RDP: Really Do Patch

Description: This week we primarily focus upon the almost certainly impending doom of the Internet, as the Windows Remote Desktop Protocol saga finishes out its second week with a great deal of news and new evidence-based expectation for the end of humanity as we have known it. Okay, well, maybe it won't be quite that dramatic, but it already makes last year's Meltdown and Spectre flaws seem quaint. But before we get to that, we take a look at the FIVE new zero-day exploits just dropped by SandboxEscaper, Google's discovery and confession of 14 years of cleartext password storage, Microsoft's just-released Win10 Feature Update 1903, Firefox's release 67, and some interesting new data about the prevalence of validly signed malware.

High quality (64 kbps) mp3 audio file URL: http://media.GRC.com/sn/SN-716.mp3
Quarter size (16 kbps) mp3 audio file URL: http://media.GRC.com/sn/sn-716-lq.mp3

SHOW TEASE: It's time for Security Now!. Steve Gibson is here. We'll talk about the rise of ransomware, the return of SandboxEscaper, and the new Firefox 67 and why you might want to install it. And then Steve will announce the demise of the Internet. Yes, it's all next on Security Now!.

Leo Laporte: This is Security Now! with Steve Gibson, Episode 716, recorded Tuesday, May 28th, 2019: RDP: Really Do Patch.

It's time for Security Now! featuring Mr. Steven "Tiberius" Gibson, the man in charge, the guy who knows all, tells all, sees all. He's for the last, what is it, what did we say, 13 years been covering the security scene on Security Now!.

Steve Gibson: No, no, we're close - well, okay. We're closing in on the end of 14.

Leo: Holy cannoli.

Steve: Yeah, two more months. Elaine corrects me whenever I'm wrong. And it's August. This August we finish year 14 and just go cruising right into number 15.

Leo: Lordy, lordy, lordy.

Steve: If, that is to say, Leo, if there is still an Internet.

Leo: You think there's a chance there won't be?

Steve: Well, given what is apparently about to happen, we'll see. Today's podcast 716 of Security Now!, RDP stands for Really Do Patch. We're going to primarily focus upon the almost certainly impending doom of the Internet.

Leo: Well, wait, hold on there.

Steve: As the Windows Remote Desktop Protocol saga finishes out its second week with a great deal of news and new evidence-based expectation for the end of humanity as we have known it.

Leo: You've been reading too much Peter F. Hamilton. I'm just going to say it right now.

Steve: Now, maybe, maybe it won't be quite that dramatic. But it already makes last year's Meltdown and Spectre worries seem quaint by comparison.

Leo: Yeah, yeah.

Steve: But before we get to that we're going to take a look at the five, count them five, new zero-day exploits that were just dropped by our hiking friend and photographer SandboxEscaper.

Leo: Oh.

Steve: You remember her, Leo.

Leo: Oh, yeah, yeah.

Steve: She's got that really cool little one-person tent thingy that I thought was so cool.

Leo: She's the one who was really miffed.

Steve: Oh, believe me, she is. In fact, I forgot to obscure the F-bomb in her tweet that I have a screenshot of in the show notes. When I was just scrolling through it before we began I thought, ooh, I didn't blur that. She's not doing well.

Leo: Oh, boy.

Steve: Except, boy, is she, like, a gifted hacker. Anyway, so we're going to update everybody on that. We've got Google's discovery and confession of their 14 years of

cleartext password storage. Microsoft's just released, finally, Windows 10 Feature Update 1903, and we'll talk about the new features there. And how to get it because none of my machines wanted to get it by themselves. So there's a way to kind of give it a little kick, for those listeners of ours who want to play with it.

We also have the release 67 of Firefox with some new features, and some really interesting new data about the prevalence of validly signed malware. Our listeners will remember that I was annoyed when I had to replace a perfectly good, but now expired after three years, Authenticode signing certificate with a new one because none of the AV knew the new certificate. Even though it was from me, hello, they didn't care. And we're going to look at, like I didn't realize this was as bad as it was, and it won't surprise anyone to know who the number one certificate provider is for malware. More than half of the malware is being signed by certificates from just one provider.

Leo: Wow.

Steve: So, and we already know who, but we'll get there in time. So I think another great podcast for our listeners. And really, you know, enjoy the Internet while it lasts.

Leo: Well, you know, on the bright side we'll all have more time off.

Steve: We really...

Leo: I'm going to buy some books.

Steve: We'll talk to friends that we haven't actually seen in the physical presence because there won't be any Internet soon.

Leo: Holy cow.

Steve: So, yeah. So if you, like, wanted to download something, get it now because you won't be able to.

Leo: You know, the ratings are fine. You don't really need to puff this stuff up. I'm just saying.

Steve: Leo, Leo, we're going to be gone. We won't be able to do a podcast.

Leo: That's true, with no Internet.

Steve: It'll be - this is, like, 716. Remember this podcast, 716. I thought I was going to make it to 999. No. There will be no Internet to convey this quality content.

Leo: Holy camoly. Well, we'll find out what the heck he's talking about in just a second. All right. Why is the Internet over, Steve?

Steve: You know, I was just thinking, it's sort of a self-fulfilling prophecy because, if the Internet's gone, we won't be able to deliver the podcast, but we'll also have nothing to talk about.

Leo: Right. So it's perfect.

Steve: It's like, yeah, it's just sort of like, okay, it's a wrap. Okay.

Leo: That should have been the title of the podcast. It's a wrap, folks.

Steve: It's a wrap.

Leo: See you in Fiji.

Steve: Our Picture of the Week sort of surprised me because I realized I'd had it in my backlog for so long that it was now no longer current. What this shows is - it's really sort of interesting.

Leo: It looks like the London Underground map.

Steve: It does. And it takes a while to sort of parse it visually. This is the ransomware timeline up to just shy of 2017. So, first of all, it doesn't contain the last two years that have been, if anything, even more busy. But you can see where it began, what, like almost in the middle of 2012 was the first instance. Then we have 2013, maybe six or seven more. 2014, maybe twice that many. 2015, oh, it's beginning to catch on. People are thinking, hey, I can make some money. And then, boy.

Leo: Boom.

Steve: Month by month, January, February, March, April, May - and it bounces back and forth. May, June, July, August, September, October - 2016 is just insane. And I would imagine we really couldn't show 2018 and 2019 because the print...

Leo: It wouldn't fit.

Steve: ...would have to be so fine that you wouldn't be able to read it. You would just think...

Leo: Crazy.

Steve: Okay, don't know what that is. But anyway, I mean, those are all individual instances of encrypting ransomware which have landed on the industry, and people have been hoping to make some money at. So unfortunately, they're doing it enough to keep it happening, as we know.

So we talked, I think it was, what, last year? SandboxEscaper is this person's Twitter handle, and I've never seen a reference to their first name. There's a blog which is no longer public that you and I checked out at the time when we were first covering her work. And we discussed this person a little bit. It wasn't clear what this person's birth sex was or how they wanted to be referred to, so we're just saying "her" because that seems like the right thing based on conversations in the industry. But this person is a gifted hacker who, unfortunately, is also sort of unhappy with the industry. We know they love hiking and camping and photography. That was what the blog was full of. That and zero-days that they were dropping onto the world without giving Microsoft, apparently, first of all, at the time it seemed, without being interested in earning any money from them. Maybe that's changed a little bit.

So anyway, so the blog is currently closed to the public. Her GitHub, which was active a week ago, has been closed after putting five zero-days, exploits against Windows, out in public. Being zero-days, we know that means that there's no notice. So a week ago tomorrow, so that's last Wednesday the 22nd, apparently in a low moment she posted on Twitter. And I have the image of it. Wednesday, May 22nd, 2019 she says: "There's two more bugs on GitHub. Eff this" - and she didn't say "eff" - "this shitty industry. I don't plan to make a career in it anyway. I hate all the people involved in this industry. Everyone just thinks they know better. Everyone just loves pointing fingers. Bunch of apes. Bye." Posted by SandboxEscaper at 2:47 p.m. So as I said, not having a good day.

So in this very few short days, what we've received from this admittedly gifted hacker are five brand new zero-day exploits against Windows 10. One turns out to have been already fixed, which she later confirmed. But that still leaves us with four that are currently unpatched. And they were dropped, presumably on purpose, right after, I mean the day after this month's Patch Tuesday, thus creating the biggest problem for Microsoft.

Although I ought to also mention these are not, like, end-of-life problems. These are not like the Remote Desktop Protocol problem. These are privilege elevation discoveries that she made which are definitely a concern because, as we've seen, when those are combined with other exploits that would otherwise be limited to the privilege and the context of the current user, this allows them to escape that and get into the kernel and make themselves persistent and so forth, do dramatically more damage. Because we really do depend upon the privilege controls in Windows for our security.

So the security industry, I mean, we sort of don't know which way to turn, whether it's this or the Remote Desktop Protocol problems. She whipped the industry into something of a froth with her release of these because she also preannounced that more were coming. So in looking around the industry, I sort of landed on Davey Winder's thoughtful reporting for Forbes which I'll share because he sort of explains it and puts it in context. We've got two pieces, one last Tuesday and then one the following day, on Wednesday.

On Tuesday he wrote for Forbes: "New Windows 10 Security Exploit Can Read All Your Files - What You Need to Know." He writes: "A security researcher with a history of releasing zero-day exploits for the Windows operating system has struck again, this time just days after the latest Patch Tuesday security updates were rolled out. Which means that it's unlikely there will be a fix for Windows 10 users until June 11 at the earliest. So what did SandboxEscaper just drop into the Windows threatscape, what are the risks, and is there worse to come?"

He says: "A security researcher going by the name SandboxEscaper has posted a proof-of-concept demo for a Windows zero-day exploit online. This local privilege escalation (LPE) exploit is the fifth in a series of zero-days that SandboxEscaper has dropped into the Windows environment over the last year." So fifth, considering the ones we've already talked about previously, not some that were even going to happen after this reporting.

He said: "The latest proof of concept doesn't enable anyone to actually access your computer, but it does provide a method by which those who do so can upgrade their system privileges to an administrator level and in so doing grant them carte blanche to your data. SandboxEscaper has previously used the Windows Task Scheduler tool" - which is the one we most recently discussed - "for nefarious purposes, and this latest zero-day is no exception. It uses it to import and run a malformed task file that exploits a vulnerability in the way Task Scheduler handles discretionary access control list rights for such files without permissions, giving full control to any user, rather than just the system admin who would normally have it." Oh, and it was initially released for 32 bits, but it's been confirmed to work under 64 bits also.

He says: "So what was the motivation? As mentioned, SandboxEscaper has a reputation for releasing exploit code without any prior disclosure to Microsoft. Reporting on one of these last year, Forbes contributor Marco Chiappetta suggested that 'depression may have been a factor in SandboxEscaper's decision to post the exploit,' and quoted her as saying, 'I screwed up, not Microsoft. They are actually a cool company.'" And then finishing with "Depression sucks."

"However, in her latest blog postings announcing the new exploit, SandboxEscaper writes, 'I don't owe society a single thing. Just want to get rich and give you efftards in the West the middle finger. I'm donating all my work to enemies of the U.S.'" Then he says: "Make of that what you will. The timing is also interesting as it comes straight after the monthly Microsoft update cycle, which means it leaves the window of exploit opportunity open until June 11, when the next cycle is scheduled." And of course, as we know, when something has really been bad, Microsoft has broken their own schedule in order to push out something more quickly. I don't think these rise to that level, in my own opinion. You know, local privilege escalation is not good, but it's also not RDP.

So, he asks, "Is there worse to come?" He says: "It appears that this isn't going to be the last we hear from SandboxEscaper either. In that same series of blog posts, she says that she has four more unpatched zero-days. 'If any non-Western people want to buy LPEs,' she writes, 'won't sell for less than 60K.'" And then he quotes Ian Thornton-Trump, head of security at AmTrust International, saying he told him during a conversation this morning that as far as the economics of selling exploits are concerned it's kind of a, well, he said "sh*thead" move. "You can understand why, as Microsoft is known for having a pretty generous bug bounty program." We've talked about that recently, in fact. It is pretty good. Although, again, I don't know that these would rise to the level of being a high payout, which researchers use to cash in on their findings without taking the criminal route to riches. He said: "It's sad that folks burn the opportunity to contribute to the information security community."

So what can we do to mitigate the risk? Given that it's unlikely, based on responses to the previous exploits released by SandboxEscaper, that we will see any patch to this zero-day until the next Patch Tuesday, and I concur with that opinion, he says: "What can you do to mitigate the risk?" He quotes, again, Thornton-Trump from AmTrust, saying: "I will tell you that anything that interacts with the task scheduler is going to be pretty unsubtle and fairly easy to detect." So in other words, maybe an update to Windows Defender and other third-party AV would be enough to catch these things in the short term, until Windows is able to go through a full cycle. And that's sort of the pattern that we're beginning to see now.

He says: "Of course, that doesn't mean it will be an impotent threat, and zero-days must always be considered a very real and present danger to data." That said, Davey quotes Thornton-Trump as saying he isn't panicking over this, and pretty much I agree.

So then the following day he posted on Forbes: "Rogue Security Researcher With Grudge Against FBI Goes on Windows 10 Exploit Spree." The beginning of his posting basically reprised what he had written from the day before, which I've cut out. And then he goes on, saying: "Things have just become a little more problematical with SandboxEscaper releasing on May 22nd another two of the four remaining zero-days she claims to be in possession of. The first is similar to the local privilege escalation exploit released on the 21st, but this time exploiting a vulnerability in the Windows error reporting service. It's harder to exploit, and SandboxEscaper admits as much, to the point of conceding it's 'not that much of an issue.' It is, however, still a vulnerability that can be exploited, and others could well find more efficient methods to do so until it is patched.

"The second zero-day targets IE11, specifically allowing for the injection of malicious code. Again," he writes, "this would not seem to be a critical vulnerability as the proof of concept code appears to reveal that it isn't a remote exploitation threat, but rather something a threat actor with access to the machine could use to disable Internet protected mode for further attacks." So again, something that might be chained with other exploits, not itself a huge problem.

He says: "I think that these zero-days are all worrying, but not critical, as they all require the attacker to already have access to the target system, or possibly use these exploits alongside a remotely executable one that amounts to the same. As such, the immediate threat to most users would appear quite low." I don't know that I would go quite that far. For myself, remember that we've recently seen several examples where a local privilege escalation vulnerability has been combined with one or more different exploits in a chain to create a quite significant threat.

For example, in March, we talked about Google's report that a then-unpatched local privilege escalation vulnerability in Windows was being used in combination with an unrelated exploit in Chrome. On its own, neither exploit was able to do much damage, thanks to the mitigations built into both Windows and Chrome. But together, the exploits allowed hackers to remotely execute malware of their choice. So again, they couldn't have done that were it not for the combination of both. So when you pair a privilege escalation vulnerability such as what SandboxEscaper published, with something else that also itself doesn't seem to be a problem, you know, that's what we're beginning to see.

For example, I think, if I recall, every single one of those Pwn2Own major cash awards involved a chain of several exploits, cleverly put together, in order to get ownership or code running on the target machine. So that was a comment I made at the time is that we're no longer seeing a world where just one problem - well, except in the case of this remote desktop protocol that we'll be talking about a lot here shortly. Normally it is a chain of these sorts of things. And we also have a malicious side of the industry that wastes no time jumping on these things. So you've got to know that the instant these zero-days dropped, there were bad guys immediately looking to see how they could be leveraged while they were still useful and had not been blocked.

So anyway, Davey continues and finishes, saying: "That said, there are still two more exploits in the SandboxEscaper arsenal." Although since we haven't seen them, maybe she did find buyers for them, who knows. He says: "And we will have to wait to see what they bring when, and suggest it isn't going to be if," he says, "they are released." Although maybe she didn't find buyers, as I said. "Given the events so far this week I suspect we won't have long to wait."

And he says: "The motivation behind the release of these exploits doesn't seem to be financial. The exploits themselves are not without value to both vendors and threat actors alike, but given their relatively low threat impact, probably wouldn't be worth a fortune in bug bounties" - and I agree with that - "or if sold via an exploit broker. There are clues in the SandboxEscaper blog as to the real reasoning, and they are not subtle. The motivation would seem to be getting back at the U.S. for a perceived injustice." He wrote, and this is the result of his digging, and I was unable to get into the blog because by the time this was all public, the blog - actually she took her blog private and posted that it was for now access by invitation only. So who knows what was happening.

But he writes: "The most telling is the confession that she has 'most definitely given portions,' she says, 'of my work,' meaning her work, 'to people who hate the U.S.' because 'that's what happens when the FBI subpoenas my Google account and intrudes my privacy.' SandboxEscaper goes on to suggest that the people who have access to the exploits 'are going to use those bugs to get back at U.S. targets,' before finishing with 'an eye for an eye.'" And then he writes: "It's not just the FBI and the U.S. that are on the receiving end of this apparent hatred. Some of it is reserved for the information security industry itself." And then he quotes that posting that I had quoted at the top of this. So anyway.

So then finally, in an update of that - that was the 22nd. So then on the 24th, on Friday, he said, he updated his posting, said: "SandboxEscaper has now confirmed that the Windows error reporting bug was apparently patched this month, and so that's one fewer to worry about. Unfortunately, she also now has released two more zero-day exploits," which he enumerates, also local privilege elevation. So that makes a total of nine exploits, eight of which are zero-days released across the last 10 months by SandboxEscaper.

He says: "It also marks the end of the exploit spree, at least for now, as there is no further information to suggest she has any more exploit bombs ready to drop." He writes: "I would also like to add that mental health issues in the information security industry are rife; and, reading her blog entries, it certainly appears that depression has played a part in SandboxEscaper taking this destructive path with her undoubted abilities. I sincerely hope," he says, "despite what she has done, that she can get some help with all this and find some inner peace."

And of course, as I had commented, I remember taking a very close forensic look at some of her earlier work. I think it was the task scheduler piece that she did where there was a detailed analysis of it. And I remember saying on the podcast how very impressed I was. I mean, it was some beautiful work. And so sad that this is how she's chosen to use her very obvious skill. So that's the way it is.

It turns out that, following in the steps of Facebook and Twitter, Google has become the latest technology giant to confess that it accidentally stored G Suite enterprise passwords unprotected in plaintext on its encrypted servers. I have a link in the show notes to their blog post. The blog post, published last Tuesday, was titled: "Notifying administrators about unhashed password storage." This was posted by Google's Suzanne Frey, who's a VP of Engineering for Cloud Trust.

She wrote: "Google's policy is to store your passwords with cryptographic hashes that mask those passwords to ensure their security." I cut out a bunch of her explanation about hashing of passwords because we all know about that. She said: "However, we recently notified a subset of our enterprise G Suite customers that some passwords were stored in our encrypted internal systems unhashed. This is a G Suite issue that affects business users only. No free consumer Google accounts were affected." And she wrote: "And we're working with enterprise administrators to ensure that their users reset their

passwords. We have been conducting a thorough investigation and have seen no evidence of improper use or access of the affected G Suite credentials."

She says: "In our enterprise product, G Suite, we had previously provided domain administrators with tools to set and recover passwords because that was a common feature request. The tool, located in the admin console, allowed administrators to load or manually set user passwords for their company's users. The intent was to help them with onboarding new users." For example, a new employee could receive their account information on their first day of work, for example, and so already have an admin-set password in it, and then also for account recovery. "The functionality to recover passwords this way no longer exists."

She said: "We made an error when implementing this functionality back in 2005," thus the 14 years. She said: "The admin console stored a copy of the unhashed password. This practice did not live up to our standards." Certainly Google knows about hashing passwords. So, yes, just a mistake. She says: "To be clear, these passwords remained in our secure encrypted infrastructure." So the point being, although they themselves were not hashed, the servers themselves were encrypted. However, of course, the concern is that employees who would have access to the server content unencrypted would then also have access to the passwords in plaintext.

So she says: "This issue has been fixed, and we have seen no evidence of improper access or misuse of the affected passwords. In addition, as we were troubleshooting new G Suite customer sign-in flows, we discovered that starting in January 2019" - so just the beginning of this year - "we had inadvertently stored a subset of unhashed passwords in our secure encrypted infrastructure. These passwords were stored for a maximum of 14 days. This issue has been fixed; and, again, we have seen no evidence of improper access to or misuse of the affected passwords. We will continue with our security audits to ensure that this is an isolated incident." And so on.

So as we know, this can happen. We recently talked about Facebook "discovering logs" of unprotected passwords for hundreds of millions of its users, both Instagram and Facebook. And nearly a year ago, Twitter reported a similar security bug that unintentionally exposed passwords for its 330 million users in readable text on its internal computer system. So I guess we have to hope that, for the sake of our whole industry, that our whole industry's legacy behavior will be fixed. So like, you know, everybody will know you just don't do this. You hash passwords. That we cannot mistakenly have code that leaves passwords lying around in plaintext.

And of course hopefully where we will end up being is adopting systems eventually like SQRL that we'll be looking at very shortly in some more detail, which in the first place gives websites no secrets to keep. That's one of my favorite slogans for SQRL is "It gives websites no secrets to keep." They get a public key that they don't even have to keep secret because there's nothing malicious that can be done with it. So I have a feeling we'll be migrating in that direction sooner or later.

As I mentioned at the top of the show, Microsoft has released Windows 10 version 1903, which is also now formally called the May 2019 Update. This is unlike the monthly security updates. This is the one which they do twice a year which adds features to Windows 10. None of my systems - I Skype over a Windows 10 machine, and I have a couple Windows 10 machines set up in VMs. I also have a couple laptops. Unfortunately, as I've mentioned before, when I discovered the long-term servicing channel, the LTSC version of Windows 10, I immediately fell in love with it. I mean, it makes Windows 10 bearable because you're not having Candy Crush Saga put onto your Professional Windows 10 setup. So but of course the long-term servicing channel, I mean, the whole point of it is that you have an install that Microsoft is committed to providing security

updates for over the long term, rather than forcing you into this every six month update cycle. So of course you explicitly cannot do feature updates on those.

So while covering this, I updated my Windows 10 machines, went to Windows Update, brought them current, and then pressed Check for Updates, and nothing happened. So we know that, although it is officially released, Microsoft is throttling its availability, so it may not be available. I will get to a URL. I created a bit.ly shortcut that takes users directly to a small downloadable from Microsoft which will trigger the on-demand update of Windows 10 to version 1903.

Leo: Ooh, don't do that. No, no, no. Don't do that. Why would you want to do that?

Steve: We have listeners, I'm sure, who are going to want to play with 1903.

Leo: Microsoft, no, my strong advice, if Microsoft does not offer you 1903, it's not necessarily because they're staging it. It may also be because there's hardware compatibilities. They check issues, and they don't install it or offer it to machines that are not supposed to run it. When you override that, well, good luck.

Steve: Well, this does not override that, Leo.

Leo: Oh.

Steve: This performs all of those tests. And this is the officially sanctioned Microsoft link.

Leo: Oh, I see.

Steve: Yes. It's possible to dig down through some menus and get there, but it's just, I mean...

Leo: I still wouldn't do it. I would wait until Microsoft offers it to you, honestly.

Steve: Yeah.

Leo: Why not? Unless you're, like, I have to have the newest thing. But you know better than anybody how dangerous that is.

Steve: Yeah. So I have it installed on some VMs. It's not on any production machine of mine. So anyway, it is there. You can ask Windows Update for that. They call it the Windows Update Experience, which is the way you're able to get that. And so you can click a couple buttons in order to ask Microsoft for it. Because, I mean, it is officially available now.

Okay. So what's new? We get the Windows Sandbox, which is probably the biggest new feature of it. It's a lightweight virtualization sandbox which allows a Windows 10 - this is

Pro - let's see. It's not Home. And it's not Enterprise. It's Pro. There's two versions of Windows, Pro and one other. I can't remember the name of it now. Oh, it says "Pro and above." It requires a 64-bit processor. You do have to have virtualization enabled in the BIOS. And it's going to eat up a lot of RAM. Windows 10 will run in 4GB, but when you launch this it immediately starts to take up about the same amount. So you probably need to have 8GB of RAM in the system.

Microsoft is billing this as a way for power users to run things in a true Windows 10 VM that they don't want to allow have access outside of the VM. There are some downsides, which are that every time you start this, it starts with a fresh image. So I guess the good news is nothing that you do in there persists. But that can also be a pain, for example, if you wanted to check Word or Excel extensions or add-ons or downloads. Well, you're not going to have any apps installed in that whenever you run that. It starts blank every time. Which is, again, sort of a mixed blessing. It does use the Hyper-V system. So if you install this sandbox, then it will fight with VMware and Virtual Box that want to use the same virtualization features on your system.

And I guess Microsoft must figure that, well, somebody who is a VM user, like who's using a full VMware or Virtual Box, well, they're probably not the profile of the user who might want to use this Windows 10 Sandbox. But it's there. The way you get to it is in the features and settings. You click that, and it brings up a list of all of the installed features under Windows 10. And if it's available, you'll see that Windows Sandbox is listed down toward the end of that list. When I first fired it up in a Windows 10 VM, it was grayed out because I had not enabled virtualization for that VM. So I shut down, turned that on, then I was able to install Windows Sandbox and play around with it and get some sense for how it works.

Anyway, that's pretty much the biggest new feature. There's a light theme for Windows 10, sort of the flipside for those who like the dark theme. And then there's also the new update controls that we've been talking about where users are able to defer updates for some length of time. But again, Microsoft is still going to keep people moving forward through the update channel if that's the version of Windows you're using. Oh, and Search and Cortana are split, so that they're no longer merged. There are separate appearances of them on the taskbar. And I meant to look at that and forgot when I was playing with it this morning.

So anyway, that's all there. And I imagine, if you don't ask for it, Microsoft will roll it out over time. Oh, and they did solve the problem with external drives being connected. For a while, remember, for the insider and fast ring, they were no longer installing 1903 because there was a problem with drive letters being changed on the fly. That they sort of incrementally fixed and got to a point where they are convinced that it's now working, so 1903 is rolling out again.

So Mozilla just released Firefox 67 with a bunch of welcome enhancements. They have improved its use of memory, meaning using less, and also managed to increase its speed. It has a smarter page renderer that no longer waits to load features that are not necessary to display a web page. So it brings the page up, and then other non-display things that the page has asked for it then is able to catch up with and load in the background. So apparently, I mean, it's like way snappier, they're saying 40 to 80% faster on common pages that people load. So that should improve the experience.

Also, I was just talking maybe a couple podcasts ago about all the memory that tabs consume, for those of us who use lots of tabs. And I was talking about an add-on, Auto Tab Discard, which was sort of smart about releasing memory from tabs that you weren't using. Turns out that's now build into Firefox 67. It will be smart about that. If it sees that you're beginning to consume memory, what they said in their announcement is that, when less than 400MB is available, just to give you some headroom, Firefox will start

suspending and unloading the least recently used tabs. So that's nice. I mean, you can always just click on it, and then it just reloads it.

So it does slow down instant tab switching, if that's the case. But if you've got lots of tabs open and not enough memory to contain them all - and, boy, I'm surprised by how much memory web pages take these days. I mean, I watch it as I open tabs and think, what are these doing? But, you know, that's the nature of the world now. 67 also adds fingerprint and cryptomining blocking features natively, which is kind of cool. You need to go to - in Firefox there's Standard and Strict, and then there's a third, which is Custom. You need to go to Custom Content Blocking, which brings up a dialog that now has two by default unchecked checkbox options, one for cryptominers and one for fingerprinters. So it will introduce some technology to block both of those behaviors.

Oh, the other nice thing is that it used to be that the private browsing mode did not bring with it the extensions which have been installed outside of private browsing. And that of course created some inconvenience for users, for example, who depend upon LastPass in order to log into anything anywhere these days, as I do. We're not supposed to know our own passwords any longer.

So one of the new features in 67 is that, on a per-extension basis, you're able to permit extensions to also appear in private browsing mode. When I upgraded to 67, it defaulted to them all being enabled for private browsing. If you go to the add-ons page, you'll see like a purple tag saying "Allowed in Private Windows." You can selectively turn those off if you don't want those extensions to also be over on the private side. Then moving forward, whenever you install a new extension, Firefox will prompt you for whether or not you also want that to appear in private windows. So you've got complete control over whether or not they are there.

Oh, and their built-in Login Credential Manager for storing and retrieving the browser's native username and password storage will now also be available there. So a lot of the things, I mean, it used to be a very restricted environment which in some cases made it awkward for people to use the private browsing mode, so they've made that much smoother.

Finally, they're getting ready to test what they're calling their WebRender rendering engine. They'll be rolling it out slowly. It will be using the system's GPU for 2D rendering of web pages to further improve rendering speed. It'll be first available for Windows 10 users who have NVIDIA graphics cards, and over the course of the year it's expected to be rolled out further.

I mentioned, at the top of the show and also previously, how annoyed I was, but I could understand, the fact that a newly issued Authenticode certificate from DigiCert was not initially trusted. And thinking about it a little bit, I could understand why that was the case because, as we know, certificates function on a default trust model. That is, by default we trust a big bunch of certificate authorities, and we also by default trust any certificate which they have signed, unless and until we find out otherwise.

So there was an interesting piece in Medium which was discussing a nice piece of analysis done by a group that posted their work on GitHub. I have the Medium link in the show notes. And also, Leo, that second link, the GitHub link, is probably worth scrolling down through, if you're curious. It is a list of malware signed with valid security certificates. And for what it's worth I understand this problem now more than I did. The group of researchers are with Chronicle. They scanned VirusTotal to gain a deeper understanding of the issue.

As Medium explained, for this investigation researchers only included Windows standard Portable Executable, as they're called. All Windows EXEs are called PE, Portable

Executable files. They filtered out samples that had less than 15 detections. So for what it's worth, I don't think mine ever had that many, so it would have never even been a false positive in this research. They aggressively filtered out what they called "grayware" files, and calculated the distinct individual number of samples each signing Certificate Authority was responsible for. So they collected data within the past year, that is, the 365-day span starting on May 7th, 2018, ending on May 8th, 2019.

So what did they find? Okay, this is VirusTotal, malware signed with certificates that have initially been trusted. They found 3,815 distinct separate malware, pieces of malware, that met all of their filtering criteria. So in other words, there is a lot of true malware now being signed by valid certificates which are being issued by trusted certificate authorities. And as I teased at the beginning of the show, suggesting that we can guess who the number one CA was providing certificates that malware is using, that's our old friends, Comodo.

Leo: I almost said the Hong Kong Post Office. But then I knew it's got to be Comodo, yeah.

Steve: It's got to be Comodo. And remember it was a couple months ago that I ran across some malware where the note was that it was signed with a valid certificate, and the certificate was Sectigo. And I thought, huh? Never heard of them. And I said at the time, I did a little bit of digging. Turns out that Comodo renamed themselves Sectigo because, of course, who would want to be Comodo with the reputation that they had acquired? So it turns out that, if you sum the certificates issued by Comodo and Sectigo, because after all they're the same company, they provided the certificates which signed 1,957 of those 3,815 malware samples. That's more than half. 51.3% were all signed by certificates issued by that one certificate authority.

I have in the show notes a graph that you had on the screen a second ago, Leo, which demonstrates just, I mean, just the dramatic skew that exists. Comodo was at 1,775. Then it dropped to less than a third of that, meaning that Comodo is more than three times more than the second greatest number of malware samples signed, and that's a Thawte, you know, T-H-A-W-T-E, at 509. Then at less than, well, okay, almost just half of that is VeriSign. And then Sectigo is also - of course that's the same as, as we know, Comodo - is at 182. And then the fifth is 131 with Symantec.

So as the Chronicle guys note, there's a precipitous drop-off in the numbers showing a decidedly non-uniform skew. And I don't know why. When I looked through that list on GitHub, there were some weird-looking names. I think that what happens is Comodo has resellers, and so they're selling through third parties. And maybe that's the problem is that they're not vetting their third-party resellers well. There was something like Lemon Lime, or Lemonade or something, it was a name that came up in a lot of the certificates that had to have been - in fact, those also had Comodo in their name. But it was like Lemon Time or Lemon - I don't remember what it was. I was looking through it, thinking what the heck is going on here?

So anyway, so there is a certificate revocation process in place. Certificates are being revoked. The problem is that, unlike the real-time communications certificates that we have with websites, for example, when a web server's cert expires, nobody will ever trust its signature who has their clock set correctly. But think about it. You don't want that to be the model for code signing because you could have a repository where a perfectly legitimate signed piece of code has a certificate. After all, the certificates, as we know, only last three years.

So the way this is handled is that, as long as the certificate signing time is timestamped and the certificate was valid at the time of timestamping, then the thing that certificate has signed is considered valid in perpetuity, unless that certificate is specifically revoked. So what happens, for example, when I'm signing things, there's a URL as part of the signing process that makes a real-time query to DigiCert's time server. The hash of what's being signed is sent to DigiCert. They sign it and return it with a timestamp. And all of that is bound into the final signature. So you have an absolutely guaranteed timestamp on the signing.

But the point is that then, even after that certificate expires, the package continues to be valid, which is what you want for validly signed things. You're saying this was legitimate at the time it was signed. And if for some reason you had a need to continue updating the signature, you can certainly just re-sign things with a new Authenticode certificate. But that's not the way things operate in this static-signing model as opposed to the dynamic communications model that we have between web browsers and servers.

So anyway, we have the same problem, it turns out, with revocation of Authenticode and code-signing certificates that we have with web certificates, and that is that certificates are being revoked, and it is the case that, when notified, the certificate authority will check. What these guys that did the research did was that they had to reprocess the malware that VirusTotal already had in order to cause VirusTotal to recheck the certificate which was valid at the time it was initially ingested by VirusTotal, but which may have been revoked subsequently. VirusTotal wasn't automatically going around and doing that.

But by forcedly causing VirusTotal to rescan the malware that it had, they found, I think the number was 21 percent of the instances had certificates that had subsequently been revoked. And of course that's ahead of them expiring, and of course you're able to revoke a certificate even if it's been properly signed because, if any certificate has been used to sign malware, you have to assume that that certificate will be used to sign other pieces of malware, and you want to wipe them all out and cause them to no longer be trusted.

Anyway, so where we are today is we're in a world where, unfortunately, it is very possible for malware one way or another to arrange to get a certificate which is trusted at the time it is signed and will then be trusted by systems that are trying to decide whether or not they want to trust software until they earn a reputation, which is what I've gone through now. Even if I freshly sign something with the certificate which is now more than a month old, it's acquired a reputation. I think there's, like, two off-brand AVs out of more than 70 over on VirusTotal, only two that are saying we're not sure about this. Like DNS8 or something is one of them, and there's some other.

So anyway, it's no longer causing us a problem. Microsoft recognizes it. And I just think this is the world we're in now. I understand now why any new certificate needs to be regarded with some skepticism. The good news is, thanks to certificates, it is possible to earn a reputation. And once you have earned a reputation, which you're able to hold onto for up to three years, then you're able to no longer have a problem with AV, which is, you know, it's not that the code is doing anything malicious. It's just that AV is really suspicious. It has to be.

Leo: Can we talk about the end of the Internet?

Steve: I think we've got to do that. So I chose to discuss this RDP problem further since it's been quite a while since we've had one of these truly perfect, which is to say actual

and serious, Internet-wide threats to observe and discuss in real-time for the podcast. And by all appearances, this one is not going to disappoint.

Okay. So it turns out - I will get to this - the scanning by the bad guys has begun. But let me sort of take us through the last week first. As we know, we have a wormable Windows Remote Desktop Protocol vulnerability which is bad enough that, two weeks ago, Microsoft reached all the way back to Windows XP to offer a patch for those systems. We now also have a name for it that we didn't last week. It's known as BlueKeep, CVE-2019-0708.

So last Thursday, on May 23rd, Dan Goodin for Ars Technica wrote: "It's been nine days since Microsoft patched the high-severity vulnerability known as BlueKeep, and yet the dire advisories about its potential to sow worldwide disruptions keep coming. Until recently, there was little independent corroboration that exploits could spread virally from computer to computer in a way not seen since the WannaCry and NotPetya worms shut down computers worldwide in 2017. Some researchers felt Microsoft has been unusually tight-lipped with partners about this vulnerability, possibly out of concern that any details, despite everyone's best efforts, might hasten the spread of working exploit code.

"Until recently, researchers had to take Microsoft's word that the vulnerability was severe. Then five researchers from" - that's five researchers at one firm, but we'll talk about others - "five researchers from security firm McAfee reported last Tuesday that they were able to exploit the vulnerability and gain remote code execution without any end-user interaction. The post affirmed that CVE-2019-0708, as the vulnerability is indexed" - and of course we know it as BlueKeep now - "is every bit as critical as Microsoft said it was."

McAfee's team wrote: "There is a gray area to responsible disclosure. With our investigation we can confirm that the exploit is working, and that it is possible to remotely execute code on a vulnerable system without authentication."

The next day, last Wednesday, we saw two more posts about BlueKeep. One, from security firm ESET, was succinctly headlined: "Patch now! Why the BlueKeep vulnerability is a big deal." In it, ESET's Security Evangelist wrote: "Right now, it is only a matter of time until someone publishes a working exploit, or a malware author starts selling one on the underground markets. Should that happen, it will probably become very popular among less skilled cybercriminals and also a lucrative asset for its originator."

Security vulnerability researchers at Check Point via Twitter. @Eyal Itkin tweeted: "The last three days were intense, but with help from the @_CPResearch_ team, we now have a" - so this is Check Point. We have ESET; we have McAfee; we have Check Point now. "We now have a working BSOD" - meaning Blue Screen of Death - "proof of concept for CVE-2019-0708. Time to catch some sleep."

Kaspersky security researchers interested in reverse engineering. This is Boris Larin at Kaspersky tweeted: "We analyzed the vulnerability 0708 and can confirm that it is exploitable. We have therefore developed detection strategies for attempts to exploit it and would now like to share those with trusted industry parties. Please contact nomoreworm@kaspersky.com."

Also Chaouki Bekrar, the founder of Zerodium. He tweeted: "We've confirmed exploitability of Windows Pre-Auth RDP bug 0708 patched yesterday by Microsoft. Exploit works remotely, without authentication, and provides system privileges on Windows Server 2008, Window 7, Windows 2003, and XP." He added: "Enabling NLA mitigates the bug." Then he said: "Patch now or GFY." And we'll let our listeners figure out what GFY stands for.

Also Valthek, who is a well-known malware analyst with more than 20 years of experience, he tweeted: "I get the CVE-2019-0708 exploit working with my own programmed PoC," he says in parens, "(a very real dangerous proof of concept)." He says: "This exploit is very dangerous. For this reason, I don't will said to anybody or any enterprise nothing about it. You are free of believe me or not. I don't care."

Anyway, he actually had this confirmed. The senior principal engineer and lead scientist for McAfee said: "After many hours, @ValthekOn was able to get a working proof of concept for this. We're not going to reveal technical details or release code. We urge everyone to patch. It is really nasty." And then a bunch of other people were - that was a tweet from Christiaan Beek at McAfee who included a bunch of other researchers in his tweeting so that they would see this also.

So I'm trying to see if there's anything else here. McAfee did one of the two most complete breakdowns. And so there's additional information. And I'm going to sort of - I want to, without getting into the weeds because there's no point here, I'm going to sort of explain what's going on to give everybody a sense for it. So I'm going to share the description of the problem. It uses a bunch of terms and presumptions that we have not defined, and there's really no point. But you'll get a sense for it.

So McAfee wrote: "The Remote Desktop Protocol (RDP) enables connections between a client and an endpoint" - that is, you know, the server - "defining the data communicated between them in channels." That is, that's the way the protocol works. It has virtual channels. "The virtual channels are bidirectional data pipes which enable the extension of RDP. Windows Server 2000" - that is way back then, Windows Server 2000 - "defined 32 Static Virtual Channels (SVCs) with the release of RDP 5.1, but due to limitations on the number of channels" - which is to say they probably, what, so 32 means six bits, so they probably only allocated, they only allowed six, no, I'm sorry, five bits for 32. They allowed five bits in the protocol somewhere, so that wasn't enough.

"So due to limitations of the number of channels, they further defined what was known as Dynamic Virtual Channels (DVCs), which are contained within a single dedicated SVC, a static virtual channel. SVCs are created at the start of a session and remain until session termination, unlike DVCs, which are created and torn down on demand. It's this 32 SVC binding which CVE-2019-0708 patch fixes with the - and then there's two functions, IcaBindVirtualChannels and IcaRebindVirtualChannels functions in the RDP driver termdd.sys. As can been seen in Figure 1" - and this is from McAfee's disclosure - "the RDP Connection Sequence connections are initiated and channels set up prior to Security Commencement, which enables CVE-2019-0708 to be wormable since it can self-propagate over the network once it discovers open port 3389." In other words...

Leo: Now, just to be clear, 3389 has to be open, and it has to be on a machine that supports RDP.

Steve: Correct.

Leo: So Windows 7 Home or XP Home doesn't have RDP support.

Steve: Correct. Oh, correct. And this is - it's worth mentioning. This is not a concern for the typical Home end user.

Leo: Because you don't turn it on; right?

Steve: Well, not only is it not on, but you're also behind a NAT router. And so you're not going to have that exposed. And anybody running Windows 10 is also okay because this only affects 7 and XP. And not even Windows 8, for anyone who still has that. So, yes. So the concern here is the publicly exposed instances on the Internet, and that's where we're getting to.

Leo: Or inter-LAN communication.

Steve: That's true, too, yes.

Leo: And that was the big problem with WannaCry because of EternalBlue, which allows you to use SMB-1 to worm across the LAN. So if you have a bunch of RDP ports open on your local LAN, that's as bad; right?

Steve: That is true. Yes, yes, yes. So McAfee explains, without us defining these terms: "The vulnerability is due to the MS_T120 SVC name" - that's the Static Virtual Channel, the MS_T120 SVC, the Static Virtual Channel name - "being bound as a reference channel to the number 31 during the GCC Conference Initialization sequence of the RDP protocol." Again, that's all gobbledy-gook, but just sort of - you sort of hear that. "This MS_T120 Static Virtual Channel name being bound as a reference channel to the number 31 during the GCC Conference Initialization sequence of the RDP protocol. This channel name is used internally by Microsoft, and there are no apparent legitimate use cases for a client to request connection over an SVC [Static Virtual Channel] named MS_T120.

"However, during GCC Conference Initialization, the client supplies the channel name which is" - remember, that's the client meaning something outside, or soon malware - "supplies the channel name which is not whitelisted by the server" - which is to say, which does not need to be whitelisted is what McAfee really meant to say, which does not need to be whitelisted - "meaning an attacker can set up another Static Virtual Channel named MS_T120 on a channel other than 31. It's the use of MS_T120 on a channel other than 31 that leads to a heap memory corruption and remote code execution."

So essentially this was - there was some feature that Microsoft has always had in the protocol which normal clients don't use, but which is in the code. And if it's used on channel 31, the Static Virtual Channel 31, no problem. That's the only way Microsoft ever would use it for whatever purpose they might have. But it turned out someone discovered, well, and I don't remember now the genesis of this, whether - I don't think we ever heard who found it. You know, nobody else came out with a public disclosure.

So this may have been privately reported. Maybe Microsoft found it themselves. But one way or the other, if somebody hooks up to an RDP protocol server, from XP through Windows 7, and asks for this MS_T120 to be bound to a Static Virtual Channel other than 31, the server crashes in a way that basically everybody who has looked at it, we went through all of those tweets and all the companies, that is to say, this is not hard to do, which is the reason everyone's running around with their hair on fire is that everyone who's looked at it has been able to make this happen. And now here we have public disclosure of exactly what's going on, how to do this yourself.

So McAfee says: "The MS_T120 reference channel is created in the rdpwsx.dll and the heap pool allocated in rdpwp.sys. The heap corruption happens in termdd.sys when the MS_T120 reference channel is processed within the context of a channel index other than 31. The Microsoft patch adds a check for a client connection request using channel name

MS_T120 and ensures it binds to channel 31 only [in those two functions] IcaBindVirtualChannels and IcaRebindVirtualChannels, within termdd.sys."

Okay. So translating this, as I said, there was some flexibility. What happened was Microsoft fixed this by enforcing the use of MS_T120 only for channel 31. It turns out that there's another reverse engineering elsewhere on the Internet that is also public, where this is all shown. So this was a very simple thing to fix. And in fact there is a 30-byte patch involving 22 instructions from the 0patch guys, you know, those guys who make the micropatches? Such that you don't even need to reboot the server to fix this. You can do a non-reboot, in-memory patch to close this problem.

And one of the things I want to address here in a second is the problem that we have with our laws at the moment, and the fact that Microsoft can't do this themselves because we probably really are facing an event on the Internet, and it could be avoided technically, but it cannot be avoided legally.

Anyway, McAfee says: "After we investigated the patch being applied for both Windows 2003 and XP and understood how the RDP protocol was parsed before and after the patch, we decided to test and create a proof of concept that would use the vulnerability to remotely execute code on a victim's machine to launch the calculator application, a standard litmus test for remote code execution." And then they reiterated, saying: "There is a gray area to responsible disclosure." Meaning that that's - they don't want to go any further.

They said: "With our investigation we can confirm that the exploit is working and that it is possible to remotely execute code on a vulnerable system without authentication." And then they confirm that: "Network Level Authentication should be effective to stop this exploit if enabled; however, if an attacker has credentials, they will bypass this step."

They said: "As a patch is available, we decided not to provide earlier in-depth detail about the exploit or publicly release a proof of concept." Not that it matters because, again, this is just not that hard to do. "That would, in our opinion, not be responsible and may further the interests of malicious adversaries." And they said: "It is important to note as well that the RDP default port can be changed in the registry and after a reboot will be tied to the newly specified port. From a detection standpoint, this is highly relevant."

Okay. So I guess they're saying there may also be not yet discovered RDP servers that are running on non-default ports that nobody has found yet. So of course I want to make sure people don't think that using a non-default port is meant as a security solution. It certainly isn't. We know how to fix this problem, and that is to use a VPN only with strong security to allow access to the Remote Desktop Protocol. As I said last week when we first talked about this, that's the way to do this is run OpenVPN on a machine and use its very strong authentication with certificates in order to enable access to the server.

So as of today we have Zerodium, McAfee, Kaspersky, Check Point, MalwareTech, and Valthek all having confirmed that they've successfully developed exploits for BlueKeep. None of them is publishing, but there have been cross-verifications of each other's work. So this strongly suggests that the vulnerability is decidedly not difficult to weaponize.

A security researcher, Sean Dillon at RiskSense, has created a tool to allow companies to test whether their own PC fleets have been correctly patched against the BlueKeep flaw. So Leo, just as you were talking about within an Intranet, there are now, and even since I wrote this, I have found some others. RiskSense is packaged as a docker. I've got the link to the GitHub docker in the show notes. It has been turned into a Metasploit module, so Rapid7 now has it in Metasploit.

And Robert Graham, who tweets as ErrataRob, and that's where his blog is, he trimmed down the RiskSense work, produced a nicer, leaner, and higher speed "C" scanner. I've also got - he's posted his on his GitHub page as rdpscan. So I first have his initial take, and then I have an update from this morning. So Rob first wrote a couple days ago of rdpscan for this vulnerability, BlueKeep vuln. He said: "This is a quick-and-dirty scanner for the 0708 vulnerability in Microsoft Remote Desktop. Right now" - okay, now, this is no longer current. I will update this in a second.

He said: "Right now there are about 700,000 machines on the public Internet vulnerable to this vulnerability, compared to about two million machines that have Remote Desktop exposed, but are patched or safe from exploitation. Many expect that in the next few months a devastating Internet worm will appear similar to WannaCry and NotPetya. Therefore, scan your networks and patch your systems. This tool makes it easy to scan your networks to find vulnerabilities. To use this tool, you can download a binary to run from the command line, or you can download the source and compile it. For Windows, there's a precompiled binary available."

He writes: "The tool is based entirely on the desktop patch from," and then he cites the code from RiskSense. He says: "I've simply trimmed the code so that I can easily compile on macOS and Windows, as well as added the ability to scan multiple targets." He says: "This is only a couple days old and experimental. However, I am testing it by scanning the entire Internet with the help of masscan" - which is his parallel scanner - "so I'm working through a lot of problems pretty quickly. You can try contacting me on Twitter (@erratarob) for help and comments."

So that was yesterday, so dated 5/27. He says: "You Linux peeps get only source as usual. It seems to be working well on all three platforms." This morning he updates. Oh, well, first GreyNoise Intelligence over the weekend said BlueKeep scans started. They tweeted, @GreyNoiseIO: "GreyNoise is observing sweeping tests for systems vulnerable to the RDP BlueKeep." Now, of course some of these could be, for example, ErrataRob doing a scan. So certainly we know that we're going to have some white hat researchers scanning in order to help us appreciate the size of this threat. It's clear also, well, actually in this case we know that it's not Rob. I'll explain why. They said: "...vulnerable to the RDP BlueKeep vulnerability from several dozen hosts around the Internet. This activity has been observed from exclusively Tor exit nodes and is likely being executed by a single actor."

ZDNet reports: "Intense scanning activity detected for BlueKeep RDP flaw. A threat actor hiding behind Tor nodes is scanning for Windows systems vulnerable to BlueKeep flaw." And they wrote: "While the infosec community was holding its collective breath, thinking attacks may never start, things changed over the weekend. On Saturday, threat intelligence firm GreyNoise started detecting scans for Windows systems vulnerable to BlueKeep. Speaking to ZDNet, GreyNoise founder Andrew Morris said they believe the attacker was using the Metasploit module developed by RiskSense to scan the Internet for BlueKeep vulnerable hosts. So far we've observed only scans, no exploitation attempts. But it appears that at least one threat actor is investing time and effort compiling a list of vulnerable devices, presumably in preparation for the actual attacks."

And of course, as I've noted, at least six research groups have announced that they've come up with private BlueKeep exploits, and there have been two detailed write-ups, one from McAfee that I shared, and there's also one from WazeHell, W-A-Z-E-H-E-L-L dot I-O. So I think it's only a matter of time before we see more. Now, this morning, Robert Graham posted to his own Errata Security blog the title: "Almost One Million Vulnerable to BlueKeep Vuln."

He starts his posting saying: "Microsoft announced a vulnerability in its Remote Desktop product that can lead to robust, wormable exploits. I scanned the Internet to assess the

danger. I find," he writes, "nearly one million devices" - and actually it was 923,000 is his count, although some successive rescans found some additional ones, so it looks like it's approximately 950,000, thus nearly - "one million devices on the public Internet that are actually vulnerable to the bug." And I'll break this down in a second.

He says: "That means, when the worm hits, it'll likely compromise those million devices. This will likely lead to an event as damaging as WannaCry and NotPetya from 2017, potentially worse, as hackers have since honed their skills exploiting these things for ransomware and other nastiness."

He said: "To scan the Internet, I started with masscan, my Internet-scale port scanner, looking for port 3389, the one used by Remote Desktop." And of course we know that's by default. There may be other RDP servers on different ports. He says: "This takes a couple hours and lists all the devices running Remote Desktop, in theory. This scan" - so just the scan for TCP connection acceptance on 3389 - "returned 7,629,102 results. However," he writes, "there is a lot of junk out there that'll respond on this port. Only about half are actually Remote Desktop. Masscan only finds the open ports, but is not complex enough to check for the vulnerability.

"Remote Desktop," he writes, "is a complicated protocol. A project was posted that could connect to an address and test it to see if it was patched or vulnerable. I took that project and optimized it a bit, called," he says, "rdpscan, then used it to scan the results from masscan. It's a thousand times slower, but it's only scanning the results from masscan instead of the entire Internet. The table of results is as follows." And I have them in the show notes for anyone who's interested.

So we have 1.447 million that timed out, meaning nothing happened. 1.414 million he says are safe. The target appears to have been patched. 1.294 million unknown. The connection was reset. So again, probably just a closed port. 1.235 million, those are safe. It looks like credentials are required. Remember, these are all IPs that did respond to 3389. So then we hit 922,671 are vulnerable, successfully verified that RDP is there and not patched. 651,000 received a FIN, meaning a closed port. 438,000 connection timeout. And I won't bother enumerating. They fall off from there.

He says: "The various unknown things fail for various reasons. A lot of them are because the protocol isn't actually Remote Desktop and respond weirdly when we try to talk Remote Desktop to them. A lot of others are Windows machines, sometimes vulnerable, sometimes not, but for some reason return errors sometimes. The important results are those marked vulnerable. There are 923,671 vulnerable machines in this result. That means we've confirmed the vulnerability really does exist, though it's possible a small number of these are honeypots deliberately pretending to be vulnerable in order to monitor hacker activity on the Internet.

"The next result are those marked safe due to probably being patched. Actually, it doesn't necessarily mean they are patched Windows boxes. They could instead be non-Windows systems that appear the same as patched Windows boxes. But either way, they're safe from this vulnerability. There are 1.414 million of them. The next result" - anyway, he talks about the safe ones and goes down. Then he talks about: "But since a lot of those unknowns could be due to transient network errors, then in theory I should be able to rescan them and get some more positive results. I did this," he says, and here's what he found.

So of those that were unknown, 28,000 were safe, saying that the target appeared patched. He found another 20,000 new vulnerable ones, and a third scan found an additional 6,000 vulnerable ones. So he says: "The upshot is that these tests confirm that roughly 950,000 machines are on the public Internet, right now today, that are vulnerable to this bug. Hackers," he writes, "are likely to figure out a robust exploit in the

next month or two and cause havoc" - a month or two? A week or two - "and cause havoc with these machines." He says: "There are two things you should do to guard yourself."

Okay, and I'm not even going to bother with that. Of course the problem is these are non-maintained machines. I mean, they would not be vulnerable today if they were being maintained. If anyone knew that they had a machine out on the public Internet that was listening to any security alerts, any podcasts, any people tweeting security dire warnings, they would have fixed this.

So here's the problem. We have a real problem that needs to be fixed right now. Code Red was not the last worm. Nimda was not the last worm. MSBlast was not the last worm. WannaCry was not the last worm. NotPetya was not the last worm. And whatever this worm, I mean, this will be, what, maybe the Blue worm. Who knows what we're going to call it. It won't be the last one, either.

What's upsetting is that this can be patched in RAM. If it were legal for Microsoft to do this, they could, before ever having gone public - I'm sure they know how many machines are there. They were probably looking at them themselves before they decided how to handle this. They could reach in there and close this problem and then back out so that, even without rebooting these systems - because this patch that's been developed does not require a reboot. Even without rebooting them, they could close this hole which is there and keep what is absolutely, definitely certain to happen. I mean, there's just no way this is not going to happen with all the attention this is getting. They could keep it from happening, but they cannot do that legally.

And so it seems to me that we need some sort of a worldwide consensus that allows white hats with proper oversight, for example, a body composed of individuals from the security industry and appropriate government leaders, to whom Microsoft could present in private, in secret, under obvious nondisclosure because these would all be white hats, could explain the problem and get an agreement that they're going to fix this problem before it wreaks havoc on the Internet. But today it's not legal for Microsoft to do this.

So it seems to me, since this is not the last time this is going to happen, and none of the last five worms that have been really problematical have been the last worm, we have to assume this is not the last worm. We need to be proactive and somehow get some global treaties with the other nations that we care about, or maybe just do it unilaterally with the U.S.; say look, folks, if software from a supplier that we care about in the U.S. or a friend of the U.S., an ally, if the problem can be proactively fixed in a way that can be shown not to hurt somebody who is unaware of the problem, then it needs to be made legal to reach out and do that because otherwise we're going to have happen what is almost certain that we'll be talking about on next week's podcast, or the week after.

Leo: Well, we know we're going to be talking about malware, ransomware spread by this. I mean, you know. And I don't think it's an accident, that its name is an accident since EternalBlue is the NSA exploit that helps people worm their ransomware from machine to machine. So what is it, Solid Blue? What do they call it?

Steve: It's BlueKeep.

Leo: BlueKeep. This makes a lot of sense. It's keeping Blue active. And ransomware is just getting worse and worse. So it's not the end of the Internet, though. Let's be fair.

Steve: I know. I was just kidding.

Leo: But it isn't good for a lot of businesses. Man, this ransomware seems out of control. Steve, this show will appear in a variety of places.

Steve: Right, hopefully.

Leo: If the Internet doesn't disappear between now and tonight, Steve will post it on his website, GRC.com. He has audio and transcripts there. He also has SpinRite, the world's finest hard drive recovery and maintenance utility, dare I say really the only hard drive recovery and maintenance utility, the only one worth a darn.

Steve: Yeah.

Leo: And you can get that there, his bread and butter, but lots of other stuff including the latest on SQRL, Vitamin D, ShieldsUP!, Don't Shoot the Messenger, DCOMbobulator, I could go on and on. It's basically a rathole of fun. Jump right in.

Steve: That's right.

Leo: Go all the way down. If you want to go to our site, which is much less interesting, we do have at least this show at TWiT.tv/sn. You can watch us live. We do it live Tuesdays, 1:30 Pacific, 4:30 Eastern, 20:30 UTC at live.twit.tv. Audio and video streams are there. Or you can ask your Echo: "Echo, listen to TWiT Live." I think if you say "TWiT Live" it should be enough, but sometimes you have to say "on TuneIn." You can also listen to the most recent version of Security Now! by just saying "Echo, listen to Security Now! podcast," and it'll play it. So it's a nice way to keep up. Or subscribe in your favorite podcast application. That way you'll have it wherever you are, whenever you want it.

Steve: You know, I forgot to mention, I gave the SQRL presentation last Wednesday...

Leo: How did that go?

Steve: ...to the Los Angeles Chapter of OWASP. These rehearsals are being useful. Lorrie, who has sat through both, said it was unbelievably better than the first one, which [crosstalk].

Leo: I like her. Good.

Steve: She said it was unbelievably better. And one of the things that I did was, and I said I was going to at the beginning, I untethered myself from my PowerPoint presentation.

Leo: Yes, good.

Steve: Because I'm just - I'm never in sync with it. I'm running off on different directions and things. So what I've done is, what I realize what I want is, and I'll have this for the next one I do, whenever that's going to be - we're trying to negotiate the right time for the Orange County OWASP Chapter. That'll happen. Also the guys in Dublin confirmed that they still want me, and they're going to fly Lorrie and me out there to present to them.

Leo: Nice.

Steve: So what I'm going to do is, the only thing I really need, because I know this thing so upside down, backwards and forwards, that I don't need prompting by bullet points. What I need is diagrams because pictures of different little algorithms and things, those are the things that it's fun to be able to refer to. And it turns out that with Office 2016 there was a new feature added where you can do a slide of thumbnails, and then I'll be able to tap one, and that'll zoom in. So I can randomly access the images of the various things, sort of like as I'm going, in any sequence that happens. So anyway, I'm rehearsing all of this so that when we do the presentation in your TWiT studio, Leo, it'll just be slick.

Leo: Nice. I can't wait. And yeah, I completely agree. I abandoned PowerPoint slides a long time ago because, well, for one thing, people just read those instead of listening to you.

Steve: Right.

Leo: The way we are, if there's a slide on the screen, we're going to look at that and not listen to what you're saying. So I think you're right. Diagrams yes, presentation no. I think you're exactly right.

Steve: Yup.

Leo: I can't wait to see it. Did anybody record it?

Steve: No, it was not recorded. I've had a bunch of people asking if it had been. Really, I just know what a quality production your guys will produce, and I'm sort of - I guess I'm sort of biased toward that.

Leo: Thank you. Good. We'll give you a nice video the minute you want it.

Steve: We'll take it one step at a time, yeah.

Leo: Good, good. All right, my friend.

Steve: Okay, well, assuming that we're still here next week and that Skype is still working. We'll see.

Leo: I downloaded and ran that scanner. It's a command line utility. I didn't find any open RDP ports on our network.

Steve: Nice.

Leo: Yeah, nice.

Steve: Whew.

Leo: I'm not surprised.

Steve: Okay, buddy.

Leo: See you next time.

Steve: Talk to you next week. Bye.