDrive. For $6.99 a month, an Office 365 subscription gets you 1TB" - and then it explains that's 1,000GB - "of OneDrive cloud storage as well as Word, Excel, and PowerPoint." Then you have of course two buttons, Learn More or Not Now. So, yeah, you know.

Leo: They put ads in the menu. Now they're putting ads in your File Explorer. It's just ridiculous.

Steve: Yeah, well, that's what you get. So you also get, today, on the second Tuesday of March, the long-awaited Patch Tuesday. It was no bigger than they have been because, as we know, the technology is now different. Basically it's all the things that are being changed are given to you at once, rather than granularly. And as I said when we first talked about this upcoming change in Microsoft's patching strategy, as a developer I really empathize with how difficult it was for them to do what they had been doing. The idea that - and I don't even know how you would go about this. The idea that you could be individually offering fixes on a granular basis where a user could opt out of any of them, or even potentially remove them after the fact, yet the whole system would somehow continue operating even without that, I just don't even know how you did that.

So the fact that they're doing this this way makes a whole bunch of sense. The important takeaway is we got in today's patch update everything we've been hoping for. We got seven critical sets of patches. There was an Adobe Flash Player update which was critical. The Microsoft graphics component that we have been talking about for a couple months and waiting for, we got that. That flaw in the SMB, the Server Message Block, so the

Windows File and Printer Sharing problem. That was where, if someone could induce your browser to reach out to a malicious SMB server, probably on the Internet, it's more likely to be there than on the Intranet, but basically any server anywhere could remotely execute code on your client. So that's been fixed. The problem with the Windows PDF library got fixed. The problem with Microsoft's Hyper-V got fixed. And then updates, critical updates - all these are critical - but then also to Microsoft Edge and IE.

So they fixed everything that we were hoping they were going to fix in February. We got 'em fixed in March. And then in addition there were 11 important update bundles, all documented in their security bulletin page. So this would be a good thing to update, specifically because a lot of these have been, I mean, you know, some of these were zero-day, and that's why they were so critical is we knew, I mean, they were first discovered because they were being exploited in the wild, and there weren't good workarounds. In fact, some independent security companies offered some patches that we talked about a couple weeks ago. And I said, you know, I don't know.

Leo: Yeah, that always makes me nervous.

Steve: I don't think I would go with that, yeah. We did also talk a few weeks ago, we've been talking about Google's reCAPTCHA and talking about how, because it's a piece of JavaScript which runs in your client and is sourced from within the Google domain, it's able to acquire your browser's Google credentials for you when you visit a site that is hosting Google's reCAPTCHA CAPTCHA. So what that means is that allows Google to deploy reputation scoring systems for you when you visit a third-party site that uses this reCAPTCHA.

Essentially, your browser, because the way cookies work, your browser makes a - it fetches the Google reCAPTCHA from the page you're visiting, which tells Google who you are. Google is then able to assert to the page you're not a robot. And that's what underlies just the click on the checkbox to say, yes, you're right, I'm not a robot. Everybody agrees. And then you're able to do whatever it is you want to do, post on a blog or log in or create an account somewhere where they're trying to prevent robots from doing that.

Well, we also talked about how in pre-release form for the last few months has been the so-called Invisible reCAPTCHA, where there's nothing on the page. That is, it doesn't actually, I mean, if this has been reduced to checking a box, certainly a bot could check the box. And people have been playing around with, like, well, do you have to kind of servo the cursor around? Do you not always click in the same spot? It turns out all of that was sort of on the surface. Essentially, Google has enough reputation information normally to make this assertion on your behalf to the site you're visiting. So the Invisible reCAPTCHA, it's in a web div, a division, at negative 10,000 pixels vertical. So it's like, where is that? That's, like, way up above where you can't see it.

Leo: Oh, that's interesting.

Steve: Yeah. So it doesn't even show. But it still executes. And if it decides it's not sure about you, then it will present itself on the page and maybe make you jump through some hoops - solve the puzzle, select the cat pictures, or whatever it chooses to have you do. So in the normal case, you see nothing. And in fact in the show notes, if anyone's interested, I've got two links for "invisible equals true" and "invisible equals

false." So you can demonstrate what it looks like. It essentially looks like nothing. It's just like as if the page is somehow able to determine that you're not a bot. And of course now we know how they do that. So it's sort a, in retrospect, an obvious thing. Google deployed it incrementally over time, confirmed as they rolled it out very slowly that it was going to do the job.

So essentially we all tolerated this horrific CAPTCHA stuff. I mean, there were some that I would just look at and go, okay, give me another one. I have no idea what that is trying to be, you know, where they were just random "jibbles" of letters and numbers. And it was like, okay, this is a problem. That's all gone now. So, yay. The world wins.

Check Point, that's of course a very well-known security firm, discovered through their mobile threat prevention system, on two unnamed clients, a large telecommunications company and a multinational technology company who are clients of theirs, they discovered severe malware infections on 36 Android devices within those organizations. So for it to be that widespread, it sounds like those organizations would have signed up for Check Point's Mobile Threat Prevention. Whereupon Check Point, upon doing their job, would have said, uh, you've got some problems here.

So here's what's interesting, though, is that in - it's like, okay, Android malware is not that big a surprise. But in backtracking what happened, they discovered that the devices arrived out of the box with this malware preinstalled. One was the Loki trojan that we've talked about before, which first appeared about a year ago, around this time in 2016, in fact February of 2016. It obtains root privileges and includes features such as grabbing the list of current applications, your browser history, your phone's contact list, call history, and location data.

They also found something called SLocker, which is mobile ransomware that uses AES encryption, as ransomware does, to encrypt all the files on the device and then demand payment in order to unlock it. In their forensic analysis, however, they were able to verify that this was not installed by users doing something wrong after the fact. The malicious apps were not part of the official ROM supplied by the vendor. They were added somewhere along the supply chain. Six of the malware instances that they found were added by a malicious actor to the device's ROM using system privileges, meaning that they could not be removed by the user, and the device had to be reflashed. So I think that the takeaway here - oh, and I should mention also these were name brand devices.

Leo: Most of them were really old. And they don't say where they were purchased, but they say it's safe if you buy it from the company store. My guess is these were purchased on eBay and places like eBay. And I think if you're stupid enough to buy a phone from a third party on eBay, I would - this actually wouldn't even surprise me that this would be the case; right?

Steve: Correct.

Leo: It's not - the manufacturer didn't do it. It's whoever sold it. It's a little self-serving because at the end Check Point says this wouldn't be a problem if you install check - I think that it probably is the case because they don't say that these were places, these were third-party resellers like eBay. Right? Who might [crosstalk]. And these were very old, for the most part, old phones, although I don't know what the

Galaxy Note 8 is because that's not a product that's being sold today. But it's the Note 2, the Note 4, the A5, the S4. These are all fairly, you know, years-old devices. Which sounds to me like it's eBay, probably.

Steve: So here's - I think our takeaway would be, I would say, if you were to purchase a phone, as you say, Leo, especially from some sketchy source?

Leo: I'd reflash it. I wouldn't even...

Steve: Well, yes, do that if you can. But at a minimum, immediately, before you start using it, put on some, do some Android mobile antimalware scanning. I did a little bit of browsing around, and it looks like BitDefender Mobile is currently near the top in the rankings. They offer a free download and installation with 14 days to try the app before you need to pay for it. And if you just, I mean, I think this is a nice tradeoff. Download it. Install it. You can do a manual scan to make sure that there's nothing that it knows about. They're claiming 100% hit rate on malware scanning. Certainly it would know about year-old trojans or ransomware. So just as a point of conduct, if you're going to get a new phone, why not download a free trial, run the scan, and then decide, maybe you want to keep it or remove it. And at least you know that you've given your phone the opportunity to find out whether it's carrying more than you expect it is.

Leo: By the way, it's the Galaxy Note 8.0, which is actually a three-year-old, a four-year-old tablet.

Steve: Ah, yes, right. And I think I heard you talking about over the weekend a Find My iPhone phishing scam.

Leo: Yeah. I've been bit by it, yeah.

Steve: Our friend Brian Krebs - oh, really?

Leo: Oh, years, well, not years ago. But when Henry was in Barcelona last spring, so almost a year ago, he lost his iPhone. And two or three days later I started getting texts purporting to be from Apple that said "click here." And because he had just - now, whether that's a coincidence or somehow they knew he'd lost his iPhone, I don't know. But when I looked more closely, it was not Apple.com, it was Apple.es. Or, no, it was something like that. Anyway, not Espana, it was Apple.estonia, I think. Or maybe ee. But nevertheless, I did click it first. And it pulls up an Apple login, what looks like an Apple login page.

Steve: Yup. Yup. So what's happened is this has gone mainstream. Brian Krebs has found, essentially, a rapid increase in this. So this is the sort of - and the way he does, he's dug back in and has determined that it's organized crime, often located in Russia, that are attempting to phish owners' Apple login credentials in order to first unlock, then examine, and then ultimately wipe and resell these phones. So someone loses their

phone, maybe they're part of a family plan. And so the remaining phone texts the other phone saying, hey, you know, if you've got my phone, I'd like to offer you a reward. So then the person who has it is able to respond, pretending to be the, oh, look, you've lost your phone. We're Apple. Click this link, and you'll be able to determine where your phone is, lock it or whatever. And of course the whole goal here is to phish the credentials from the phone's original rightful owner and then essentially acquire access to the phone that they wouldn't otherwise have. So again, as we have often said, attacks don't ever get worse. They only get better.

Speaking of which, a couple years ago, we talked about a problem with WiFi-based tracking of mobile devices. The problem is that Ethernet, which is what WiFi runs on top of, was never designed with privacy in mind. And we're talking 20 years ago Bob Metcalfe, originally at Xerox PARC and then at 3M, who was the inventor of Ethernet, back then it was a miracle if a network worked, rather than us putting all this extra requirement on top of it, like that it worked securely or it worked privately. The idea that you could even tie a bunch of computers onto a common communications backbone, and they would be able at high speed to talk to each other, that was amazing.

So what you needed to have in order to do this was some kind of unique addressing. And the solution was a 48-bit MAC address - M-A-C, all in uppercase, is the way it's normally seen, 48 bits. And that is organized as 24 bits, which is to say three bytes, of registered manufacturer ID, and then another 24 bits of ID within that manufacturer. The idea was that a company like 3M would have its own 24-bit designator, and it would generate NIC, Network Interface Controllers, each with their own guaranteed unique address, by incrementing the lower 24 bits. So the 3M address, plus the serial number, the 24-bit serial number within 3M, would be concatenated into 48 bits.

And the idea was then that way, if you had different NIC adapters from different manufacturers, the upper 24 bits would be different because they're different manufacturers. So there'd be no problem with colliding, with a whole 48 bits colliding. Even if the lower 24 bits happened to have the same within the manufacturer serial number, the concatenation of 48 bits would be guaranteed to be globally unique. So what that - back then, I mean, today it almost seems sort of like a quaint solution. Well, isn't that cute.

What back then it guaranteed was you could just attach network interface cards, as many as you wanted to, from wherever. And every single one on the planet, even though they weren't all connected together globally, but within your own little LAN, they would have different addresses, different MAC addresses. And so that's the way packets on Ethernet find each other. And of course we've talked in the past about how the address resolution protocol, ARP, that's the mapping between the IP address. If you have Internet protocol running on top of Ethernet, it's the way for IP address to find the MAC address of the adapter with that IP and get the data where it's supposed to go.

Okay. So then we go to WiFi. Well, WiFi is an Ethernet protocol. 802.11 runs on top of Ethernet. So what that says is that our mobile devices, and even PCs that we take out with us roaming, they have a MAC address. And so some years ago, as privacy began to be a problem or a concern, people said, hey, you know, our smartphones that have physical global MAC addresses are a privacy problem because, essentially, even if you don't know the encryption of the IP content, even on an encrypted WiFi network, the underlying packet is where the MAC address is, and that's never encrypted. So what that means is anyone passively just receiving packets out of the air, even if it's an encrypted access point, they see the physical MAC address of all the devices in that region.

So as we discussed a couple years ago, when this was first mitigated, devices began to

back off of that and to randomize their MAC address. They would still have a true global MAC. But if they were just pinging access points in order to see who's in the neighborhood, like when you're not associated with an access point, and you say, okay, where can I connect, and you get that list of access points, well, what's happened is your device has sent out a broadcast saying, hi, I'm here. Who can hear me? And all the access points in the area respond with their SSID, which is what gets listed in that list.

The problem is, if that's your actual physical MAC address, you've just announced your identity, essentially, trackably, to all the access points that receive that, and anybody else listening. So the change that was made was to - the realization occurred that we don't really have to use a fixed MAC address for this purpose. We could randomize the MAC address when we're just out roaming, you know, driving down the street, because essentially WiFi is always out probing, looking for access points in order to list. So there's a lot of this communications going on, all of which is, unless we mitigate it, is broadcasting a fixed globally unique ID as part of this request, this essentially sort of a WiFi Ethernet ping to say, hey, you know, who can hear me?

So a group of eight researchers at the U.S. Naval Academy decided to take a close look at this WiFi MAC address randomization, primarily on Android phones, although they also saw some problems on iOS. And what they discovered, unfortunately, is that it was failing to provide the intended protections. They have a long paper, I think it's maybe 17 dense pages of details.

But just from the abstract they said: "Media Access Control" - which is what MAC address stands for - "randomization is a privacy technique whereby mobile devices rotate through random hardware addresses in order to prevent observers from singling out their traffic or physical location from other nearby devices. Adoption of this technology, however, has been sporadic and varied across device manufacturers. In this paper," they write, "we present the first wide-scale study of MAC address randomization in the wild, including a detailed breakdown of different randomization techniques by operating system, manufacturer, and model of device.

"We then identify multiple flaws in these implementations which can be exploited to defeat randomization as performed by existing devices. First, we show that devices commonly make improper use of randomization by sending wireless frames with the true global address when they should be using a randomized address. We move on to extend the passive identification techniques to effectively defeat randomization in around 96% of Android phones. Finally, we show a method that can be used to track 100% of devices" - and that does include iOS devices - "using randomization, regardless of manufacturer, by exploiting a previously unknown flaw in the way existing wireless chipsets handle low-level control frames."

So this is not a huge concern. The takeaway is that this is the kind of research that we need people to be doing because I read the whole paper, and there's lots of details that we don't need to get into, the point being that mostly from what looks like just laziness, this wasn't done right. What they found was lots of simple mistakes in the logic of what was being done. And in fact, many Android devices don't do any MAC address randomization at all. It is absolutely present across iOS, but iOS is a monoculture; and so, yes, Apple did that in iOS, I think from back at 8, if I remember correctly. It's been there ever since, and it's there universally. However, not without some exploitability. But that's an active attack rather than just passive listening.

They posited that maybe some chipsets didn't support it, but then they were able to verify that wasn't the case because they found that there were some devices with the same chipsets that weren't supporting it on Android that were also on Android, meaning

that some manufacturers just don't care. They're just not bothering to do this. And who would know, if someone didn't look and didn't report on this? So nice piece of research. Nothing we as end users can really do except, if you were really concerned about this, then at least now you know that this is happening, that this global MAC which is in the hardware is passively leaking in a majority of cases and can be requested in an active attack across the board.

So what I expect will happen is that, to the degree that manufacturers care - for example, I would imagine Apple will care, and they will at some point fix this so that this newly revealed attack, which can be thwarted, but just isn't being right now, would be fixed. And then other manufacturers, to whatever degree they care, can fix it if they choose. But again, this is the kind of feedback loop that the industry needs, with researchers taking it upon themselves to check what's going on and then publish their findings. So again, not a huge deal, but good to know.

There is a huge deal, however. And this is a nightmare. About 10 years ago there was an earlier version of something called Struts, which is a widely, now a widely used open source web application framework, based on Java, for Apache. I think the first release was in '05. And then a couple years later Struts 2 was released in 2007, so 10 years ago.

It turns out there's a vulnerability in the Jakarta file upload multipart parser, which is a standard part of the framework. And Struts supports something called OGNL, which is Object-Graph Navigation Language, which is an expression language for getting and setting properties of Java objects. So it's a sort of a scripting language for JVM. However, Struts would mistakenly execute this OGNL language, even if it appeared in the Content-Type header, which is unconscionable because it then allows - it gives attackers control, which makes this very dangerous.

So here's the problem. The awareness of this decade-old - wait a minute. I'm sorry. Struts 2 is a decade old. This problem was introduced in 2012. So, still, five years ago. So for the past five years all of the Struts 2 server-side web application frameworks have had this problem. The problem is that it's not a standalone dynamically linked or dynamically invoked library. Rather, it is statically linked, meaning that it is compiled into these web servlets running on the server into Java executables. That means that fixing this problem is not just a matter of running apt-get and fixing a module and then rebooting your system or installing an update. The problem is this is compiled into often hundreds of little applets which are running applications on servers. And every single one of them has to be recompiled from source, using the just-a-few-days-ago-made-available patched Struts 2 when this problem was fixed.

So an organization may have hundreds of these little Struts-using web apps, all with their own Struts JAR embedded in them. And many of the apps may essentially be abandoned. So they're still in use, and they're deployed; but they're not receiving ongoing maintenance because they're finished. And it's been, what, five years. The developers may have moved on, may be at other companies. The point is it's way more difficult to fix this.

In Ars Technica coverage, Dan Goodin wrote that researchers at Cisco Systems said they were seeing a "high number of exploitation events" by hackers attempting to carry out a variety of malicious acts. One series of commands that attackers are injecting into web pages stops the firewall protecting the server, then downloads and executes malware of the attacker's choice. Payloads include "IRC bouncers" which allow the attackers to hide their real IP address during Internet chats, denial-of-service bots, and various other packages that conscript a server into a botnet.

And just to take this out of the realm of theory, I already had the show notes put together when I picked up on another piece of news about this. Canada has just taken a major tax site offline due to these attacks. Reuters, it was a story in Reuters that I saw saying that a newly discovered vulnerability in Apache Struts 2 software has forced the Canadian government to close down the Statistics Canada site used for filing federal taxes. The site came under attack from hackers, but was immediately shut down before any damage could be done. Well, of course, except that not having a site up is a problem. So this is a big problem.

The security bug in Apache Struts 2 software is used mostly in websites of government, banks, and retailers. It was reported last week, after Apache Software Foundation came out with an update to fix the vulnerability. But again, the vulnerability was released. Then it was reported two days later. Except that this is not something you can patch and reboot your server. You have to rebuild all of the applets that were statically bound to the previously, for the last five years, incorrectly functioning software. So a number of security professionals, one was quoted, Chris Wysopal at Veracode said: "This vulnerability is super easy to exploit. You just point it to a web server and put in the command you want to run." And it's easy to scan the Internet for servers using these vulnerable web applets, that is, delivering pages from them, and exploit them. So this doesn't look good, and it's not clear that it's going to be fixed anytime soon.

Leo: Maybe we won't have to pay taxes this year.

Steve: Yeah.

Leo: No. Said no one ever.

Steve: We can wish. So what would happen if your ISP decided to block a service that you depended upon? We know that they block lots of things to protect us, so they believe. For example, they for years have been blocking Windows Printer and File Sharing by blocking port...

Leo: Thanks to you.

Steve: Yeah, ports 137 through 139 and 445. And they'll block things like port 25 because they don't want people to run spamming servers within their networks. So some things they do for themselves, some things they do for their customers. Well, it turns out that the well-known ISP TalkTalk decided unilaterally, I mean, I'm not sure who they would ask to make it bilateral...

Leo: Is it okay if we do this?

Steve: Yeah. They just decided that TeamViewer was a problem because TeamViewer was being used maliciously by phishers and scammers, getting their clients, TalkTalk's customers, to download TeamViewer, pretending to be, for example, Microsoft Tech Support, and oh, your computer is infected. Download this, and we'll take a look and fix it for you.

Leo: Oh, yeah. So maybe that's why, yeah, yeah.

Steve: Well, yeah, I mean, that's definitely why. So it is a very popular, easy to use, remote desktop application. The problem is it's also a good system. I mean, it's frequently used by IT to support remote users. So just out of the blue they added some port filtering to their entire network, specifically for the purpose of blocking TalkTalk. Well, it's successful. It worked. So a whole bunch of users started complaining, initially, probably because they hadn't gotten of the attention of the right people. I don't think TalkTalk deliberately was prevaricating. But someone was initially saying no, no, no, it's not our problem, not our problem.

Finally they said, okay, yes. In their posting, in their community blog, they said, or their page: "Apologies for the confusion" - and this is a TalkTalk employee. He's saying, "But I can confirm that we have implemented a number of network changes that have blocked a number of applications including TeamViewer. We constantly monitor for potentially malicious Internet traffic so that we can protect our customers from phishing and scamming activities. As part of this work, we've recently blocked a number of sites and applications from our network, and we're working hard to minimize the impact on our customers. We're working with TeamViewer and other third parties on implementing some additional security measures that would enhance the security to all customers," blah blah blah.

Anyway, the point is that this has been a major inconvenience to their customers. And I read through some of this ongoing dialogue to see what was there, and I pulled a couple things. One person said, quoting TalkTalk, "but we will continue to block any sites/applications reported by customers to reduce the opportunity for fraud to take place." And this person responded, "Great. What about the same consideration for customers who don't report any problems with sites and applications because they haven't had any?" And this guy goes on to explain that he's been using TeamViewer via TalkTalk for many years without any problems. Now all of a sudden, with no warning, we can't, and no one is saying how long, if at all, before we can use it again. Corporations, businesses, and IT departments worldwide can use TeamViewer, but TalkTalk customers can't. Extremely unsatisfactory customer service, and so on.

So to me, the takeaway here was, well, consumers don't have much control. The only thing you could do, if VPNs are not blocked, would be to use a VPN in order to get your [audio dropout] outside of the control of an ISP in order, then, to get unfiltered access. And I should mention that, when I had my two T1s, I was using Verio back in the day and had paid for expensive commercial bandwidth. Now I'm using Cox, and I am behind Cox's filters. And it was an inconvenience for me because there were things I was doing that, for my residential connection to GRC's remote servers, that Cox is blocking. And I had to work around it. I did because I have all the technology I need. But most consumers don't. And suddenly something that they were using and had been depending upon stops working.

Leo: I don't know if this is the case with TalkTalk. They're a British ISP. But for Cox and Comcast, for instance, if you get their business class service, which is much more expensive, then they are much slower to block these kinds of things. The other thing I'd note is that TalkTalk has had, of late, a big problem because Indian tech companies have somehow gotten the data from TalkTalk's customers, probably a breach; right?

Steve: It was a SQL Server mistake, yes.

Leo: Yeah. And so as a result a lot of their customers are targeted by these scammers, and they know stuff; right? So that it's even more credible. So I think TalkTalk's probably doing a rearguard action to protect themselves.

Steve: I think that's exactly right, yes. I'm glad you reminded me of that because I'd forgotten that they had - they did have a breach. And so with that data, that meant that TalkTalk's customers could be effectively targeted.

Leo: Right.

Steve: And so now they're saying, oh, you know. And now they're, like, reducing service in order to make up for the fact that their users are known.

Leo: Well, we'll keep you from getting scanned. Can you use - I wonder if you can use TeamViewer on a dedicated port? Wouldn't that - well, depends what they're doing, I guess, if they're doing packet inspection or just blocking the port.

Steve: Yeah. And I'm wondering, from some of the dialogue, I didn't look to see whether it was a point to point. It might be running through the TeamViewer servers. So they might be blocking access to the TeamViewer servers by IP range, for example, rather than by port number. So that may be what's going on.

Leo: Yeah, that's what's going on because they'll do NAT traversal, I'm sure.

Steve: Right, right. And I'm sure I heard you talking over the weekend about Confide. This got in the news. And I think Matthew Green, our friend at Johns Hopkins, must have been having a bad day, because he had kind of some grumpier than usual tweets about this. So I have to say, okay, so this is a venture-financed startup about four years ago. 2013 these guys appeared. And this is one of those super slick, polished-looking sites. If you go to GetConfide.com, G-E-T-C-O-N-F-I-D-E dotcom, I mean, that site is everything GRC is not. It is, I mean, clearly it's got - stuff jiggles around and swoops in from the side as you scroll the page. And, I mean, it just looks fabulous. Super slick, polished, confidence-inspiring website. And unfortunately what they offer is garbage.

Leo: I'm sorry, I shouldn't laugh.

Steve: But, boy, is it pretty. I mean...

Leo: And also unfortunately it's being apparently used by a lot of whistleblowers in Washington, D.C.

Steve: Well, yes. One of the many things that they offer is this quickly erasing messages, where I guess you draw your finger across the screen. I mean, the fact that it's implemented in JavaScript - the whole thing is actually written in JavaScript. It's like, okay. That's not necessarily bad. But it just sort of says, oh, this is more eyewash than anything else. So in mid-February, middle of last month, Alan Woodward, a security researcher and professor at the University of Surrey, characterized Confide as, quote, "a triumph of marketing over substance." And of course the home page, not only do they have military-grade encryption, but theirs is also, Leo, battle-tested.

Leo: Ooh.

Steve: So the home page says: "Your Confidential Messenger. Communicate digitally with the same level of privacy and security as the spoken word." Meaning that after it's been spoken, it disappears. "With encrypted messages that self-destruct, Confide gives you the comfort of knowing that your private messages will now truly stay that way." And I said in parents, "(or not)."

Leo: This is the paragraph that would make anybody who listens to Security Now! laugh: "All communication goes through transport layer security, preventing any possible man-in-the-middle attack."

Steve: Right.

Leo: Well, of course.

Steve: Okay. It actually says that. I'm glad you read that right off the page because - so Matthew Green, apparently having a bad day, first he tweets: "The encryption in Confide looks genuinely bad. Don't use it, people. What's the matter with you?" And then he gives us a link to an Ars Technica article. Then in his next tweet: "Here's the technical blog post from Quarkslab."

Leo: What's the matter with you?

Steve: What's the matter with you? Then he says, "In short: no key fingerprinting, bad encryption mechanism, blegh." And then he gives us the link to Quarkslab. And finally, as if that wasn't enough, he says, "Oh, but Confide uses TLS pinning. That's nice. I'd ask" - this is Matthew in his tweet. "I'd ask why people keep trying to reinvent their own end-to-end crypto, but I know the answer. People are just the worst." It's like, okay, Matthew. Maybe you've had [crosstalk].

Leo: Definitely a bad day.

Steve: It's just a bad day.

Leo: What's wrong with you people?

Steve: So Quarkslab, their blog, I've got the link in the show notes if anyone really wants to go into it because these guys go into great detail. They tear it apart with a step-by-step walkthrough, showing the way they reverse-engineered and examined the system. In their summary, they say: "TL;DR. Confide server can read your messages by performing a man-in-the-middle attack." In other words, this TLS that they boast about with its military-grade encryption...

Leo: Battle-tested, don't forget.

Steve: Battle-tested, is to them, and they are the man in the middle. And then in their own FAQ they ask themselves the question...

Leo: But we would never.

Steve: Oh. Yeah. How secure is this, and do messages really disappear? And they say: "We employ end-to-end encryption to ensure conversations remain confidential and are private to you. Even we at Confide cannot decrypt or see any messages. Yes, after messages are read once, they disappear." And that's like, yeah, okay. Their big claim to fame is that they intercept the screen snapshot. Of course you could take a picture of the screen with a different [crosstalk].

Leo: Right. Nothing can stop that, yeah.

Steve: Yeah, exactly. That's why this whole, all of this nonsense about disappearing messages is like, as Matthew would say, "Blegh." It's just...

Leo: What are you people thinking?

Steve: Or actually as Matthew DID say. Oh, boy.

Leo: You've got to love Matthew Green.

Steve: Yeah. As we know, producing a secure end-to-end encryption messaging system is truly difficult. But that's a problem that's already been solved. If you want maximum security, choose Signal or Threema, and take the time to verify your contact's key fingerprints. And if you're using Signal or WhatsApp, turn on the not-on-by-default "notify if the fingerprint ever changes" feature. And then pay attention to that if you're ever notified because there should be a reason for that to change. And as we'll be discussing as the topic later in this podcast, end-to-end encryption is only secure if both ends are secure.

Leo: Yeah, guess that's true, too.

Steve: Yeah. The tunnel maybe absolutely impenetrable. But if you can watch the traffic going in and out of the tunnel, sorry.

Leo: The good reason for keys changing is if you change phones. Right? I'm going to install Signal on my new phone, and my key will change; right?

Steve: We talked about that. I don't remember now...

Leo: Maybe not.

Steve: ...if you can use the same key on the other phone. I think...

Leo: Maybe you can, yeah.

Steve: I think maybe you do, yeah. But again, if the key changes, then you just need to reverify that it is still the person [crosstalk].

Leo: New key. Who dis?

Steve: Exactly. So we are, as of last Saturday, March 11, that was the 28th anniversary of Tim Berners-Lee submitting his original proposal for the World Wide Web. And I wasn't that impressed with what he just wrote. It's a little political.

Leo: Yeah, me neither. Yeah, yeah.

Steve: Yeah. It's like, okay. So "I invented the web," he writes. "Here are three things we need to change to save it." And The Guardian picked up the story. And the tag was, "It has taken all of us to build the web we have, and now it is up to all of us to build the web we want for everyone."

And so just the little, brief beginning of this, he says: "Today marks 28 years since I submitted my original proposal for the worldwide web. I imagined the web as an open platform that would allow everyone, everywhere to share information, access opportunities, and collaborate across geographic and cultural boundaries." Okay, I don't think he did that 28 years ago. It was network publishing 28 years ago. It has certainly, you know, our definition of it has radically changed in that time.

Anyway, he continues: "In many ways, the web has lived up to this vision [I would say it's far outstripped that vision] though," he writes, "it has been a recurring battle to keep it open. But over the past 12 months," he writes, "I've become increasingly worried about three new trends, which I believe we must tackle in order for the web to fulfill its

true potential as a tool that serves all of humanity." And then he enumerates those three. And I won't go into any detail, but those three are: We've lost control of our personal data, it's too easy for misinformation to spread on the web, and political advertising online needs transparency and understanding. Okay, well, to me, number three seems like a special case of, like, a bigger problem.

My own personal take, and I've sort of been referring to this lately, if I were to address what's wrong, I would say we need a usability fix. We need a meta layer to wrap and hide what I would consider legacy protocol and domain name mess. You know, the http, https. And even hierarchical domains should all be hidden so that users interact with labels that can be mapped transparently to all of that other gobbledy-gook so users see Amazon and Google and Apple, and none of this other nonsense. To me, that would be a nice change. I don't disagree with the things that Tim said. But it's like, okay. I don't know that I think that's the biggest problem. Leo?

Leo: Also, how to solve. I mean, I don't know how - does he propose a method? I mean, maybe these are problems. They aren't not problems. But this is general problems of the 'Net, and I don't know what you do about that.

Steve: Right, right. Well, and he talks about how there's been a fight to keep it open. Well, we've lost control of our personal data. The only way that gets fixed is legislation, heavy-handed, you know...

Leo: Right. And that's what worries me. It sounds like government interference at this point.

Steve: Yes.

Leo: Who's going to decide what political advertising is good and what's not? That worries me more than anything; right?

Steve: Exactly. And he says it's too easy for misinformation to spread? Well, okay. But then now you're talking about some sort of authoritarian control over, or maybe a reputation system, I mean, again...

Leo: Yeah, I don't see any easy solution for what he's talking about.

Steve: No. I mean, we're talking about, you know, it is very democratizing. But these are the consequences of...

Leo: This is free speech.

Steve: Yes, exactly.

Leo: Yeah. We have more experience with free speech here in the U.S., I think, than they do in the U.K. So maybe that's it.

Steve: Let's hope we continue having that experience.

Leo: Yes.

Steve: So a couple quick takes. I got an interesting question that I wanted to address, just because it was perfect for our crypto technology. Tom Elliott sent me a tweet saying: "I have a dev telling me that storing SSN" - meaning a Social Security Number - "in SQL using an unsalted SHA-1 hash is secure. Isn't this susceptible to time-memory tradeoff attack?" And he says, parens, "(Rainbow tables and some GPUs)."

And so I responded - because he had DM'd me, so I could give him a longer response. I said: "Tom, that cannot be secure. SSNs do not contain sufficient entropy to prevent having their hash brute-forced. It doesn't matter whether it's SHA-1, SHA-256, salted, or unsalted. The search space for a nine-digit" - which is what Social Security Numbers are - "all-numeric identifier is too small. Also, storing the hash of an SSN, that is, putting the SSN behind a one-way function, suggests that the SSN is being used as an identity authenticating token. Since it cannot be decrypted, it can only be verified in the future. This is horrible design and policy since Social Security Numbers are inherently tied to people's identity. This makes it much worse than 'your mother's maiden name' or 'your first dog's name.'"

Anyway, so I thought that was an interesting question. I don't know what developer said, oh, yeah, you know, we're just going to hash the Social Security Number. Well, okay. If you're hashing it, that's because in the future you want to query for it again for someone to prove their identity. But Social Security Numbers are leaked out on the Internet all the time. So it's already tied to identity, which makes it a bad additional authentication token for identity. Anyway, great question. And even so, it just, you know, a nine-digit number is just not sufficiently entropic to be used because it would be trivial to brute-force that through any hash. And even if it's salted, assuming that you've got the database with the hashed SSNs, you're going to have the salt there. And again, it just doesn't have enough entropy to protect it.

Aaron Watt send me a tweet saying - actually to both of us, @SGgrc and @leolaporte: "Panasonic hasn't patched their current Firefox OS for their Smart TVs in over a year. Should I be worried about it?" And I would argue, that's the wrong question because it's an IoT device. We do know, after the Vault 7 attacks, that SMART TVs are not remotely vulnerable, as far as we know, but certainly can be exploited, as many IoT devices can be.

I would say, if you are worried, and if you do not need to have access to it in your main network, that is, if it's just an Internet-connected device, do what we've been talking about for all IoT devices, which is to put it on its own network segment. Let it have Internet access, but don't let it see the rest of your network. Isolate it. And then, although there's still a problem from, for example, a privacy standpoint, if it could be used to spy on you, at least it can't be a beachhead from which attackers can then get access to the rest of your network.

So network segmentation really does seem like an increasingly important thing. And I

imagine at this point a lot is required from people who want to do that. I'll bet you that we see IoT-oriented routers in the future, where they're just offering...

Leo: Oh, god, yeah. Yeah.

Steve: ...a network segment, IoT network segment, and solve this problem for their users.

Leo: The current crop of mesh routers in many cases could do that because they're already identifying devices and modifying paths and so forth, based on the device. And so I don't think it'd be - it would be a fairly easy thing, I would imagine. They wouldn't require hardware updates; right? They could just - can't you do VLANs in software? I wonder.

Steve: You can if the hardware permits it.

Leo: Right.

Steve: One of the things that I discovered was that many of the - if it's just a router connected to a switch inside the box, then you have a problem because a switch won't isolate. But what we learned was that many of these firmwares are using a much more capable chip than one would think. That is, they've configured the hardware to be a switch. It is actually a router. So, yes. If they took the trouble to update their firmware, they could potentially offer actual port-level network isolation.

Leo: If you can do a guest network on a router, doesn't that mean you have at least some kind of VLANing capability? Isn't a guest network - depends, I guess, on how the guest network is handled. But if it's isolated from the main network...

Steve: Yeah. Normally guest networks will be a different subnet. So you'd have, like, 192.168.0.something and then .1.something. So it's a different subnet.

Leo: That's not isolating, though.

Steve: Right. And normally it has, for example, in the WiFi case it has its own SSID and password. But again, once you're on the guest network, then you could reach over to the other network.

Leo: All right.

Steve: A bit of errata. Two people tweeted, a Dan Hankins and TradMan. Dan tweeted: "@SGgrc Mic jack trick won't work on laptops, where jack function is software assignable. There isn't and can't be a hardware cutout." And TradMan said same, you know:

"@SGgrc Also the" - because he had sent me another tweet - "the mic plug hack isn't sufficient. That just activates a software switch. Even with mic plugged in, software can see the internal mic."

So I wanted to correct the record because I had shared a tip from another reader saying, hey, you know, plug something into that hole, and that'll disconnect the mic because it will do it, it'll connect - the hardware will route the mic out to the hole, the mic jack, which you then short in order to send nothing in. But I'm sure these guys are right. We talked in fact about how the RealTek chips, which are by far the most popular in the industry, all of those ports are software assignable as inputs and outputs. Which is how, for example, even your speaker can be turned into a microphone. The software just reassigns it as an input rather than an output.

So I'll bet these guys are right. All of the recognition of something being plugged in is handled in software. So while it might up the ante a bit, that is, it wouldn't just listen to the default microphone. Malware might have to do a little bit more work. It probably could. So it's certainly not as good. It's not the equivalent of putting an opaque piece of tape over your webcam that physically blocks the image. We don't really have that power, from a software standpoint.

Leo: Steve Gibson, Leo Laporte, and the CIA.

Steve: I was wrong about something else, Leo.

Leo: Uh-oh. What? Never.

Steve: Last week I believed that I had found the coolest waste of time ever, by someone who built, remember, the working digital clock out of Conway's Game of Life.

Leo: Yeah.

Steve: Okay, I was wrong.

Leo: Something cooler?

Steve: Yes.

Leo: By the way, I let that run overnight. It was awesome. It totally worked.

Steve: Thanks to some of our listeners who took and were interested and dug around, they found something even more incredible: Conway's Game of Life implemented in itself.

Leo: What?

Steve: Look at this YouTube link in this Miscellany here.

Leo: Okay.

Steve: Our listeners won't be able to see what's going on, but our viewers - because it's just a fabulous, fabulous presentation. I will tweet the link after the show. And if anyone is interested in cellular automata, this is just - this is another, just an incredible piece of work. Someone built Life in itself.

Leo: I'm not sure I really understand what that means. But I guess we'll watch, okay.

Steve: Watch. Watch.

Leo: This is Life, and it's expanding. We're zooming out.

Steve: We're zooming out.

Leo: Okay. Is there audio? I could turn on the audio. Oh, yeah, perfect.

Steve: And those are little gliders that we're seeing moving together and then annihilating themselves.

Leo: Okay.

Steve: And we just keep moving out. So that gives you a sense of scale.

Leo: We're pulling out farther and farther. It's kind of cool. Where the lines meet they explode, they disappear.

Steve: Yup.

Leo: Yeah. But then they continue to be created by the replicators at the end of the line.

Steve: On the edges, yup.

Leo: Yeah, yeah.

Steve: To create a filled-in square.

Leo: Yeah, yeah. Okay. Zooming out some more, we've got streets, it looks like. Or I don't know, what could this be?

Steve: Watch.

Leo: Wear headphones for the best experience. It's a grid. Some of the boxes are filled in; some are not. Oh, is it going to spell Life? No?

Steve: No. That is...

Leo: Oh, it just changed. Oh, it's a Game of Life.

Steve: Yes.

Leo: That's just crazy.

Steve: Is that unbelievable?

Leo: Because, okay, so the squares that were filled in were pieces of life in the Game of Life, following the same rules.

Steve: Yes. They were live cells.

Leo: Oh, this is insane. Oh, this is insane.

Steve: Oh.

Leo: So it's Google or YouTube, Epic, Conway's Game of Life. And you can see it for yourself. Wow, that's wild.

Steve: Yes. It is just unbelievable.

Leo: It is a Game of Life. And then, as you zoom out, it is a macro Game of Life

written in, I guess, I don't know even how to describe it.

Steve: Wow.

Leo: Wow.

Steve: Okay. And our final bit of miscellany. It turns out that potatoes can grow on Mars.

Leo: If you read "The Martian," you'll know what this references.

Steve: In a nod to Mark Watney, the International Potato Center - who even knew there was such a thing? There actually is.

Leo: The International Potato Center?

Steve: Yeah, the acronym's a little different. It's CIP, apparently the Center for, I don't know, International Potatoes?

Leo: It's probably in French. They always do everything backwards, you know.

Steve: This is from the Phys.org site, P-H-Y-S dot org. And oh, my goodness, Leo, anyone who is a science geek if you don't - just put Phys.org into your browser and look at the home page of this site. You will lose yourself in cool articles. But anyway: "The International Potato Center launched a series of experiments [I kid you not] to determine whether potatoes can grow under Mars atmospheric conditions, and thereby prove that they are also able to grow in extreme climates on Earth. The Phase Two effort of CIP's proof-of-concept experiment to grow potatoes in simulated Martian conditions began on February 14 [last year] 2016, when," they write, "a tuber was planted in a specially constructed CubeSat contained environment built by engineers from the University of Engineering and Technology in Lima, based on designs and advice provided by the National Aeronautics and Space Administration," our own NASA, "at Ames Research Center in California. Preliminary results are positive."

Leo: There's a tater, right there.

Steve: Yes, you can grow potatoes on Mars.

Leo: Tuber in a test tube.

Steve: Just in case you thought that might have been a little fictional.

Leo: I mean, he did have to provide it with nutrients.

Steve: Yes.

Leo: It was sterile soil, but...

Steve: Yes, there was a lot of recycling being done.

Leo: Yes.

Steve: Yes. He needed a bacteria.

Leo: By the way, they have a Twitter account, CIPotato.

Steve: So Sean, oh, I didn't pronounce - I didn't practice spelling his name before. Kloeckner, Sean Kloeckner. I guess he's nearby, HB in the OC, he said. He wrote: "Took your advice on CRC errors." This was interesting. He said: "Hi, Steve. I've been a listener for probably a couple years now, and I appreciate all the advice you give on the show. I sent in a previous note about Syncthing and what you think of it since I heard you complaining, in a good way, about BTSync previously." Meaning that they refuse to document their protocol, so okay.

He says: "Syncthing is totally open source. I have to say it's been functionally everything BTSync is and works great as a Dropbox replacement." And I'll just say, yes, I have heard good things about it. I've looked at it. But I've never had a chance yet to dive in deep. He said: "I no longer need to worry about my data in a company's hands, and it replicates to all my other PCs." So that's a great tip.

He says: "Anyway, I wanted to let you know I've been an avid listener and bought SpinRite recently. I'm a Linux user and don't really need to buy software, but I know SpinRite would come in handy, and thankfully haven't needed it for anything other than testing and maintaining drives in my lab.

"I recently bought new PNY drives and used them for a ZFS root system using Proxmox, and noticed whenever I scanned my zpool there would be checksum errors that it would correct. Few weeks went by without me touching them, and everything works generally okay. But every time I would perform a scrub on my pool, it would still return errors." Well, I'll explain what that means. And that proves the point I will make.

He says: "I learned in more detail that certain drives can return junk when under duress, but this is on a fresh install, every time with the same result. Anyway, long story short, after seeing CRC errors in the syslog and then running SpinRite, it confirmed the CRC errors. So I took your advice and replaced the cables. I ran some further benchmark and dummy data tests on my pool and SpinRite. And lo and behold, no more checksum errors.

"Love the show, wanted to pitch in my two cents for other listeners out there who may

be in the same situation. What you do is a public service." Well, okay. So my...

Leo: So SpinRite works with ZFS, which is neat.

Steve: Well, okay. So here's the problem. A checksum error, when found, will force a repeat of the failed operation. So as we mentioned before, the IDE or SATA and SAS cables, they add a checksum to the data transfer in order to detect a transmission error because, think about it, the computer is sending its data to the drive. Well, we need the data to get there safely. Then we need to trust the drive to store it. But if the data, if there's an actual data in the transmission from the motherboard to the drive, the drive could be right. In other words, it could correctly write what it received, but what it receives could be wrong.

So consequently the transmission over the cables is CRC checked. And SpinRite, as I had mentioned before, and this is what Sean was referring to, SpinRite will count and track those errors. And you should never see any. And some people see a lot of them. Well, what that means is the cable has a problem. So SpinRite showed that the cable was having a problem. But the bigger point here is that you cannot depend upon a cyclic redundancy check, a CRC, to find them all because it's not big enough. There's a statistical likelihood of it finding a problem. When it detects a problem, it will force a reread or a rewrite. That is, that transfer will fail because of its CRC failure. And then it will be retransmitted, and it will probably work.

But the checksum, it is not big enough to guarantee catching everything. It's not like an SHA-256, where the chances of a collision are diminishingly small. So what he was doing was his system was finding checksum - he was finding transmission errors that were missed because his cables were so bad. First of all, the badness of the cables were slowing down the system's operation because so many retransmissions would have been required. But the fact that he was finding errors in his ZFS log meant that errors were being written. So that proves that his cables were generating errors, and some were being missed.

So the takeaway is, if you run SpinRite, and it shows you you're getting CRC errors, the smallest effect that can have is an impact on performance because those errors are forcing retransmissions. The bigger problem is, especially if you're not running a checksum ZFS file system, that is, if an error is missed, you will write the wrong data on your drive. Reading is not such a big problem because when you read it again you'll get the right data. You won't know you read it wrong because it'll be missed. But, boy, writing it wrong, that then is it's wrong forever on the drive.

So cabling errors is what SpinRite calls them. Because people kind of like, what's CRC? Anyway, I call it a cabling error in the SpinRite UI. And I do track them and check them and show them. They need to be taken seriously because there's no guarantee they're all going to get caught. So if you're seeing a lot of them, that increases the likelihood that some are going to get through.

Okay. Vault 7. So as we know, at early morning a week ago, seven days ago, WikiLeaks released 8,761 documents and files which allegedly and believably - even the CIA, they said, "We won't comment," but they did not deny. This exposes and discloses the tactics and technologies the U.S. Central Intelligence Agency uses to hack into secure devices, systems, and communications.

This is the equivalent, roughly, of what we learned from Edward Snowden after he left

the NSA, taking his large collection with him of these things that he felt he had an obligation to reveal to the world because he felt that this was wrong. And so the devices covered by this are pretty much everything: mobile, Android and iOS devices; routers; Windows, Mac, Linux PCs; many IoT devices from smart light bulbs all the way up through televisions. And Kellyanne tells us also microwaves. So pretty much the works.

I will note, as many of our listeners did, that "The Gibson" and "SQRL" appear as accident...

Leo: What? I didn't see that.

Steve: Yes, "The Gibson" is in there, as is "SQRL."

Leo: What?

Steve: As accidental name collisions within the document dump. So, no, they're entirely different things.

Leo: [Humming "Twilight Zone" theme]

Steve: Okay. So the big takeaway is, as you said at the top of the show, Leo, and I completely agree with you, there really was nothing hugely new here. Imagine if someone were to listen to all previous 602 weeks of Security Now!, and after doing that, went out to find and collect everything that we discussed on the podcast. That's what you'd end up with. That's about what the CIA would have if they had done that. That is, as I was looking through everything, there wasn't a single thing I saw that we haven't talked about.

Leo: Really. Interesting. Wow.

Steve: Yes. It's all - so essentially what this is, you know, we've talked about how - I love the word "porous." I think that's exactly the right analogy because something which is porous, think of maybe like pumice or something, which is - it's not very porous, but it is a porous stone. So if you just put some water on it, nothing's going to happen. But if you pressurize water on it, you can force some molecules through. That is to say, if you really want it bad enough, you can make it happen.

And, I mean, that's the perfect model for today's security, unfortunately. Our security, and for all kinds of reasons, ends up being porous, even if it's social engineering. Unfortunately, social engineering works. You can trick someone into clicking a link. So, and if you didn't try to trick them, then they wouldn't have clicked it. But if you put pressure on them by designing something so that they will, then it can happen.

So that's where I think we are. To me, this whole adventure with the CIA clearly demonstrates, as I just said, one of the fundamental distressing realities of today's computing and communication technologies, which this podcast often highlights. When our defenses are inherently soft and porous, placing pressure upon them will cause them

to leak. Well-funded and highly motivated state actors such as the NSA and CIA can bring significant resources and thus a great deal of pressure to bear against the many technologies, none of which are particularly secure, or not absolutely secure, which we use in our daily lives.

And so, in looking at all the coverage of this and reading everything, I guess maybe perhaps the most controversial aspect of this is the notion that not a malicious hacker, but a taxpayer-funded organization would be discovering and concealing zero-day vulnerabilities in our systems, and keeping them to themselves for their own purposes and not disclosing them to their devices' manufacturers.

But other than that, I mean, everything we've talked about is stuff that they've got. So essentially they're just collecting. They're aggregating and deploying publicly disclosed, discussed in security forums, I mean, I'm sure they have people. In the same way that Brian Krebs has infiltrated and penetrated the discussion groups where a lot of the organized crime groups are, you have to know that law enforcement has people who have an online persona, and they're collecting the same sort of intel and information for their own purposes.

Now, in response to this, there's been a response from the industry, which is what we would expect. Apple has said that they've already patched "many," in quotes, WikiLeaks iOS exploits. Their formal statement was: "Apple is deeply committed to safeguarding our customers' privacy and security. The technology built into today's iPhone represents the best data security available to consumers" - which we believe - "and we're constantly working to keep it that way. Our products and software are designed to quickly get security updates into the hands of our customers, with nearly 80% of users running the latest version of our operating system. While our initial analysis indicates that many of the issues leaked today" - meaning last week - "were already patched in the latest iOS."

And again, that's what we often see is that some of these things are older, yet we know that, due to patch delays, they can also still often be effective. Apple finishes: "We will continue working to rapidly address all identified vulnerabilities. We always urge customers to download the latest iOS to make sure they have the most recent security updates."

And in an interesting little aside, just again, sort of a sense for how scattershot this also was, even the very popular Notepad++, which I use, fixes a CIA hacking issue. They released v7.3.3 with the title "Fix CIA Hacking Notepad++ Issue." And in their notes, this was a DLL hijack: "The following DLL hijack works for both the portable and non-portable variants of Notepad++. The issue of a hijacked DLL concerns" - and it was a DLL, S-C-I lexer dot dll [scilexer.dll], which is needed by Notepad++. And they go on. I won't go into the details. But this was - it's not their fault. Essentially, what was found among these documents, among many of these, was that the CIA had created a compromised version of scilexer.dll. And if they could arrange to swap theirs for the real one, that gave them a beachhead.

And so what happened was I'm sure people going through this trove spotted Notepad++ referred to - and I've got the link on the WikiLeaks page for anyone who's interested - and notified the guys at Notepad++, saying hey, guys, do you know the CIA has, like, got a replacement for one of your DLLs? And so what they did was they've cryptographically signed the real one now, and then they altered notepad++.exe to verify the signature.

Now, the problem with that is that that depends upon Notepad++ not being modified not to care about the signed signature of scilexer.dll. And this is not a remote vulnerability

anyway. So the point is, if anybody is going to have access to our local machines or have some means for replacing files on the machine, then there's all kinds of mischief they can get up to. But again, this is sort of like the CIA clearly, based on these documents, doing everything that they can, looking for any opportunity to get a wedge into our systems.

Leo: Was it your sense - you read through more of this than I did because there's 8,000 pages. But was it your sense that almost everything here required physical access to the hardware? I didn't see any remote exploits.

Steve: If there were zero-days, they may be able to get into things, yeah.

Leo: [Crosstalk]. Yeah, yeah.

Steve: Yeah. Certainly, for example, the much talked about TV spying on you. The Smart TV hack, that apparently required malware on a USB to be stuck in, essentially modifying the firmware of the television, so that it pretended to be off when it really wasn't.

Leo: A lot of this stuff is a stack. You start with a compromising tool, and then you use that compromising tool to add something else, to add something else.

Steve: Right.

Leo: So what seemed to be missing, certainly from the Samsung hack, but others, was just how would you get this on here?

Steve: Correct. And one of the ways to differentiate this from the NSA was, you know, that was a massive vacuuming experiment or exercise, where they were just sucking, you know, they were tapping into main Internet backbones and sucking everything in. This is much more targeted in nature. And, I mean, it shouldn't surprise anyone that this is what the CIA is doing. We want them to be doing it, not to us, but to other people. And we presume that they are.

Leo: NSA does signals intelligence generally.

Steve: Right.

Leo: The CIA is a spy organization. They're going after targets.

Steve: Right. And not surprisingly, they're using technology which is vulnerable, which we discuss every week.

Leo: I would expect them to. I mean, all you have to do is watch a Jason Bourne movie, for crying out loud.

Steve: Yes, yes. And so our bottom line, as I once indicated back in the time when we were talking about that Sony Pictures APT, the Advanced Persistence Threat breach, I would have a nervous breakdown if I was responsible for securing something as lumbering and massive as Sony Pictures.

Leo: Yeah. Or Yahoo, or...

Steve: It is, yes, it is simply impossible. And so the NSA/Snowden and the CIA/WikiLeaks disclosures demonstrate conclusively, if nothing else, that despite all of their best efforts, those high-end government intelligence agencies are unable to secure their own working assets.

Leo: Yeah.

Steve: They haven't. So how could they possibly be trusted with any explicit golden key which would allow them to access our encrypted communications? Past is prologue, and all of the evidence demonstrates that, even with the best of intentions - and I don't suggest they don't have the best of intentions. Well, I'll give them that. But U.S. law enforcement cannot be trusted, should not, must not be trusted with any sort of carte blanche backdoor access to the Internet's encrypted communications. Of course they want it. It's fine for them to ask. You can ask. But we must not capitulate.

As our experience with OpenSSL vulnerabilities continues to demonstrate, any sort of cryptographic monoculture is inherently dangerous. The system we have now, where a court order search warrant must be obtained and served, creates a heterogeneous system with distributed responsibility and built-in checks and balances. Companies like Amazon push back, and Google, and Apple, and so forth. Even requiring companies to be able to decrypt their customers' data, when they choose not to be able to, subjects customers to unnecessary risk. And besides, if the CIA has all of this now-demonstrated technology, they clearly don't need to ask for permission.

Leo: The other issue is, of course, not merely the golden key issue, but the Vulnerabilities Equity Processor, VEP, where the government had agreed a couple of years ago to, if they had an exploit, and it was leaking out, that they would immediately notify the companies. And there was a process where they could appeal. In fact, they were supposed to tell everybody every time, unless in the appeals process they could convince the VEP panel that they needed this flaw for espionage.

Steve: Ah.

Leo: So that's all out the window, by the way. I mean, I don't think, even in when it

was created in 2014, it was much adhered to. But this was the theory, is how do you handle this? How do you...

Steve: I'm sorry. I was going to say, and the FISA court never said no.

Leo: Right.

Steve: You know, it became a [crosstalk].

Leo: Yeah, I don't think it was FISA was involved in this. There was a panel including intelligence agencies, executive branch people. But I think that that's clearly not been, you know, the process hadn't been adhered to. And so really the big question is people are discovering flaws all the time. What is their moral obligation to reveal these? And the moral obligation gets higher if they can't control them. Right? It's one thing if, well, we could find flaws that only we would have access to them. It's another thing entirely if we can't keep a lid on them. Then bad guys are going to get them. And that actually hurts us, the citizens, because industrial espionage, hacking suddenly is empowered by tools created by our own government because they didn't reveal them to the companies responsible.

Steve: Well, and remember, too, in the case of zero-days, we only discover them when we discover them being used. Which really leaves the question open, what is going on that we don't yet know about?

Leo: Right. Well, that was one of the things the NSA said in the Vulnerabilities Equity Process, that's hard to say, VEP, was, well, we have telemetry. We'll know if these things have leaked out. We'll know if these tools are being used because we're watching for the fingerprints of these tools. Then, if we see that, oh, then we'll let you know, Apple.

Steve: In that case, I guess their alarms all went off last week.

Leo: Yeah.

Steve: Agh.

Leo: Well, to WikiLeaks' credit, they didn't reveal code; right? I think there was one inadvertent release. But all this code is being held onto. It'll get - it'll come out.

Steve: It'll be interesting to see what they do. WikiLeaks does have a tendency to a little bit overinflate what they're offering. And then some of the things they promise never comes out. I don't know what's going on behind that.

Leo: Right.

Steve: But it'll be interesting to see, over time, if it comes out. I'm glad, for example, that companies affected by this, like the Notepad++ guys, like Apple, are being proactive, going through this and making sure that anything that is now known publicly, they either have already dealt with or can deal with very quickly. But again, it's like, yeah, I mean, we want our CIA to be good and to be protecting us from bad guys. And if they hack somebody else's phone and are able to do that, well, that's their job.

Leo: I mean, were people watching "The Bourne Identity" and thinking, well, none of that could ever happen? Or were they thinking, yeah, this is probably the kind of capabilities the CIA has? I mean, they...

Steve: I've got to say, I have to say, I know our listeners watching "Mr. Robot" are like, yes, that happens, that happens, that happens.

Leo: Yeah, yeah. Well, that's because "Mr. Robot" had a pretty good team of advisers, including our guest from last week, Marc Rogers, who was just really good on Triangulation. Well, good. You didn't say anything I didn't expect, but I'm glad to hear your take on this. We've been waiting all week long to say, well, what, what does Steve say? What does Steve say?

Steve: Well, and I really do, I hope the takeaway from this is let's leave things the way they are. We do not need to make this easier. Nobody needs a golden key. They obviously already can get anywhere they need to. It's just this demonstrates that not only is all the software they're attacking having problems, but so are they. Their own software, I mean, their own networks and systems are having problems. And so we all have to do the best job we can of keeping our security as high as possible. Giving anybody some sort of unrestrictable access is just - it's a recipe for disaster.

Leo: Which is why we've talked so much about not giving your phone and your computer to the Border Patrol when you cross into the country because - we talked on MacBreak Weekly about data exfiltration, them taking everything off your phone. What we didn't mention is at that moment they also have the opportunity to maybe apply some of these handy-dandy hacks to your device.

Steve: Yup.

Leo: And observe on you and, you know, watch you from then on. Anyway, great subject. Thank you, Steve. We do Security Now! every Tuesday at 1:30 Pacific, 4:30 Eastern. New time because we're in summertime now, here in the United States, so the UTC time for this, I know UTC doesn't change, but we do, is now 20:30, if you'd like to watch live. We'd love it if you do, and join us in the chatroom at irc.twit.tv. You can tweet at Steve, @SGgrc, if you've got questions. With any luck, we might have a Q&A session next week. You never know.

Steve: Let's hope it's a quiet week.

Leo: You never know. Yeah, be nice if it was. You can also go to his website, GRC.com, and leave questions there at GRC.com/feedback. While you're there, pick up SpinRite, the world's best hard drive maintenance and recovery utility. And check out all the freebies Steve offers there, as well, including Perfect Paper Passwords and SQRL and the Healthful Sleep and all that stuff. It's all at GRC.com. He also has this podcast there, I should mention, 64Kb audio, and he is the unique purveyor of written transcripts of the show. So if you like to read while you listen, or you want a searchable text file that will jump you to the part of the podcast you'd like to find, that's where you get that: GRC.com. We have audio and video at our site, TWiT.tv/sn. You can watch us live. There's apps on every platform, or at YouTube Live at YouTube.com/twit, or download the show after the fact. We have on-demand also. And everywhere you go to get podcasts you should be able to find Security Now!. I think that does it.

Steve: My friend, until next week.

Leo: We'll see you later. Steve and I have a fight at the end of every show to see who gets the last word in. See you then, Steve.