SECURITY NOW!

Transcript of Episode #697

Zerodium

Description: This week we examine the intended and unintended consequences of last week's Windows Patch Tuesday; and, speaking of unintended consequences, the U.S. government shutdown has had some, too. We also examine a significant privacy failure in WhatsApp, another ransomware decryptor (with a twist), movement on the DNS over TLS front, an expectation of the cyberthreat landscape for 2019, a cloudy forecast for The Weather Channel App, a successful 51% attack against the Ethereum Classic cryptocurrency, another court reversing compelled biometric authentication, and an update on the lingering death of Flash, now in hospice care. We then look at a bit of miscellany and errata and finish by examining the implications of the recent increase in bounty for the purchase of zero-day vulnerabilities.

High quality (64 kbps) mp3 audio file URL: http://media.GRC.com/sn/SN-697.mp3
Quarter size (16 kbps) mp3 audio file URL: http://media.GRC.com/sn/sn-697-lq.mp3

SHOW TEASE: It's time for Security Now!. Steve Gibson is here. Lots of security news. He'll have an update on last week's Patch Tuesday for Microsoft Windows users. We'll talk about a major court decision when it comes to unlocking your smartphone. And, finally, who's willing to pay up to $2 million for zero-day exploits? Zerodium. We'll have all the details coming up next on Security Now!.

Leo Laporte: This is Security Now! with Steve Gibson, Episode 697, recorded Tuesday, January 15th, 2019: Zerodium.

It's time for Security Now!. Yay. The best part of your week has just begun, featuring this fellow right here, Mr. Steve Gibson, the host and moderator of our show. Hello, Steve.

Steve Gibson: And a big surprise this week, Leo. I know not only what episode number this is, but also what year it is.

Leo: Congratulations.

Steve: And those will all be correct on the show notes and in all communication going forward.

Leo: Yay.

Steve: Normally I have a hard time - well, not normally. Sometimes I have a hard time focusing down on a single topic. And actually there was some competition because there was a lot of interesting news this week. But the huge inflation of reward for zero-days paid by Zerodium - also sort of I thought this was apropos because we've been talking about SandboxEscaper, who seems so annoyed by the fact that she's unable to cash in her zero-days. It's like, what? Anyway, and again, we've talked about this before, but I wanted to kind of come back to it, especially in light of the fact that, well, I'll save it. I won't step on this. But the bounty has gone insane. And it really does sort of create an ethical question, I think, about the idea of an entity paying an insane amount of money for zero days, which is what Zerodium purchases. And it's like, okay. So anyway, we're going to talk about that.

But before we get to that, we've got some intended and unintended consequences of last week's Windows Patch Tuesday. And sometimes I'm feeling self-conscious that we don't talk about Patch Tuesday on Tuesday, which is when the podcast is. It turns out that the aftermath of what happens is almost more interesting than the news of what's being fixed. It's what the patches broke and what the remediation is for those. Because Microsoft, anyone who listens to Windows Weekly with Paul and you and Mary Jo on Wednesdays is knowing how they're now feeling about Microsoft's attempts to keep this sinking ship afloat. Anyway, so we've got intended and unintended consequences of last week's Patch Tuesday.

Also, speaking of unintended consequences, the U.S. government shutdown has had some, as well, that we'll talk about. We've also got a significant privacy failure in WhatsApp, which of course is based on the Signal protocol, which itself is excellent. But someone's found somebody else's communications in her new phone, and we look into and explain that. And we actually have a corollary to the TNO - Trust No One - slogan for this podcast as a consequence.

We've got another ransomware decryptor with a twist; some forward motion on the DNS over TLS front, that is, this general movement to move DNS away from unencrypted UDP over into encrypted and authenticated TLS or HTTPS. We've got two different takes on expectations of the threat landscape for 2019. And of course I couldn't resist saying a "cloudy forecast" for The Weather Channel app. I heard you also referring to them in the last week, so we'll touch on that. And then also one of the things that was competing for the title of this week's podcast is a successful, what's known as a "51% attack." In this case, it was against the Ethereum Classic cryptocurrency. We'll talk about that, and also a little bit of the background behind 51% attacks, which we haven't touched on for quite a while.

I heard you mention at the end of MacBreak Weekly something else that we need to talk about, which is another court reversing compelled biometric authentication. We have an update on the lingering death of Flash, now in hospice care. And then we'll take a look at a bit of miscellany, some errata, and finish by examining, as I mentioned at the top, the implications of this recent stunning and literally headline-pulling increase in bounty for the purchase of zero-day vulnerabilities. So I think another great week for our listeners.

Leo: That's Zerodium. That's what that's all about.

Steve: Yeah.

Leo: Zero-day-ium.

Steve: Right. So our Picture of the Week is just - it's one that I had in my collection of images that people have sent me over time. And it was apropos of one of the stories that we'll be talking about. I shook my head when I saw this because this is for a domain, RelianceCP.com, and the tab is Reliance Capital Partners. And I haven't gone there recently, and I don't know if this is the image you get. But it makes it very clear that they are still one of those old-school Flash-based sites. Apparently when you go without Flash, you get a big banner that says: "To enjoy this site" - because of course enjoyment is the goal - "you'll need to update your Flash Player. It's easy, painless, and will take just a moment."

Then, of course, they pretty much go against that by saying: "1. Download and install the latest version of Adobe's Flash Player." Then two starts with "Unfortunately," which is really not the first word you want in the second step of "easy, painless, and will just take a moment." But "Unfortunately, you'll then need to close your web browser," which means you are unable then to read steps three and four because your web browser is gone.

Leo: Memorize them. Memorize them.

Steve: Yes. Then "Go back to this site." And there is this whole issue of retention which they're sort of fighting against here because maybe they were hoping you would stay if you just came by mistake. But anyway, "Go back to this site after you restart your web browser." And then "4. That's it. Have fun." And of course that is if you then survive what it is that having Flash loaded in your system - because obviously you didn't have it before. And in order to go to RelianceCP.com, at least when this picture was taken...

Leo: No, no. I just went there. This is the current site.

Steve: No.

Leo: What's really frustrating, by the way...

Steve: Oh, boy. Wow.

Leo: No, that's it. What's really frustrating is you don't get anything if you don't have Flash. I mean, you don't get - they don't degrade gracefully. They just don't show you anything. So I don't know...

Steve: So again, we remember those days where somebody built an entire website in Flash. And so...

Leo: That's what this is, obviously; right?

Steve: Yes, exactly. And so the assumption was those browsers, you know, who wants text? We want Flash. And so, okay. And actually this - I was really tempted just to jump down, I don't even know how deep it is in our notes, because I talk about this. Well, we'll get there. Anyway, great Picture of the Week. And, boy, whoever you are, Reliance Capital Partners...

Leo: Really they don't want any viewers because I don't think - is there any modern system that has Flash on it by default? No. My Windows 10 machine doesn't. My Mac doesn't. No iOS device does. No Android device does. So basically what they're telling people is go away.

Steve: Yeah, really. If you're visiting this century...

Leo: I don't know who these guys are, but they mustn't get much web traffic.

Steve: I'll bet they're kind of thinking, gee, I wonder what happened? Maybe the Internet just kind of was a fad after all because no one's coming by anymore.

Leo: It's a real estate investment company focused on acquiring, repositioning, renovating, and stabilizing multi-family properties. I only know that because of the Google cache. Oh, boy.

Steve: Yeah, boy. Well, that's how to make sure you don't get any business.

Leo: Yes.

Steve: Okay. Last Tuesday was the second Tuesday of the month, the earliest second Tuesday possible being the 8th, and 51 vulnerabilities were fixed. So a good number of vulnerabilities for the first Patch Tuesday of the new year. Seven of them were rated critical. However, it also broke Windows file sharing for Windows 7 and its matched companion, Server 2008 R2. But we'll get to that in a second.

First the good news. There was a surprising DHCP vulnerability discovered internally by one of Microsoft's own guys in their enterprise security group, and it could allow an attacker to send a specially crafted DHCP response to a client to perform arbitrary code execution. Now, that's scary because DHCP is sort of promiscuous. I mean, it's on all the time if you are in any normal mode, especially if you have a laptop, and you're roaming around. When you go into the airport or the coffee shop or pretty much anywhere, it's DHCP which is reaching out and saying, "Hey, I'm somewhere new. Give me an IP. And it turns out that a malicious DHCP server could have been, for Windows systems, installing code on those systems.

They wrote: "A memory corruption vulnerability exists in the Windows DHCP client when an attacker sends specially crafted DHCP responses. An attacker who successfully exploited the vulnerability could run arbitrary code on the client machine. To exploit the vulnerability, an attacker could send a specially crafted DHCP response to the client." So the good news is they discovered it internally. They have no knowledge, as opposed to it being found by one of our AV companies in the wild being used, and so hopefully foreclosed this before anyone got bit by this. So that's good news about last Tuesday's updates.

They also found a pair of Hyper-V vulnerabilities which could have allowed powerful escapes from the virtualized containment. Microsoft wrote: "To exploit the vulnerability, an attacker could run a specially crafted application on a guest operating system" - meaning in one of the Hyper-V VMs - "that could cause the Hyper-V host operating system to execute arbitrary code." So not only an escape, that is, like poking a hole, but actually getting the external Hyper-V host to do something for you, so not good. And those are closed.

There were also three other critical flaws were patched in the ChakraCore scripting engine, which was failing to handle memory objects in Edge. And there was also a problem in their Jet database engine which was publicly disclosed, but had not been observed in the wild. So all those things got fixed, and that's good. In addition to, what, 46, 47, 48 others. However, as has become all too commonplace lately, included in last Tuesday's release were two updates that caused problems connecting to network shares on Windows 7 and its server version, as I mentioned, Windows Server 2008 R2. Three days later, that would have been on Friday the 11th, Microsoft released a standalone update to resolve that problem which had been introduced by Tuesday's patches.

So first of all, if you didn't have any problem, then you're okay. If you did, the two patches that caused the problems were KB4480960 and KB4480970. So you could remove those, or you could install the subsequent fix, which was 4487345. And if anybody hasn't known what's going on, you haven't dug around or found an answer, I do have the link to that update in the show notes.

There was also a registry fix, I mean, because this was, for a lot of people, this was a big problem. So there was some scurrying around. A quick fix was found that involved adding a DWORD value to a key in the registry under LocalAccountTokenFilterPolicy. If you did that in order to temporarily solve this problem, you should remember to take that out because you don't want to leave that in there, and instead apply this KB4487345 fix.

Also last week Windows 7 machines being activated through Microsoft's Key Management Service, KMS, began receiving "Windows is not genuine" notifications, indicating that the Windows license was not valid, which of course you can imagine upset some of the valid Windows users. It turns out that, due to its coincidence with Patch Tuesday, the initial suspicion was that something else that Patch Tuesday had done had broken this. But it turns out that we later discovered that there was a change that Microsoft had pushed to their activation servers that broke that, which Microsoft later backed out.

They wrote: "A recent update to the Microsoft Activation and Validation unintentionally caused a 'not genuine' error on volume-licensed Windows 7 clients that had" - and there was some interaction - "that had KB971033 installed." They said that the change was introduced at 10:00 UTC on January 8th and was reverted at 4:30 UTC on the 9th. So made a mistake, realized what had happened, and backed out of it. So anyway, as I said, it's almost more useful now deliberately holding off a week just to see what happened after our monthly Patch Tuesday, since Microsoft has, I mean, I don't think we've had one that's been uneventful for quite a while.

I mentioned unintended consequences to the U.S. government shutdown. Certainly, as our U.S. listeners know, unless you're really uninvolved with the news of the day, at the time of this podcast the U.S. government is partially shut down over a political funding dispute about the proper nature of the enforcement of security at our southern U.S. border. In a classic example of unintended consequences, a growing number, somewhere around 80 as of the most recent reporting and counting that I've been able to find, 80 different government websites are no longer accessible or have been marked as insecure connections because, during this government outage, their servers' TLS certificates have expired and cannot be renewed, apparently, until the government reopens.

I have a shot in the show notes here of a notice: "Your connection is not secure." And this was Mozilla. This was Firefox last night. It said: "The owner of [and the site is] ows2.usdoj [that is the United States Department of Justice] .gov has configured their website improperly. To protect your information from being stolen, Firefox has not connected to this website. This site uses" - here's another little twist that we'll get to in a second - "HTTP Strict Transport Security" - as we know is HSTS - "to specify that Firefox may only connect to it securely. As a result, it is not possible to add an exception for this certificate." So you the visitor using Firefox have no recourse.

And this U.S. DOJ site was not alone; .gov websites with expired certificates which are also on the HSTS preload list, including all of the U.S. Department of Justice .gov subdomains, are completely inaccessible by Chrome and Mozilla because the U.S. DOJ, wanting to strengthen their security, has been declaring historically and is on the preload list for these browsers. So you can only get to them this way. So naturally, first of all, I'm sort of surprised that this happened so quickly, but I guess certificates are constantly rolling over and needing to be renewed.

There was also a rockettest.nasa.gov site that is gone, and a Lawrence Berkeley National Lab site, d2l.lbl.gov. What's interesting is that I was curious about those because the rockettest.nasa.gov site, its cert expired on January 5th, and this Lawrence Berkeley National Lab site expired on the 8th. They initially showed warning messages like I read before. Oh, but they did not have HSTS, so you were able to push past the warning, saying, you know, I want to go there anyway. I know what I'm doing, or I'll take the risk or whatever. The problem is those sites also required authentication. And so if you pushed past that, you were then sending your username and password credentials without encryption to the site.

So even though it has apparently been impossible to issue new certs during the shutdown, at least the sites have been taken off the air. In the case of rockettest.nasa.gov, when I looked this morning, DNS had been pulled from it. And the ows2.usdoj.gov site had had its traffic blocked. So you can't get there at all. It's still resolving DNS; but they said, okay, we can at least pull the plug on this so that people are not trying to log in insecurely.

So clearly the takeaway here is that we should learn a lesson. We're having these temporary government funding shutdowns over political disagreements now, relatively routinely. I think this is the third one under the Trump administration. And we've broken a record in terms of its length. So certainly it's a function of the size of the window during which certificates cannot get renewed. If these are now breaking length records, there's a greater opportunity for certificates to expire during this time. The good news is we always see these approaching. So I'm hoping that this experience and the fact that we've moved the Internet forcefully to HTTPS connections, which is all for the better, and it's certainly good that sites are on the HSTS preload list so that, even when their certs expire, there's no loss of security, except for the fact that you can't get there.

But my point is that, since we always see these potential shutdowns approaching, it would be great if there was some look at the certs on the servers. Because we know that all of the certificate authorities will credit us with the amount of remaining time on a certificate when they are renewed. So there's no downside to renewing the cert ahead of time. It doesn't cost more money. You're not losing the amount of time that was remaining on the certificate when you renew.

So I hope that there's enough awareness of this that the people in the government, as we see the next shutdown approaching, will say, oh, gee, we only have three weeks left on this cert, and that may not be enough time. So let's renew now so that the site can stay on the air. Although, frankly, I think it was - I remember a previous one where several sites just put up a banner and said: "We're sorry. During the funding dispute and the government shutdown, our services are not currently available. Come back when you hear otherwise." So anyway, the good news is at least the sites that were offering you the ability to ignore security, at least in the two that I looked at, they had been taken down. They were off the air, which is an improvement.

An interesting case of WhatsApp security being broken. We've talked about, of course, a lot about messaging security. We did a podcast on the Signal app, whereas I have said a number of times, as I was reading through the detailed protocol spec, I remember thinking initially, boy, this thing is overdesigned. And then, as I got into the details more, I realized why the bullet point features that were mentioned at the beginning were there, and I came away with a lot of respect for the Signal protocol. The problem is it's still up to the implementer to deal with some of the details. And at least WhatsApp has failed in one way to handle some of this.

This came to light on the 10th, which was, what, last Thursday. An Amazon employee, Abby Fuller, tweeted: "Logged into WhatsApp with a new phone number today and the message history from the previous number's owner was right there. This doesn't seem right." And apparently there was - I don't know how many people followed her. The news got out. It drew some attention to her tweets. She followed up with additional tweets. She said: "Now I'm wondering how many other times it's happened. Like does whoever has my old number now have my WhatsApp history?" And she also tweeted in response to others: "Yes, it was a new device. No, it wasn't second-hand. It was not a second-hand SIM. Yes, I'm sure they weren't my messages or groups that I was added to. Yes, they were in plaintext. I'm sure it's my new phone number. It was not restored from a backup."

Okay. So we know what happened. The apparent leakage of someone else's WhatsApp messaging stream into Abby's phone should raise privacy concerns. As we know, WhatsApp uses our phone number as our authentication in lieu of username and password. The argument has been that WhatsApp only sends to that number, and so our phone is our authenticating device. So the fact that it just uses our phone and our phone number is not a vulnerability. But what exactly happens when phone numbers change hands? It's clear from an online FAQ that WhatsApp is aware of this issue. The problem is that its users aren't aware, and WhatsApp has made everything so simple and automatic that it's difficult to then ask users to pay attention to something that's far from obvious because its security implications have been deliberately hidden in order to make this system easy to use.

On their FAQ, I've got a link to it in the show notes for anyone who's interested, they have a subject, "Changing phone numbers and/or phones," and then the subhead "Changing your WhatsApp phone number. Before you stop using a particular phone number, you should migrate your WhatsApp account to the new number. For a simple way to do this, use our Change Number feature. By using this feature, you'll be able to migrate your account information, including your profile information, as well as your groups."

They say: "Make sure your contacts delete your old number from their phone's address book and input your new number, as it is a common practice for mobile providers to recycle numbers. You should expect that your former number will be reassigned." In other words, this is a complete failure of the privacy guarantees that WhatsApp is promoting as a consequence of the fact that it's phone number tied, yet people are not necessarily tied to their phone numbers when they change. So Abby's tweets indicated that the chat history she received on her new phone was "not full, but definitely actual threads/DM conversations," she said elsewhere.

So we know that WhatsApp doesn't archive messaging on their servers, but we also know that - and really WhatsApp is Signal because it's the Signal protocol. And this is something that we explained and covered when I talked about the Signal protocol on our podcast of that name. We know that undelivered messages will persist in encrypted form for up to 45 days. The other problem is that once a device's SIM and phone number have been used to establish the local device's encryption keys, the SIM can be removed. Yet that device, now absent any cellular telephony, can continue to use the encryption keys it still has, until such time as the phone number associated with its absent SIM becomes assigned to some other WhatsApp user.

So that means the binding, the real-time binding between the phone number and WhatsApp encryption is weak. I mean, there is no real-time binding. It's a first-use establishment. Which does create a rather large window during which time there's a presumption that you still have this phone number, even though it's on a device that may have no phone number at all, no cellular telephony service. And so that creates a rather glaring loophole. When you combine that with storage and catch-up delivery of pending messages, it creates an opportunity for some significant privacy leakage.

So this is the way WhatsApp operates. Oh, it also trusts new encryption keys broadcasted by a contact and uses them to automatically reencrypt undelivered messages and send them to the recipient without informing or leaving an opportunity for the sender to verify the recipient. Again, it's doing a lot of things behind the scenes so that it just works. Unfortunately, we're seeing a perfect example of how this could be broken. And of course this brings us back to my number one complaint about ease of use versus security and privacy tradeoffs, which we inevitably encounter anytime someone else manages our keys for us.

This made me go back and visit Threema. I haven't looked at the Threema website for a while. And I've always liked them because they keep this in the hands of their users. Yes, there's a little more setup in the beginning. You are asked to do - you remember that Threema's the one that has the green, yellow, and red sort of stoplight signal for the level of authentication of the other person's keys that you have achieved. So, yes, a little more setup. Also it's not free. It's a few dollars in order to purchase this.

Leo: And it's not open source.

Steve: Is it still not?

Leo: No. They use NaCl, but we don't know how they use NaCl; right?

Steve: Okay. Anyway, so all these...

Leo: Unfortunately, Signal also uses a phone number. I wish they didn't.

Steve: I know. And yes, and they do.

Leo: It's a drawback, yeah.

Steve: Yeah.

Leo: They don't have the same problem WhatsApp does, I'm sure. Because once the keys are invalidated, they say, well, that's that, you know.

Steve: Right. So as I said at the top of the show, if I were to coin a phrase to be a perfect corollary to our TNO, Trust No One, it would be, "If you're not managing your own keys, someone else is." Keys are the way, encryption keys are the way our systems work today. They need to be somewhere. And someone needs to manage them. Someone needs to create them, curate them, verify them, use them, destroy them. And if you're not doing that, then that's being done somewhere else. So again, it's a perfect corollary to Trust No One. If I'm using some sort of communications tool - and again, nothing I'm doing needs it. But if someone's really concerned about privacy, you want something where you're managing your own keys. And Leo, I would agree with you that it should also be open source and, as we know, have been audited, closely looked at by people who have an adversarial role.

Leo: They claim they're audited, but you have to sign an NDA to audit it. So, I mean, it's a commercial program. Let's just say it, call a spade a spade, yeah.

Steve: I thought I was going to have another story about a free, useful, ransomware decryption tool. We've been talking about those over the last few weeks. This one is PyLocky, P-Y-L-O-C-K-Y. It turns out - I got a kick out of the headline. It was "Unlock files for free." And I thought, okay, well, this is probably another encryption done wrong. Turns out that's not the case.

Leo: We should mention it comes from Cisco, from Talos; right?

Steve: Well, actually their research does. What they discovered was that, if you happened to be capturing your network traffic...

Leo: Oh, okay.

Steve: Uh-huh.

Leo: Well, there you go.

Steve: ...at the time that this thing grabbed a hold of your computer and had its communications with its command-and-control server, so that you had a PCAP file, a packet capture file of the network traffic between your machine and the command-and-control server, because the initialization vector and the password were transiting the network at the time, well, the good news is...

Leo: Yeah, we got it.

Steve: What's the problem?

Leo: Yeah, you've stored it.

Steve: Yeah, thanks very much, but I wasn't capturing my traffic. And you might argue also that anybody who is doing, whoever they are, who actually has Wireshark running or something capturing their packet traffic, is probably astute enough not to get infected with this in the first place. So I'm not exactly sure what the overlap is between people who are being caught out by this ransomware and happen to have packet capture running at the same time. To me this felt like, oh, hey, look, we were playing with this ransomware. We noticed that, oh, look, we could capture the packets and reverse the decryption. Okay. But that isn't really going to help anybody. And other than that, the encryption was done correctly.

On the 9th, which was Wednesday, Google Public DNS began to support DNS over TLS. So Google has added themselves to the ranks. We know that Cloudflare has been doing this for a while. Google now is there, too. In their posting, under their security Google blog, they said: "We implemented the DNS over TLS specification along with the RFC 7766 recommendations to minimize the overhead of using TLS. These include support for TLS 1.3," they said, "for faster connections and improved security, TCP fast open, and pipelining of multiple queries and out-of-order responses over a single connection."

And actually I would argue that that's probably one of the most important things to have for DNS because the user is going to be establishing a semi-persistent connection to a remote DNS server. But remember that, even though it's caching, you may very well be making queries for DNS domains that it doesn't have in its cache, meaning that it's going to have to go out and recursively resolve that query in order to get for you the IP address that you're asking for.

If you're going to a web page, where oh my god, you know, just scores of domains are now appearing on individual web pages, that creates a flood of DNS queries. So you're definitely not going to want to wait for an in-order resolution of DNS. You really need out-of-order resolution. So you need to be able to dump a whole bunch of queries onto that remote DNS server and start getting answers back immediately for IPs that it has in its cache while it goes out and looks for other things so that you can get those back and start making your queries and not be stuck in a serialized pipeline. So that's all for the good.

They also note - this is Google, of course - that Android 9 (Pie) users can use DNS over TLS today. So I dug down a little bit because I was curious to see where we are in the state of the deployment of this. And so what they have is known as a "stub resolver." The stub resolver is - essentially it replaces, or I guess I would say it's a local proxy for DNS. So it's a little resolver that runs on your whatever, on your smartphone in the case of Android. There are stub resolvers for other platforms I'll get to in a second. So what it does is it rewrites your DNS servers to itself, so typically 127.0.0.1, the local host. So it creates a local DNS server on your machine so that your machine then makes queries to it, which it then proxies out securely over to the DNS server, the DNS over TLS server that it's been configured with.

In their notes they explained that the stub resolver is configured with DNS over TLS, resolver name dns.google in their case. The stub resolver obtains the IP addresses for dns.google using the local DNS resolver because of course first it's got to get the IP of where it's going. It makes a TCP connection to port 853 - as we know, 53 is the normal port for DNS, so this is 853 - at one of the IP addresses that it has received from the normal DNS resolver. It then initiates a TLS handshake with the Google Public DNS resolver. And the Google Public DNS server returns its TLS certificate along with a full chain of TLS certificates which chain up to a trusted root certificate.

So the stub resolver verifies the server's identity based on the certificates that it receives over the TLS connection. If the identity cannot be validated, the name resolution fails, and the stub resolver returns an error. So naturally you want to make sure that you are connecting to the real dns.google server and that you're not subject to any sort of spoofing because the whole, you know, not only are you hoping to get privacy by running your DNS over TLS, but you're also wanting to solve the problem of DNS spoofing through any sort of a man-in-the-middle attack or interception.

As we know, DNS over UDP provides essentially no protection for that because, if someone can somehow get into your connection, it's trivial to spoof DNS. This solves that problem, but you want to make sure that your other end is anchored at the real trusting or trustable DNS server. And so after the TLS connection is established, the stub resolver then has a secure communications path between you and Google's Public DNS server over which these queries can be sent.

So we have DoH, which is the acronym or the abbreviation of DNS over HTTPS. Chrome and Firefox already support that. So that gives us already a lot of these benefits. But as we know, our computer systems are doing lots of DNS queries also. So DNS over HTTP supported by our browsers only solves the problem for browsing. DNS over TLS, if our OS has a DoT resolver, that is, DNS over TLS, then our whole system gets protected, and all of the DNS queries that are being made are going to be running over TLS. And so long as the DNS over TLS server is giving us good performance and is well connected and is nearby, it can run very quickly, and we get absolute authentication of the queries subject to its security and privacy so that nobody can see what we're doing.

There is DNSPrivacy.org for anyone who is interested. There are now resolvers, there's a stub resolver for Mac and Linux and Windows. They're still in the early stages of development. But this would allow somebody who really wanted that kind of privacy, now that we're seeing some good heavyweight support for DoT, DNS over TLS, to get privacy for all of their systems' DNS lookups. So yay.

Their paper is titled "A First Look at the Cryptomining Malware Ecosystem: A Decade of Unrestricted Wealth." And of course we're all familiar with the expression "Crime doesn't pay." The point I think is that getting caught eventually makes any crime which may have appeared to be going along nicely, right up until that point, suddenly a source of regret. That adage would certainly apply to those who are behind efforts to steal others' computing resources for the mining of cryptocurrency. We've been talking about cryptocurrency mining now for, well, browser insertion. For a while websites were voluntarily putting the script on their sites, like in lieu of advertising, to say, "Hey, while you're here, we'd like to borrow some of your processer or GPU in order to mine some coin for ourselves."

Anyway, these guys, two researchers - I'm trying to look for it in my show notes. I don't see where they are. I have their original research link in the show notes. In their abstract they said: "Illicit cryptomining leverages resources stolen from victims to mine cryptocurrencies on behalf of criminals. While recent works have analyzed one side of this threat, i.e., web browser cryptojacking, only white papers and commercial reports have partially covered binary-based cryptomining malware. In this paper, we conduct the largest measurement of cryptomining malware to date, analyzing approximately 4.4 million malware samples," they said, "one million malicious miners over a period of 12 years from 2007 to 2018.

"Our analysis pipeline applies both static and dynamic analysis to extract information from the samples, such as wallet identifiers and mining pools. Together with open source intelligence data, this information is used to group samples into campaigns. We then analyze publicly available payments sent to the wallets from mining pools as a reward for mining, and estimate profits for the different campaigns.

"Our profit analysis reveals campaigns with multimillion dollar earnings, associating over 4.3% of Monero with illicit mining." So, what, one in 25, over 4.3%. "We analyze the infrastructure related with the different campaigns, showing that a high proportion of this ecosystem is supported by underground economies such as pay-per-install services. We also uncover novel techniques that allow criminals to run successful campaigns." So what's the cash-out value of 4.3% of Monero? $53 million.

Leo: Wow.

Steve: Yes, exactly, Leo. Unfortunately, we know that money is what drives this. And if you've got a $53 million paycheck on the other side of figuring out how to get mining on other people's hardware, you have a lot of incentive to do so. And unfortunately, one of the themes we keep coming back to here is the degree to which security is porous. We've seen instances where, for example, where routers are compromised in order to install cryptomining on any browsers that they're able to perform an injection on behind the router.

So it's just, in aggregate, we're looking at $53 million spread out among these campaigns. And in fact in their outlook for 2019, Check Point Security sees cryptocurrency-stealing software continuing to be the number one most commonly distributed form of malware. That is, they have a Top 10 list, and all of the top slots in Check Point's Top 10 list are filled with cryptocurrency miners. Coinhive continues to be the most prominently distributed malware, followed by XMRig, both which use the victim computer to mine Monero with the profits directed into the cryptocurrency wallet of the attacker. Those two are followed by JSEcoin, which is a JavaScript miner embedded into websites; and then the not very imaginatively named CryptoLoot, which is a direct competitor to Coinhive. CryptoLoot was second only to Coinhive last November, but since then its distribution has dropped a bit.

So, I mean, this is where the pressure is at the moment. I guess it's somewhat better than it being ransomware, which is encrypting all of someone's files on their entire computer. Still, it's sort of nickel-and-diming people. If you get it into your phone, it runs your battery down. We've seen instances where it's causing phones to overheat because it's pushing them so hard. Those things tend to give away the presence of something that's gone wrong in one's computer. And then you scan it with some AV software and figure out what's going on.

But it does say that that's what the bad guys are trying to do is just get this stuff into your machine any way possible. And of course advertising is one of the ways this happens because our browsers are pulling in ads from everywhere, and those ads are then able to run JavaScript on our machine. And unless there's proactive measures being taken to prevent those ads from consuming undue resources, there's some slight chance that they're going to be mining cryptocurrency and score a fraction of a coin, or be part of a mining pool, in which case it's just incremental wins.

And speaking of Top 10 lists, Avast's Threat Landscape Report for 2019 is out. It was a 26-page report. I'm not going to go over the whole thing. But one topic that has been a big one for us all of 2018 in the Internet of Things section was a subsection on router-based attacks stating that the worst is yet to come. It may not surprise anybody, but we've talked about how porous our routers are and how unfortunately subject to compromise. In their report, I won't go through it in detail - if anyone's interested I've pulled the whole section out, it's in the show notes - because we've been covering it to such degree.

Their research, however, shows that 60% of users around the world have never updated their router's firmware. And while we can hope that newer routers are - and I want to believe that newer routers are doing a better job, there's this pressure from the routers to enable lots of features. And one of the trends we are seeing is routers becoming increasingly sophisticated, offering an increasing number of features, and those features ending up biting people.

So all of our listeners know you want to make sure you do not have Universal Plug and Play exposed publicly. You want to have it disabled on the LAN side because it can be abused. If you know you don't need it, if you can disable it, you want it disabled, as I mentioned last week, it really is worthwhile, especially if your router is a few years old, to go out and make sure that there isn't newer firmware available. And you really ought to, if you can, purchase a router from a reputable source. They mentioned, they said, they're talking about an increase in router-based malware in 2018. They said they've also seen changes in the characteristics of the attacks, where router-based malware has traditionally taken over a device for the purpose of carrying out a DDoS.

In other words, we talked about that earlier in 2018, where they were using UPnProxy in order to bounce packets off of routers in order to just distribute their attacks. The Mirai botnet was doing that. Avast said that today's attacks use malware that infects a device and then opens up a line of communication to a command-and-control server without taking any immediate action. They said: "We saw this with VPNFilter." Now, remember, that's the malware that the FBI alerted everyone to and said please reboot your routers, bizarrely enough, because if you did that, you would at least flush it out of RAM.

Leo: Temporarily.

Steve: Exactly, temporarily. Once the router's infected, these malware strains, they wrote, listen to the network traffic, fingerprint the network and the devices behind it, and allow for the command-and-control server to send new payloads or instructions to the devices. So basically it's elevated itself to an advanced persistent threat-style infection of a router. And they said: "In this, the malware acts more like a platform and less like a virus." They said: "This 'platformification' of IoT malware opens up many possibilities for bad actors who can repurpose it for a multitude of nefarious activities including pay per install, DDoS for hire, cryptomining, or even good old-fashioned spam."

They wrote: "This evolution replicates how PC malware counterparts have evolved and indicates the sophistication of new strains of IoT targeted malware." So I don't think it's possible to overstress the fact that we talk about attack surfaces a lot. One reason to be wary about AV, as we've discussed, is that it can present an attack surface, an increased attack surface, because if it's trying to scan everything coming in, if the AV itself has made any mistakes, it can increase the opportunity for compromise. Well, there is no larger attack surface than our router. That is the face of our network to the Internet. So absolutely keep it secure.

And I wanted to mention also that I was just shopping recently for - what I went looking for was Netgate's SG-1000, which is this cute little tiny box, no moving parts, just two network interfaces, a WAN and a LAN. What's significant about it is that it's a perfect platform for running pfSense, which is my favorite and chosen platform because it is completely open, open source, and it's running FreeBSD. And literally anything you can imagine that you might want to run on it, will. Anyway, I found that it had been discontinued, but it's been replaced with the SG-1100. The price is $159, so it's not a $49 piece of plastic.

But it is, I mean, it's what you want if you want something secure on your perimeter where you know what software is running in it and a platform that can grow with you. I'm using it to establish persistent VPN links using OpenVPN. But I'm also doing port mapping, translating from one port to another in order to avoid some of the things that my local ISP is doing, and a bunch of other things. Anyway, I just wanted to point people to it. The SG-1100 has got five times the packet processing performance of the SG-1000. So it turns out that the trade tariffs forced the price to go up a little bit, but you're getting five times the performance. It's got three 1GB interfaces. So if you're interested in setting up a segmented network, this supports that fully.

And it is also the first product equipped with Microchip's CryptoAuthentication device which provides assurance that the system is running authentic, unaltered pfSense software. So the software itself is signed by pfSense. And when you download it, it's verified and cannot be changed. So anyway, just on the topic of routers, I was poking around, looking for something. And first of all, I was sorry to see that the SG-1000 was gone, but I ended up with an SG-1100, which gives me an extra Ethernet interface and five times the performance, which is good because, as we know, our cable modems are increasing in speed.

Leo: So do you then hook this up to a WiFi access point? Or how do you...

Steve: Yeah, yeah. So the cable modem plugs into its WAN interface. And then I turn what was my WiFi router into an access point. Typically they're able to run either way. And I plug that into the LAN side in order to get access.

Leo: All the DHCP is performed by the SG-1100.

Steve: Yes, yes. And, for example, you can run a DNS server there. You can run DNS over TLS there. So it could provide - so it could be running your DNS over TLS proxy, connected to the strong DNS provider of your choice. And then all of your devices just use DHCP, get it as the DNS provider, and your entire network then is doing DNS secure. So, I mean, as an example of a perfect use for this. It's just it's a perfect little FreeBSD box. And there it is. It's a cute little thing, too.

Leo: It's not really a box. It looks more like an Altoids tin.

Steve: Yes, just barely big enough to hold the connectors that it needs.

Leo: That's really cool. So would you recommend - you'd recommend this over the EdgeRouter. I mean, this is what the EdgeRouter X does kind of. And it's got pfSense built in.

Steve: Yeah. I like - yes, it's got - this has a beautiful web interface. The problem with the EdgeRouter is that it really, I mean, you've got to really have your propeller wound tightly on your beanie in order to deal with the EdgeRouter. The EdgeRouter is very powerful, but it isn't easily configurable. This is really - you log in with a browser, and it's got a beautiful web-based GUI where you're able to fill out forms, create static and dynamic mappings, set up connections, I mean, it's just amazing.

It's got packet capture. You're able to do - you know how in Linux you have a top that shows the assorted list of processes by how much power, by how much processing time they're using. You have the same thing for your network so you can see what connections out to the outside world are using how much bandwidth in order to monitor what your entire network is doing. And, I mean, it just goes on and on and on. It is, for someone who wants to play with their network, I couldn't recommend anything better than this.

Leo: Oh, I might have to get it.

Steve: It's a cutie.

Leo: The only problem is the mesh routers that I use like to be their own DHCP server.

Steve: Well, and this doesn't prevent them from doing that. So you could [crosstalk].

Leo: But this is bridge mode. Or you could have double NAT.

Steve: You could have, well, so they want to be DHCP or NAT.

Leo: Oh. I've always used those interchangeably. They're not?

Steve: No. DHCP is Dynamic Host Configuration, which is giving machines IP addresses and so forth. NAT, of course, is stateful routing of packets across. Typically those are in the same box.

Leo: I understand the different functions, yeah, yeah, yeah.

Steve: But they wouldn't need to be, yeah.

Leo: So you have to let it do NAT for it to do the pfSense functionality. Otherwise it couldn't really, you know, do packet inspection.

Steve: Well, you could set it up as a bridge, in which case it would have some functions. But it would really, as you say, it would make more sense for it to run your network.

Leo: It should do the NAT, and then you let the Eero do the DHCP. I don't know if I can do that, but I'll try.

Steve: That would be interesting, yeah.

Leo: Yeah. A lot of these mesh routers want to see all the traffic because they want to do QoS. They want to be able to control it more directly, as opposed to just being a dumb access point or radio.

Steve: Yup, that does make sense. So I want to say it right up front that I get it that not everybody is going to be concerned about this. But as an indication of where practice and the law are colliding, I think this was interesting. And it does - there is a sort of a little bit of a creep factor in this also. And tongue-in-cheek I called this "A cloudy forecast for The Weather Channel App." The City of Los Angeles has sued The Weather Channel, claiming that it's been posing as a "personalized local weather data alerts and forecasts" app; but, they say, in truth makes profits by tracking users throughout the day and night, selling their private personal location data.

And I should say that's my favorite app. It's the one I use. I like it. Even though it's not necessarily super accurate, I'm a little bit seduced by the fact that you can touch Today and Tomorrow and see a guess about what the weather's going to be hour by hour, which is just sort of seductive. Anyway, this lawsuit, which was brought by, and I have it lower in my show notes, Domenic Venuto - no, no. He's on the other side. Okay, I'll just stick with what I have here. The lawsuit calls The Weather Company's practices "fraudulent and deceptive" and says they violate California's unfair competition law. TWC fails to disclose that it collects users' location data and sends it to third parties, the suit maintains.

So the City of L.A. says: "It isn't about analyzing the clouds above our heads for a personalized weather forecast. Rather, it's about collecting location data for 'advertising and other commercial purposes unrelated to weather data, alerts and forecasts.'" None of the marketing purposes of collecting geolocation data are disclosed on either Apple's App Store or Google's Android Play Store in their versions of the free app, which is also available in an ad-free version for $3.99, notes the lawsuit.

Now, first of all, I didn't know you could buy it for four bucks and be free of the ads. Now I'm tempted to do so because you can see how concerned I am about the...

Leo: I guess you don't care, do you.

Steve: About being tracked, yeah.

Leo: Take a look at Dark Sky on iOS. That's another very good app that does much the same thing. You know, IBM owns The Weather Company. This is an IBM company.

Steve: Yes, yes. And they're not happy about being sued by this. What was a little - the thing that was a little creepy is that in the lawsuit they did some digging, and they found that this Domenic Venuto, who is the general manager of the consumer division at TWC, admitted in an interview that: "If a consumer is using your product and says, 'Hey, wait a minute, why do they want to know where I am?' because it isn't an organic fit with the app, you're going to have some problems."

Leo: Wait a minute. It is an organic fit with the app. It's a weather app.

Steve: Yes, but that's the point. The whole idea is this is about tracking your location under the guise of providing you weather information.

Leo: But it needs your location to give you the weather information.

Steve: It does. So yes. But his point is that, well, okay. So what they found was...

Leo: I mean, I could see if, I don't know, a music app wanted to know your location, which by the way they all do. That you might be a little, well, why does it need to know that to play music for me?

Steve: Exactly. And so that's the point is he was acknowledging that by being a weather app, it's not going to raise suspicion that they want to know where you are. And again, so that was the point he was making. So last month The New York Times investigation found that The Weather Channel was - first of all, as we know, it's a big pack. As you just noted, music apps do the same thing - was one of at least 75 companies getting purportedly anonymous but pinpoint precise location data. And this is what I heard you mention when you were talking about this earlier, Leo, like within feet of where you're located, from about 200 million smartphones across the U.S.

In their coverage of it, they said they're often sharing it or selling it to advertisers, retailers, or even hedge funds that are seeking valuable insights into consumer behavior. In one example, a company known as Tell All Digital, which is a Long Island-based advertising firm, buys location data, then uses it to run ad campaigns for personal injury lawyers that it markets to people who are in emergency rooms. So the point is that the location data is that precise that they're able to determine if you're in an emergency room of a hospital and, if so, arrange - I'm sure in the whole advertising bidding deal - arrange to serve you personal injury lawyers ads because you may be in the mood for needing one at that time.

So anyway, I'll just note that iOS gives us good control over when we are feeding apps location data, and that you're able to say blackout location data completely for an app. Let it know where we are all the time, or only while we're using the app. And I presume that means only when it's in the foreground and has not been put to sleep by having been switched to the background. And for what it's worth, these things are very accurate. They're accurate to within a few yards and in some cases are tracking us 14,000 times a day, so basically creating a continuous stream of where we go over time. So again, Leo, as I said, I'm thinking of deleting it so I can purchase the $4 version and get it without ads. But if Dark Sky is doing the same thing, I'll probably switch to that. So it's Dark Sky?

Leo: Yeah, I like that one a lot.

Steve: Okay, cool.

Leo: That's the old Forecast.io. And, yeah, I think it's good, yeah.

Steve: Good, good.

Leo: In fact, I think you might even like it better, and there's no ads.

Steve: I'm glad we got that tip. I will check it out. Is it a for-purchase?

Leo: Yeah, I think it's a few bucks.

Steve: Oh, good. I'm happy to pay.

Leo: You know, if you don't pay, you've got to wonder who's paying.

Steve: Yup, exactly. Somebody's paying.

Leo: Somebody's paying.

Steve: Exactly. Okay. So the Ethereum Classic blockchain was hit - I think it began on the 5th, I've got it in the notes, and we'll come to it - with what is an expensive, known as "51% attack." When this podcast first described the detailed operation of the Bitcoin blockchain, in what was actually a classic podcast for us - because I remember I dug into it.

Leo: Oh, it was great.

Steve: And read Satoshi's original whitepaper. And I just was...

Leo: You were raving about it.

Steve: Oh, my goodness.

Leo: You thought this was amazing.

Steve: This is the coolest thing I've ever encountered.

Leo: And you know what? You were right because blockchain, you know, you can weigh in or out on cryptocurrencies and Bitcoin, but blockchain there's no doubt is a very valuable innovation.

Steve: Yeah, yeah. So we mentioned that one of the key assumptions, in fact the cornerstone assumption for the security and trustworthiness of any proof-of-work, blockchain-based technology is a large community pool of honest participants who mutually concur and authenticate blockchain events.

Leo: It says that? Because that really rules - if I'd seen that, I would have said, well, this will never work. Everybody's got to be honest?

Steve: Yeah. Actually, that's my jargon.

Leo: Oh, that's your line, all right.

Steve: Page 3 of Satoshi's original whitepaper, which was titled "Bitcoin: A Peer-to-Peer Electronic Cash System," it stated: "If a majority of CPU power is controlled by honest nodes, the honest chain will grow the fastest and outpace any competing chain."

Leo: Oh, that's intriguing.

Steve: So he understood even before this had happened, when it was just a whitepaper, that this was important. So stated another way, for the blockchain to remain secure, no single actor must ever be able to obtain a majority of the chain's total processing power because someone who is able to dominate the chain rules the chain.

Leo: Uh-oh.

Steve: Uh-huh. And is thus able to cheat others. And that's what has been happening since January 5th to the Ethereum Classic blockchain. At the expense, that is, they had to expend significant computation. But at the expense of significant computation, which was expended, attackers have been able to rewrite history. They rolled back and reorganized the Ethereum Classic blockchain and were thus able to double spend by recovering previously spent coins and transferring them to a new entity. I've got a link in the show notes to the Coinbase.com blog. Coinbase's security engineer, Mark Nesbitt, wrote in the blog about these events.

He said: "The function of mining is to add transactions to the universal shared transaction history known as the blockchain. This is done by producing blocks, which are bundles of transactions, and defining the canonical history of transactions as the longest chain of blocks. If a single miner has more resources than the entirety of the rest of the network, this miner could pick an arbitrary previous block from which to extend an alternate block history, eventually outpacing the block history produced by the rest of the network and defining a new canonical transaction history."

And that's what happened. They posted a timeline: "Late on the evening of Saturday, January 5th, our systems," he wrote, "alerted us to a deep reorg in ETC that contained a double spend. Our on-call engineers responded to the alert and worked to confirm the report through the night. We determined that we would temporarily halt send/receive interaction with the ETC blockchain in order to safeguard customer funds. This meant that customers who tried to send or receive ETC on Coinbase Consumer or Pro were unable to complete their transactions.

"On the morning of Sunday, January 6th, we posted an update on status.coinbase.com" - and Leo, you should go there - "stating that, 'Due to unstable network conditions on the Ethereum Classic network, we have temporarily disabled all sends and receives for ETC. Buy and sell is not impacted. All other systems are operating normally.' We performed an analysis on Sunday afternoon and evening to confirm the pattern and determine the key details of the double-spend attacks. Beginning Sunday afternoon, we observed eight more incidents, all containing double spends. Out of an abundance of caution, we did not put out a blog post prior to legal and technical review. A false alarm could have inadvertently caused market instability.

"On Monday, January 7th morning, after legal and technical review, we finalized our public analysis and posted to our blog and social media accounts." And of course that went into the news. I didn't put it in the show notes; it is there on status.coinbase. Or in this blog posting, the individual breakout of double spends, which ended up - I thought I had it. Oh, yeah, here it is. 219,500 previously spent coins were re-spent, netting the attackers $1.1 million in ETC. And on that page that you showed, of note is the fact that right now, today, everything is green except that one. ETC is still - their buy and sell, or their send and receive is still disabled, and they have it down for "maintenance." But really what's happened is...

Leo: We don't control it anymore. It's gone.

Steve: Yes. That Ethereum chain can no longer be trusted.

Leo: Now, this is not - this is a fork of Ethereum, Ethereum Classic.

Steve: Yes, it's the classic.

Leo: Didn't affect Ethereum. However, the real question is, obviously it's a smaller cryptocurrency. Could this happen to Bitcoin or one of the bigger ones? It would be awfully hard to do; right?

Steve: Yes. Historically, it's happened in the past. There was a 51% attack, I mean, these are generically called "51% attacks" on blockchains for exactly this reason. As Satoshi observed, you have to have a majority of honest nodes involved in validating these transactions. If any one entity obtains majority control, they're able to take over. And in the past we've touched on this where it's been like at risk. There have been single entities that sort of were approaching 50%, and that got people worried. And there were brief instances where someone had more than 50%. They had a majority. But as you note, the bigger these get, the more they sprawl, and then the more difficult it becomes for any one actor to obtain a majority. So there is safety in size because it just becomes an insane amount of processing power in order to pull this off. But it can happen.

Leo: Steve Gibson, back to you.

Steve: So a court recently ruled that we needn't give law enforcement the finger, I mean, our finger.

Leo: Our finger. Or face. Or iris.

Steve: So Thomas Brewster, who is Forbes' cybersecurity reporter, yesterday ran a story with the headline: "Feds Can't Force You to Unlock Your iPhone with Finger or Face, Judge Rules." A California judge has ruled that American law enforcement - Thomas wrote "cops" - can't force people to unlock a mobile phone with their face or finger. The ruling goes further to protect people's private lives from government searches than any before, and is being hailed as a potentially landmark decision.

Now, of course we've talked about this at length, where the standing law before was you could not be compelled to divulge something you knew like a password because that was being called "testimonial." But something you were, like your thumbprint, your iris, whatever, was not testimonial. So under that argument, you could be compelled to produce your fingerprint to unlock a device. So this order - and again, this is the most recent order. There is no last word until this thing goes to the Supreme Court for final judgment.

This order came from the U.S. District Court for the Northern District of California in the denial of a search warrant for an unspecified property in Oakland. The warrant was filed as part of an investigation into a Facebook extortion crime in which a victim was asked to pay up or have an embarrassing video of the publicly released. The police had some suspects in mind and wanted to raid their property. In doing so, the feds also wanted to open up any phone on the premises via facial recognition, a fingerprint, or I don't know of any phone that uses an iris, but that was in there, too. While the judge agreed that investigators had shown probable cause to search the property, they did not have the right to open all devices inside by forcing unlocks with biometric features.

On the one hand, the magistrate judge, which was Kandis Westmore, ruled the request was overbroad as it was neither limited to a particular person nor a particular device. So they were just saying, whatever we find of whoever it happens to be, we want to be able to see inside it. But in a more significant part of the ruling, Judge Westmore declared that the government did not have the right, even with a warrant, to force suspects to incriminate themselves by unlocking their devices with biological features.

And as we know, previously courts had decided that biometric features, unlike passcodes, were not testimonial. That was because a suspect would have to willingly and verbally give up a passcode, which is not the case with biometrics. A password was therefore deemed testimony, but body parts were not, and so not granted Fifth Amendment protection against self-incrimination.

So then this created a paradox. How could a passcode be treated differently from a face or finger when any of the three could be used to unlock a device and expose a user's private life? So where we are now is that Judge Westmore focused there on her ruling. She's declared that technology is outpacing the law, writing that fingerprints and face scans were not the same as physical evidence when considered in a context where those body features would be used to unlock a phone. So in other words, specifically saying that, for the case of unlocking, these are being held as different.

So anyway, as I said, this is a flip from where we've been. But various judges, as we have seen, in various part of the country, make various decisions based upon their particular reading of the case and specific examples. I think it's going to take the Supreme Court ultimately to make a decision as we move forward about what is and is not usable for unlocking our devices. And who knows the way it'll come down. We're seeing judgments now on both sides.

Firefox 69 will finally disable Adobe Flash plug-in by default, in which case our Picture of the Week company is pretty much SOL. They're finally going to have to have somebody design a regular website for them, rather than asking people to go download Flash in order to use their site. The Flash plug-in is the last remaining NPAPI. The NPAPI is, believe it or not, the Netscape Plug-in Application Programming Interface. Yes, Netscape, brought to us back in 1995 with Netscape Navigator 2.0. It was later adopted by other browsers.

The developer.chrome.com site says of NPAPI: "NPAPI plug-in support for extension has been discontinued. The documentation below is preserved for historical purposes only." So Chrome wants nothing to do with this. And at this point, Flash is the only thing still using the NPAPI, which Firefox reluctantly continues to support. Over on developer.chrome it says: "Warning: NPAPI" - and I'm not making this up, it says - "is a really big hammer that should only be used when no other approach will work. Code running in an NPAPI plug-in has the full permissions of the current user and is not sandboxed or shielded from malicious input by Google Chrome in any way. You should be especially cautious when processing input from untrusted sources, such as when working with content scripts" - gee, like a web browser - "or XMLHttpRequest," the XHR, the standard way that JavaScript reaches out and performs queries, doing AJAX and so forth.

"Because of the additional security risks NPAPI poses to users, extensions that use it will require manual review before being accepted in the Chrome Web Store." So it's like, yeah, it's still there. But really, try not to need it. So of course this is why continued Flash support is so inherently dangerous. Flash uses the NPAPI, which is non-sandboxed, which means that anything that gets into your Flash uses one of its many security vulnerabilities - I mean, we know they've still got to be there because anytime anyone looks, someone finds some - would allow malicious activity in your browser. So it's not sandboxed because sandbox didn't exist, and you could argue was not needed, back in 1995. We were just lucky that our computers booted back then. And sometimes they didn't.

Leo: Right.

Steve: So once Flash has been disabled by default in Firefox, users will not be prompted to enable Flash. But even then they will be able to activate Flash on certain sites using browser settings. So this crazy company does continue to operate, is able to operate through 2019. The final step for Flash on Firefox is due in early 2020, when Adobe also officially end-of-lifes Flash and it is completely removed from the consumer versions of Firefox. Flash will continue to be supported in the Firefox extended support release, the ESR version, until the end of 2020. And in 2021 Firefox will refuse to load the plug-in completely.

So Microsoft will also be disabling Flash by default in Edge and IE in mid to late 2019, so mid to late this year. Google will be disabling Flash by default in Chrome 76, which is due for stable release around July. Chrome users will be able to enable Flash in settings, but the plug-in will require explicit permission. And then, as of Chrome 69, users need to give permission for each site to use Flash every time the browser is restarted, which is another nice deterrent. So I'm reminding everyone of this. I mean, the boom really is being lowered on Flash.

I'm reminding everyone because, aside from the security win of removing this longstanding nightmare from our browser ecosystem, every time I mention this I receive notes and email from our knowledgeable listeners within various enterprises who are still, who today remain seriously dependent upon this creaky old technology. And it's not its age that I have a problem with. I'd still be using Windows XP if my machine hadn't died and forced me to 7. So it's not age that I have a problem with. It's that it has always been buggy as hell, and Adobe never really cared to expend the time or energy to fix it. They just kept patching it as people kept poking holes in it, and people kept getting hurt by it. So its existence has hurt countless innocent Internet users, and the sooner it dies, the better.

So those knowledgeable listeners of this podcast whose enterprises - the people who write me every time I talk about this, go tell somebody that, if this is mission critical, someone's got to rewrite this for you. And what often happens is I hear about they're using something that, yes, it's 15 years old, but they lost the source code, or the only people who knew how it worked are gone or died or who knows what.

Leo: Happens all the time, yeah.

Steve: We've got these apps in COBOL, and we need to update them. It's like, okay. Anyway, so, I mean, at some point it's just, well, I mean, it looks like you could continue to use it through next year if you really have to, but not past that.

Leo: I mean, if you've got an Intranet, you could force people to use old browsers and stuff like that; right?

Steve: Yeah. Although you'd have to make sure they didn't go reach out to see if there was an update because the browser would go, whoa, am I old.

Leo: Yeah, we have that problem. We have a banking app, a check reader app that we've got to keep using old Mozilla.

Steve: Yeah. So I got a piece of errata, Leo. Mike D., I actually saw his note in the Security Now! newsgroup at GRC. The subject was: "Steve, it is pronounced 'Gee-Drah.'"

Leo: Yeah, that's what I thought. The "H" is silent.

Steve: Yes, "Gee-Drah." He says: "Love SN, have been listening from the beginning. In terms of the current episode, clearly you are not spending enough time watching bad cinema," he says, "<chuckle>. It - Ghidra, King Ghidra, King Ghidorah, and/or Monster Zero..."

Leo: Don't. Just stop right here. Stop trying to read those names.

Steve: Thank you, one of the monsters from the same lineage of Japanese monster movies as - oh, Godzilla. Godzilla I know how to say.

Leo: Yeah, yeah.

Steve: "Mothra and many others. It appears in several of those movies and even takes on Godzilla in one."

Leo: Well, well.

Steve: Okay, I'm not being sucked into this. I'm not going to go find out.

Leo: Ghidra is good. Oh, no.

Steve: But now we know it's Ghidra. And also I forgot to mention last week, I had it in my notes - well, actually it was one of those things that - remember that my system was dead last week. It was in fact the processor. I've killed two of them. I'm no longer overclocking because, you know...

Leo: No more.

Steve: Even though nothing was getting hot, everything seemed fine, but it took about six months, and then it was the death knell. So what I had forgotten in my notes was just a little note that you'll remember that when Windows 10 Disk Cleanup utility added the Downloads folder checkbox, I cautioned our listeners that, be careful, because we may have all had the habit of turning them all on, as I certainly had. But make sure that, if you were turning the Downloads folder on, you intended to have it delete all your prior downloads, or at least all those that were in the Downloads folder. I got a kick out of the fact that this was apparently causing so much trouble that there is now a warning dialog that pops up in Windows 10 Update to the Disk Cleanup if that's turned on. So yes, it was biting people.

And my last little piece of miscellany before we talk about Zerodium was from our friend, Leo, Evan Katz.

Leo: Oh, yes.

Steve: Who said: "P.S.: Yes, Filemail is amazing, and the best large transfer service that exists," exclamation point. "I have used it for years."

Leo: Good to know.

Steve: And I certainly trust Evan and his opinion on this. I only had that one experience I mentioned because I had to move a nearly 6GB VM to Denmark, and I was just stunned by the fact that it saturated my upstream cable modem. I've never seen anything do that for an hour or two at 33Gb. It just - it was amazing. So anyway, Evan, thank you for the confirmation. And I've got someone whose opinion I trust who has used it to move things around for a long time. So yay.

Okay. I've been biting my tongue, not wanting to let the cat out of the bag because this is a ridiculous amount of money. And who are we fooling about who are buying these? I had to dig, in their own FAQ, like all the way to the bottom, every other possible question they had answered themselves. But first let me step into this from the front door.

So their website proudly proclaims "We Are Zerodium, the leading exploit acquisition platform for premium zero-days and advanced cybersecurity capabilities." Then their slogan is "We pay BIG [all caps] bounties, not bug bounties."

So, okay. Under "Our Exploit Acquisition Program," they say: "Zerodium is the leading exploit acquisition platform for premium zero-days and advanced cybersecurity vulnerabilities. We pay big bounties to security researchers to acquire their original and previously unreported zero-day research. While the majority of existing bug bounty programs accept almost any kind of vulnerabilities and proof of concepts, but pay very low rewards, at Zerodium we focus on high-risk vulnerabilities" - meaning, and I'm adding this, the juicy ones - "with fully functional exploits, and we pay the highest rewards." And I'm going to skip the number here for a minute.

"Eligible research: Zerodium is currently acquiring zero-day exploits and innovative security research related to the following products." And so it's pretty much all of the mainstream everything everyone uses. Operating systems we've got Microsoft Windows 10, 8.1, and servers. Oh, look, but not Windows 7. Good. Apple macOS, Mojave, High Sierra. Linux, BSD, CentOS, Ubuntu, et cetera. VM Escape, VMware. Web browsers: They want remote code execution, or sandbox escape and bypass, or both in Google's Chrome, Microsoft Edge, Firefox, Tor browser, Apple Safari. For clients and files, remote code execution or sensitive information disclosure from MS Office files, Word, Excel, PowerPoint.

PDF readers: Adobe or Foxit. Email clients: Outlook and Thunderbird. File archivers: WinRAR, 7-Zip, and WinZip. Smartphones: Apple iOS, Android, Blackberry, Windows 10 Mobile. Web servers: Apache, Microsoft, Nginx, PHP and ASP, OpenSSL, mod_ssl. Email servers: MS Exchange, Dovecot - never heard of that one - Postfix, Exim, and Sendmail. Web apps and panels: cPanel, Plesk, Webmin; WordPress, Joomla, Drupal; vBulletin, MyBB, phpBB; IPS Suite, IP.Board; Roundcube and Horde.

And, finally, research and techniques: Any other security research, exploit, or technique related to WiFi, Baseband RCE; routers, IoT remote code execution; antivirus remote code execution, exactly the attack surfaces we were talking about, AV; Tor de-anonymization. Okay, who cares about that? Well, we know. Mitigation bypass. All the notable mobile brands are listed by name. Eligible Linux/BSD distributions, all of them. Eligible routers brands, all the biggies: ASUS, Cisco, D-Link, Huawei, Linksys, MikroTik, Netgear, TP-Link, Ubiquiti.

They said: "Note: If you have zero-day exploits for other products or systems not listed above, feel free to submit minimal details, and we will gladly discuss the opportunity."

So now what brought the news was last Monday, January 7th, under New Payouts Highlights, they said: "Payouts for the majority of desktops, servers and mobile exploits have been increased. Major changes are highlighted below." So get this. An Apple iOS remote jailbreak, meaning zero click, with persistence, now $2 million.

Leo: Whoa.

Steve: $2 million.

Leo: To whom would that be worth that much?

Steve: Exactly. Exactly. If these guys are paying $2 million and making a profit.

Leo: Unless they're - maybe they're a front, though, for a nation state. Maybe they're not resellers.

Steve: We don't know.

Leo: We don't know.

Steve: That's a very good point. If you need one click, if you have an Apple iOS remote jailbreak which does require a click, well, you used to be able to get a million. But, no, now it's 1.5. So $1.5 million if you're not quite skilled enough to do this with zero clicks, but you can do it with one click. You could still get $1.5 million. For WhatsApp, for iMessage, or SMS/MMS remote code execution, $1 million, doubling what it was previously, $500,000.

A Chrome remote code execution with LPE, that's of course Local Privilege Elevation, for Android, including a sandbox escape, used to only net you $200,000, now half a million. Safari with a local privilege elevation on iOS, including a sandbox escape, same story. Used to only be 200 grand, now half a million. A local privilege escalation to either kernel or root for Android or iOS jumps, it's doubled, from 100,000 to 200,000. And a local PIN or passcode or Touch ID bypass for Android or iOS went from - oh, this is, in terms of percentage, a big jump, from 15 grand to 100,000.

Leo: Wow.

Steve: They said: "Note: Payouts were also increased for other products, including remote code elevation via document and media, remote code execution via man in the middle, ASLR or KASLR bypass information disclosure, et cetera." And then on the server/desktop side - that was all mobile. And you'll note that because it is the hardest to get into, Apple iOS is at the top of that pack at $2 million for a zero-click or 1.5 for a one-click. Chrome is down at 500,000 and Safari at the same. So getting yourselves into Apple iOS, that's still the crown jewel. And, wow, two million.

On the server or desktop, a Windows remote code execution with zero clicks via SMB - they give an example - or remote desktop protocol packets has doubled from half a million to one million. A Chrome remote code execution on the desktop and a sandbox escape, including - so you can do something useful once you get out or once you get code to execute, that's doubled also from 250 to half a million dollars. Apache or MS IIS, so either of those two major servers, a remote code execution, remote exploit via HTTPS requests - good luck with that, but maybe - doubled, quarter million to half a million dollars.

Outlook remote code execution from 150,000 to 250,000. PHP or OpenSSL remote code execution went from 150 to 250,000. MS Exchange Server, 150 to 250. VMware, VM escape, a guest-to-host escape has doubled from 100,000 to 200,000. And Windows local privilege escalation or sandbox escape, okay, now, this is exactly what SandboxEscaper had. She had a Windows local privilege escalation and sandbox escape.

Leo: She blew it. She could have been rich.

Steve: Went from 50,000 to 80,000.

Leo: That's a lot of one-man tents you could buy with that.

Steve: That's a lot of hikes you could do out in the wilderness.

Leo: Yeah.

Steve: Yeah, I don't get it. Anyway, so they say of their payouts: "Zerodium payouts for eligible zero-day exploits range from $2,000" - I don't even know what that is, that's probably, I mean, there's nothing here that's less than $50,000, so I guess that's got to be easy to make two grand. Anyway, "up to $2 million per submission. The amounts paid by Zerodium to researchers to acquire their original zero-day exploits depend on the popularity and security level of the affected software and system, as well as the quality of the submitted exploit - full or partial chain, supported versions/systems/ architectures, reliability, bypassed exploit mitigations, default versus non-default components, process continuation, et cetera. For more information, please read our FAQ.

"The payout ranges listed are provided for information only and are intended for fully functional, reliable exploits meeting Zerodium's highest requirements. Zerodium may pay even higher rewards for exceptional exploits and research." And I love they had this chart that I put into the show notes because it shows the Zerodium submission process.

We start with "You discover a high-risk zero-day vulnerability and manage to exploit it." Then, "You submit minimal technical details about your research to Zerodium." That's Step 2. Step 3, "Zerodium confirms its interest in the research and sends a pre-offer." Step 4, "You submit the full technical details and exploit to Zerodium." Step 5, "Zerodium evaluates the research and sends the final acquisition offer." Step 6, "You accept the Zerodium offer," so have a party.

Leo: And profit.

Steve: "And receive your payment within one week."

Leo: Fast. Fast payments.

Steve: And then 6 leads back into 1 because, having gone through this circle once...

Leo: Reinvest.

Steve: The cycle of life. You start looking for the next really bad $2 million.

Leo: SandboxEscaper needs to get to work here.

Steve: Yeah. Now, exactly. I don't get what's going on with her. Make some money, honey. Okay. So we had, down in their timeline, they noted September 19th, 2018. I wanted to dig around a little bit to see what more I could find out about who they are. September 19, 2018, so late last year they wrote: "We are acquiring pre-authentication remote code execution exploits affecting the following Routers: ASUS, Cisco, D-Link, Linksys, MikroTik, Netgear, TP-Link, and Ubiquiti. Exploits leading to authentication bypass or credentials disclosure are also accepted. Exploits relying on cross-site scripting or cross-site request forgery are not eligible." So here they are proactively soliciting a specific class.

Leo: We have a client who would like to buy.

Steve: Exactly. Exactly.

Leo: Is this illegal in any way? I mean, is it...

Steve: I mean, it really raises my hackles because it just seems like it ought to be. On December 20, a few months later, late last year: "We are currently looking for code execution exploits via USB drives on Windows and/or macOS. The exploit must achieve code execution immediately after the USB key or drive is plugged into the system without relying on visible keystroke injections or user interaction."

And so I read these, and then I went looking, I went digging in their FAQ. And all of the first 20 questions are people rubbing their hands together about how they get paid and can you transfer directly into my Swiss account and so forth. Anyway, finally, down near the bottom: "How the acquired security research is used by Zerodium."

Leo: Oh. How is it used?

Steve: Uh-huh. They say: "Zerodium extensively tests, analyzes, validates, and documents all acquired vulnerability research and reports it, along with protective measures and security recommendations, solely to its clients subscribing to the Zerodium Zero-Day Research Feed." And that was the second to the last question. And then we wrap with "Who are Zerodium's customers? Zerodium customers are mainly government organizations in need" - they need them, Leo - "in need of specific and tailored cybersecurity capabilities and/or protective solutions to defend" - that's right, it's like the U.S. Department of Defense - "to defend against zero-day attacks. Access to Zerodium solutions and capabilities is highly restricted and is only available to a very limited" - yeah, those with deep pockets - "number of organizations. Zerodium does not have any sales partner or reseller. Our solutions are only available through our direct channel."

So, yeah. I guess, I mean, I would love to get some sense for the level of activity in this channel, even if we have nothing else, if just an alarm kind of went "bong" every time one of those transited, it would be interesting because we don't know what's going on. I'm sure part of this research is, I mean, certainly the discoverers are locked up in an NDA. They can't share with anybody else.

Leo: Right.

Steve: In return for what they have discovered. If a government organization is paying big bucks to be on the Zerodium feed, they're being careful with it because its entire value to them is only to the degree that it remains unknown to the world. I mean, gee, what could you do with a USB drive that ran code on a Windows or Mac when you just plug it into a USB without requiring any keystrokes?

Leo: Hmm.

Steve: Don't know. Or why could you want to take over an Apple iOS mobile device with either zero or one keystroke? Hmm. How could that come in handy?

Leo: Wow, that's really interesting.

Steve: Really, yes, to me it just seems bizarre that there is a marketplace now, I mean, an out-front, in-your-face, we will pay $2 million. I mean, the thing that's a little annoying is that we know how porous security is. The harder you look, the more you find. And $2 million is a heck of an incentive for someone to dig down into somebody else's code and see if they can find a mistake somewhere.

Leo: Yeah. Yeah.

Steve: Wow. Wow.

Leo: Wow. Well, there you have it, ladies and gentlemen. Fascinating. It's fascinating. We have no idea who they - and when did they appear?

Steve: They've been around for a few years now. We've talked about them several times. And so an increase in bounty probably means that the government feed, that may have increased in number or in cost, so that they can afford to pay more. It might mean that security is tightening up, so they're having to incent the "researchers," unquote.

Leo: That could be, too. It could be supply and demand. If the supply shrinks, maybe...

Steve: Yeah.

Leo: Yeah. They might have to pay more.

Steve: So it's like, please look harder because we really do want these. Our customers are clamoring.

Leo: It also means there's competition from other sources like the companies themselves. Remember Apple for a long time refused to do this because they were afraid of ratcheting up the marketplace.

Steve: Well, Leo, Pwn2Own, I mean, Pwn2Own is developers basically saying, "Hey, I got a laptop," rather than $2 million, or for a lesser exploit, 200,000. "I got a really cool sweatshirt." Okay.

Leo: I owned this Macintosh, and all I got was this T-shirt.

Steve: That's right.

Leo: What is your judgment? Is it harder to find these than it has been in the past, or easier?

Steve: Yes.

Leo: It is harder.

Steve: Yes, yes.

Leo: Because we're all aware of it, and companies are working much harder to protect security, Microsoft chiefly doing a lot to lock its operating system down. Yeah.

Steve: Yeah.

Leo: Well, what a great subject, and thank you for another great show. I appreciate it. You'll find Steve at GRC.com, the Gibson Research Corporation. That's where he does his main work, which is of course SpinRite, the world's best hard drive maintenance and recovery utility. And he also gives away a lot of stuff. ShieldsUP. He talks a lot about passwords there. There's also health information and of course the latest on SQRL. Soon, soon, SQRL will emerge into the world. The little baby will be born.

Steve: A little bushy little tail.

Leo: Bright-eyed and bushy-tailed SQRL. Steve also has the podcast there, audio and transcripts of every show at his website, GRC.com. You can get them there or get them from us at TWiT.tv/sn. We also have video. And he can be reached and followed at @SGgrc, that's his Twitter handle, @SGgrc. You can DM him there, too, if you've got a tip.

If you are interested in watching us doing the show live, you want the latest, freshest security as we bake it, so to speak, security news, you can go to TWiT.tv/live every Tuesday around 1:30 Pacific, 4:30 Eastern, that's 21:30 UTC. And you can join the folks watching live in our chatroom at irc.twit.tv. But as I said, you can always get on-demand versions on our site, on Steve's site, or your favorite podcast application. Subscribe, and you'll get it automatically. I mean, it's available, and you can listen at your leisure.

Steve, have a great week. We'll see you next time on Security Now!.

Steve: Thank you, Leo. I'm sure the week will bring us lots of new things to talk about next week. So till then, bye.

Leo: Juicy stuff.