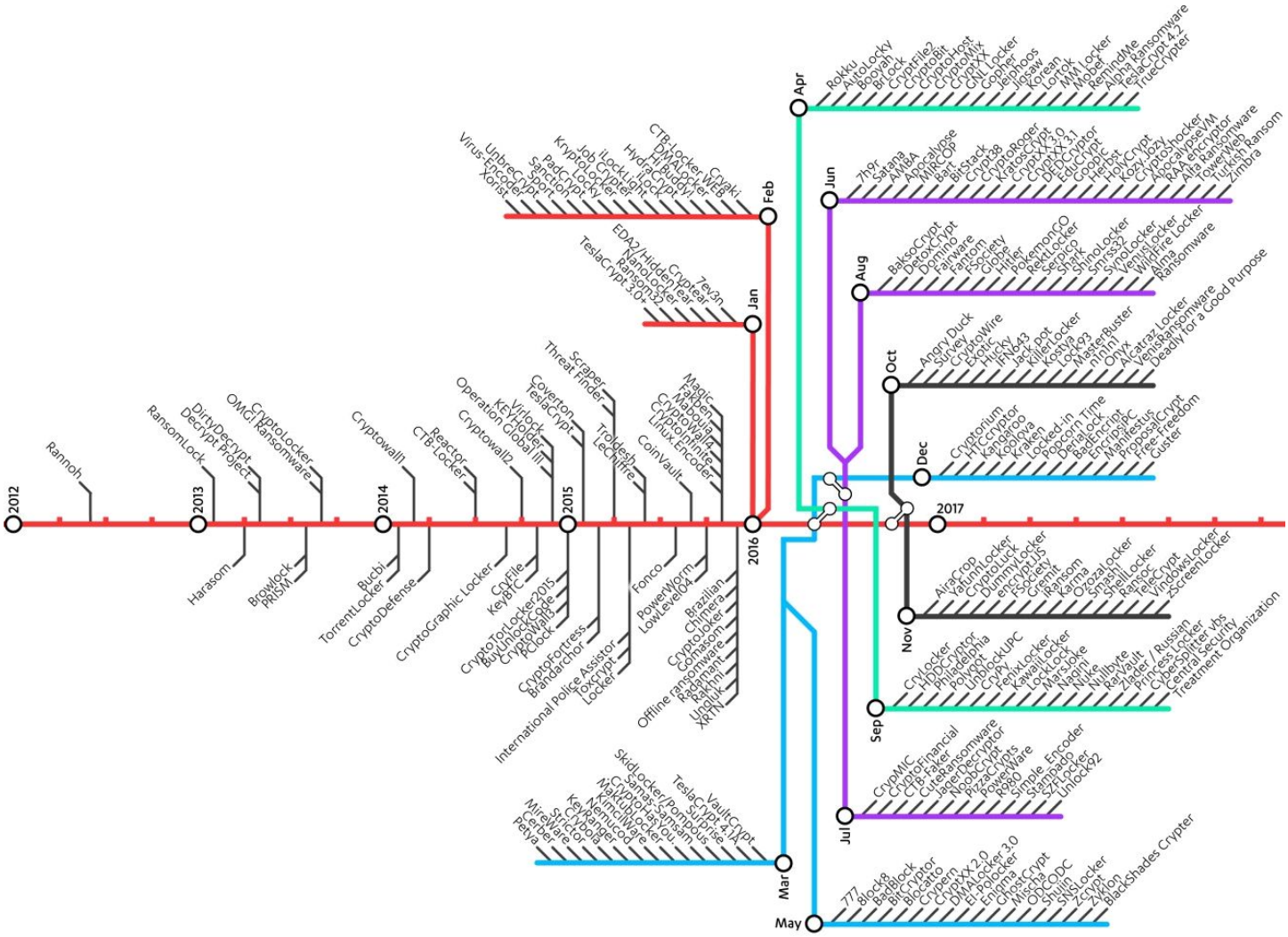
Security Now! #716 - 05-28-19

RDP: Really Do Patch

This week on Security Now!

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 0day 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, some interesting new data about the prevalence of validly sign malware.

The Ransomware Timeline

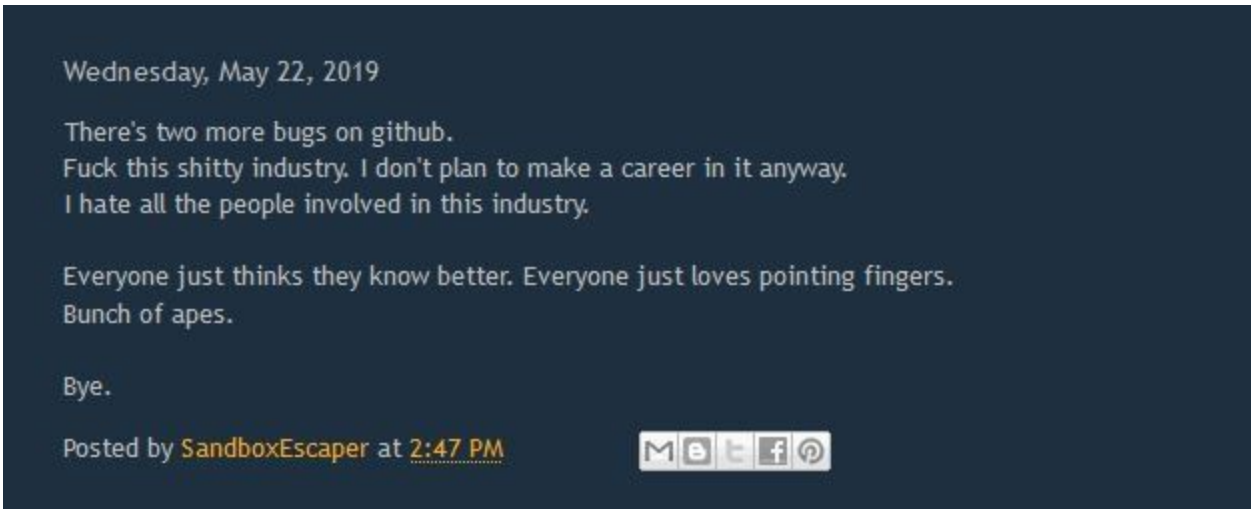


Security News

"@SandboxEscaper" is back... With a vengeance!

Recall that she apparently loves hiking, camping and photography. And that she has that very nifty one-person micro-tent thingy. She has a blog that has since been closed to the public and is by invitation only. She is an obviously-gifted but also apparently unhappy and seemingly troubled hacker. We know of her from her several previous 0-day exploits which we've covered and in some cases examined rather closely. I have come away being VERY impressed by the quality of her work.

Last Wednesday, May 22nd, during an apparently low moment, she posted the following:



In just a few short days, what we have received from this gifted hacker are FIVE brand new 0-day exploits against Windows 10. One turns out to have already been fixed, but that still leaves us with FOUR that are unpatched.

The security industry has been whipped into something of a froth by her release of these, the most recent batch were first promised and then delivered on schedule. In looking around for the best way to share the story, I landed on Davey Winder's thoughtful reporting for Forbes.

We have two pieces from him, the first last Tuesday and then a follow-up the next day, last Wednesday. So, first, last Tuesday Davey wrote:"New Windows 10 Security Exploit Can Read All Your Files -- What You Need To Know"

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?

A security researcher going by the name of 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. 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 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 that Task Scheduler handles discretionary access control list (DACL) rights for such files without DACL permissions; giving full control to any user rather than just the system admin.

[An advisory published Wednesday by US Cert confirmed that the exploit worked against both 32-bit and 64-bit versions of Windows 10.]

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 MSFT (they are actually a cool company). 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 f*cktards in the west the middle finger. I'm donating all my work to enemies of the U.S." 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.

Is there worse to come?

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." Ian Thornton-Trump, head of security at AmTrust International, told me during a conversation this morning that as far as the economics of selling exploits are concerned it's "kind of a sh*thead move." You can understand why as Microsoft is known for having a pretty generous bug bounty program which enables researchers to cash in on their findings without taking the criminal route to riches. "It's sad that folks burn the opportunity to contribute to the information security community," Thornton-Trump said.

What can you do to mitigate the risk?

Given that it is unlikely, based on responses to the previous exploits released by SandboxEscaper, that we will see any patch to fix this zero-day until the next Patch Tuesday on June 11, what can you do to mitigate the risk? "I will tell you that anything that interacts with the task scheduler is going to be pretty unsubtle and fairly easy to detect," Thornton-Trump advises, "probably even by Windows Defender." "Of course, that doesn't mean it will be an impotent threat and zero-day attacks must always be considered a very real and present danger to data. That said, Thornton-Trump isn't panicking over this as most enterprise endpoints have many compensating security controls deployed and those should provide adequate protection."

Home users are advised to ensure their security software is up to date and take care to prevent attackers from gaining access to their systems in the first place...

=====
Okay. That was Davey's posting Tuesday. The following day he posted:

"Rogue Security Researcher With Grudge Against FBI Goes On Windows 10 Exploit Spree"

[I cut out recap from the previous day.]

Things have just become a little more problematical with SandboxEscaper releasing on May 22 another two of the four remaining zero-days she claims to be in possession of. The first is similar to the local privilege escalation (LPE) exploit released on May 21, but this time exploiting a vulnerability in the Windows error reporting service. It is 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 Internet Explorer 11, specifically allowing for the injection of malicious code. Again, 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.

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'll interject a comment that I disagree here. We have recently seen several examples where a local privilege escalation vulnerability has been combined with one or more different exploits 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, however, the exploits allowed hackers to remotely execute malware of their choice. So the pair of privilege- escalation vulnerabilities SandboxEscaper published over the past 24 hours are likely to have similar capabilities when combined with the right additional exploit. We rely heavily upon our operating system's enforcement of execution and access privileges. So anything that can bypass those is no small matter.]

[Anyway... Davey continues:] That said, there are still two more exploits in the SandboxEscaper arsenal and we will have to wait and see what they bring when, and suggest it isn't going to be if, they are released. Given the events so far this week I suspect we won't have long to wait.

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 or if sold via an exploit broker. There are clues in the SandboxEscaper blog as to the real reasoning, and they

are not subtle either: the motivation would seem to be getting back at the U.S. for a perceived injustice. The most telling is the confession that she has "most definitely given portions of my work to people who hate the U.S." because "that's what happens when the FBI subpoenas my google acc 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." 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. "F*ck this shitty industry. I don't plan to make a career in it anyway," SandboxEscaper writes, "I hate all the people involved in this industry."

And an UPDATE, the next day, Friday May 24:

SandboxEscaper has now confirmed that the "windows error reporting bug was apparently patched this month" and so that's one less to worry about. Unfortunately, she has also now released two more zero-day exploits: CVE-2019-0841-BYPASS which, as the name suggests, is a workaround exploit for an elevation of privilege vulnerability that was patched in the May Windows updates, and InstallerBypass which is another LPE vulnerability, but problematical to execute. So likely not going to have a high risk impact. This now makes a total of nine exploits, eight of which are zero-days, released across the last ten months by SandboxEscaper. 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. 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, despite what she has done, that she can get some help with all this and find some inner peace...

[I noted that her Github account has been taken down and that she has closed her blog to the public. Access is now by invitation only.]

Google Stored G Suite Users' Passwords in Plain-Text for 14 Years
Following in the footsteps of Facebook and Twitter, Google becomes the latest technology giant to confess that it had accidentally stored G Suite enterprise user passwords unprotected in plaintext on its (encrypted) servers.

https://cloud.google.com/blog/products/g-suite/notifying-administrators-about-unhashed-password-storage

In a blog post published Tuesday, titled: "Notifying administrators about unhashed password storage" Google's Suzanne Frey, VP, Engineering, for Cloud Trust wrote:

Google’s policy is to store your passwords with cryptographic hashes that mask those passwords to ensure their security. 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 we are 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 access to or misuse of the affected G Suite credentials.

[Suzanne described the standard process of password hashing, which we'll skip here.]

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 upload or manually set user passwords for their company's users. The intent was to help them with onboarding new users; e.g., a new employee could receive their account information on their first day of work, and for account recovery. The functionality to recover passwords this way no longer exists.

We made an error when implementing this functionality back in 2005: The admin console stored a copy of the unhashed password. This practice did not live up to our standards. To be clear, these passwords remained in our secure encrypted infrastructure. This issue has been fixed and we have seen no evidence of improper access to or misuse of the affected passwords.

In addition, as we were troubleshooting new G Suite customer sign-up flows, we discovered that starting in January 2019 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 this is an isolated incident.

We recently notified G Suite administrators to change those impacted passwords. Out of an abundance of caution, we will reset accounts that have not done so themselves. Our authentication systems operate with many layers of defense beyond the password, and we deploy numerous automatic systems that block malicious sign-in attempts even when the attacker knows the password. In addition, we provide G Suite administrators with numerous 2-step verification (2SV) options, including Security Keys, which Google relies upon for its own employee accounts.

We take the security of our enterprise customers extremely seriously, and pride ourselves in advancing the industry's best practices for account security. Here we did not live up to our own standards, nor those of our customers. We apologize to our users and will do better.

=====

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.

Let's hope for our whole industry that these are legacy behavioral events which we won't be seeing in the future. We cannot have code that mistakenly stores passwords in plaintext if such code is never created in the first place. With time we will all learn how to be better at keeping important secrets. Or, better yet, we'll all eventually be using a system like SQRL which gives websites no secrets to keep in the first place. Then it doesn't matter what happens at the other end. :)

Microsoft Releases Windows 10 Version 1903 - May 2019 Update
Microsoft has officially started the roll out to everyone of Windows 10 version 1903, the May 2019 Feature Update.

Goto to Settings -> Update & Security -> Windows Update and check for new updates.

As we've mentioned before, Microsoft's policy about this has recently changed. If the update is available to you, for the time being it will NOT be installed automatically. Instead, it will be offered as an available "Feature update to Windows 10, version 1903" that you can choose to "Download and install now".

Although this feature update is generally available, Microsoft is throttling its availability, so it may not be listed when you check for updates. If you don't see the update, should again from time to time. And, as part of Microsoft's new "Windows Update experience", once you install the Windows 10 May 2019 Update, you will be asked to schedule a time when you wish to install it.

You can get 1903 immediately if you wish, but those who have been holding off and are still running 1803 from April of last year -- or anything even older -- will have 1903 forced down their throat starting next month, in June, like it or not.

In Microsoft's words: For Windows 10 devices that are at, or within several months of reaching, end of service, Windows Update will automatically initiate a feature update; keeping those devices supported and receiving the monthly updates that are critical to device security and ecosystem health. The Windows 10 April 2018 Update (Windows 10, version 1803) will reach end of service on November 12, 2019 for Home and Pro editions. Starting this June, we will begin updating devices running the April 2018 Update, and earlier versions of Windows 10, to ensure we can continue to service these devices and provide the latest updates, security updates and improvements. We are starting this machine learning (ML)-based rollout process several months in advance of the end of service date to provide adequate time for a smooth update process.

So what does this month's Windows 10 Feature Update bring us?

=*Windows Sandbox*=

This update adds a lightweight virtualization feature called the Windows Sandbox which allows its users to run Win10 in a fully-isolated virtual machine. This will allow the testing of suspicious programs, web extensions, or web sites for malware or malicious purposes without having fear that it could affect anything outside of the sandbox.

Microsoft describes Windows Sandbox as a lightweight desktop environment smaller than 100mb. The Sandbox is listed as an application on the Start menu. However, isn't necessarily supported on all PCs, and it is disabled by default. This makes sense since it is a rather special-purpose tool. Listeners of this podcast will love it... but it's definitely aimed at the higher-end power user.

For the Sandbox to be enabled, you'll need to have:

- Windows 10 Pro or above.
- Requires a 64-bit processor with at least two cores, but quad-core CPU device with multi-threading is recommended.
- Virtualization enabled in the BIOS.
- At least 4GB of RAM required, though 8GB is recommended.
- 1GB of disk space, preferably SSD.

If it's available it will appear in the "Windows Features" listing which you get to from the "Turn Windows features on or off control panel." So turn it on and then look in the Start Menu once Windows indicates that it's been installed.

When it's started, it will look like a new installation of Windows 10 running within a window like an application. You can open, download and run any files or programs in Sandbox and no changes will be made to the actual system. When it's closed, any changes you made will be permanently deleted. This will be very familiar to user's of VMs... but it's becoMing part of Windows 10.

However, for those of us who are accustomed to working with full VMs, the Win10 sandbox has some limitations. For one thing, enabling the sandbox also enables Hyper-V on the machine. After that, VMWare and VirtualBox will not run on the computer until the sandbox is uninstalled. Microsoft must have assumed that someone who was using a full VM would not be a candidate for Windows Sandbox, but keeping this incompatibility in mind is important.

The main benefit of the the Windows Sandbox is its extreme ease of use. It allows users of pretty much any skill level to test programs downloaded from the web. I do often want to take a look at something before committing to its installation.

However, the Windows Sandbox only consists of the base OS without additional applications. So Word or Excel content cannot be used without first installing Office into the sandbox. And since the VM is reset to its original state every time it is started, there’s no way to persist any applications.

Overall, it’s cool… but I’m unsure exactly who it’s aimed at. A user who is sophisticated enough to want a VM probably wants a full VM. But it’s built in and it’s a nice blank slate every time it’s launched. Its conflict with VMware would prevent me from using it on a base OS workstation where I always have VMware. But it DOES run inside a Win10 VMware VM… so perhaps I’ll find some use for it.

="Light" Theme=

Windows is also getting a true "Light Theme." To enable the light theme and lighten your world Open Settings, navigate to Personalization > Colors. Select the light option under 'Choose your color'.

=More Update Control=

And as we've already noted in previously, Windows Update is acquiring a few new options:

User will have more control over how and when updates are installed with the addition of a pause button below the 'Check for updates' trigger. This will allow updates to be delayed for up to seven days. Also, once a new (monthly) cumulative update has been downloaded, a new icon will appear on your taskbar as a reminder that the system requires a reboot. There will also be a new prompt when Feature Updates are available. Rather than installing immediately, from now on Win10 will prompt to "Download and install now" when you are ready to install it. Microsoft is also claiming to be using "AI" to be super-smart about when we're least likely to be inconvenienced by a forced system restart... But I think I'd prefer to mange that myself. Fortunately, that new smart feature is currently disabled by default. It's under Settings > Update & Security > Change active hours.

=Separating Search from Cortana=

This May 2019 Update also splits Windows Search and Cortana apart so that they are no longer part of the same user interface element but each has their own place on the taskbar. There are two icons, Search and Cortana. And we're getting some more control over Search to speed it up and also to index all local drive content not only our document libraries. Look under: Settings > Search > Searching Windows.

And speaking of 1903…
Last month we noted that Windows 1903 was suddenly having so much trouble with drive letter reassignments that Microsoft was suspending its early availability if the system being updated had any external USB drives or SD cards attached.

With the cumulative update KB4497935, which Microsoft has released for Windows 10, 1903 to Insiders in both the Slow and Release rings. Among the things it resolves is:

"Addresses an issue that may cause an external USB device or SD memory card to be reassigned to an incorrect drive during installation. For more information, see “This PC can’t be upgraded to Windows 10” error on a computer that has a USB device or SD card attached."

This was certainly a prerequisite for the full release of 1903, which is now upon us.

How to "force get" the Windows 10 1903 update?
I've become a fan of the "Candy Crush Saga-free" LTSC edition of Windows 10. It makes Win10 acceptable. But, of course, the whole point of those “Long Term Servicing Channel” releases is that they are NOT being continually updated and forced forward like the regular consumer Win10’s.

Not ONE of my non-LTSC Win10 machines or VM’s would voluntarily agree to auto-update. And since I figured many of our listeners might be in the same situation, I did a bit of poking around and found a little 5.9 megabyte executable file from Microsoft that will allow immediate user initiation of the upgrade to 1903:

- https://go.microsoft.com/fwlink/?LinkID=799445
- http://bit.ly/win10-1903

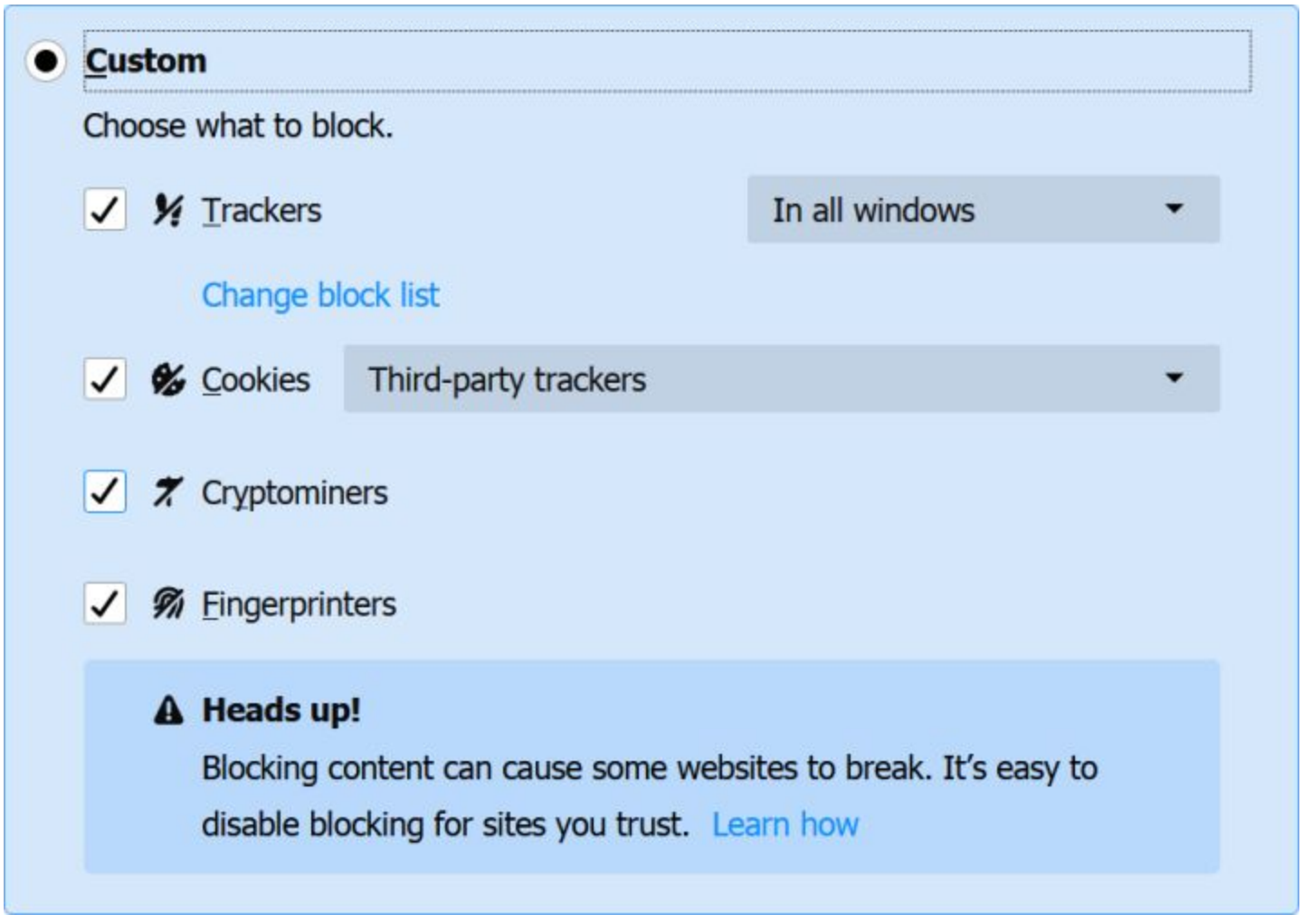
Mozilla just released Firefox 67 with a bunch of welcome enhancements.
FF67 improves Firefox's use of memory and also allows it to run faster. 67 has a smarter page renderer that no longer waits to load features that are not necessary to display a web page. This allows the browser to render the page more quickly. Mozilla has stated that Instagram, Amazon, and Google searches now execute 40-80% faster.

Previously, a browser extension was required to unload the storage used by idle tabs to save memory. I mentioned one recently"Auto Tab Discard." Now Firefox does this automatically. When less than 400MB memory is available, Firefox will begin suspending and unloading the browser's least recently used tabs.

Finally, browsers with many extensions installed would previously take longer to start. Mozilla has optimized Firefox's launching so that even heavily extended browsers will launch quickly.

67 also adds fingerprint and cryptomining blocking features.

To enable these new controls, set Content Blocking to Custom. That will display two new checkboxes:



When enabled, Firefox will respectively block known scripts and domains that utilize in-browser cryptomining and fingerprinting.

Firefox 67 also allows its Private Browsing to continue to use many extensions. When I went to remove the "Auto Tab Discard" add-on that I no longer need since it's now built-in, I noticed that all of the add-ons I use are now tagged with a big purple: "Allowed in Private Windows." Each extension can be enabled or disabled for its presence in Private Browsing mode, and whenever a new extension is installed Firefox will ask whether it should be available in Private Browsing mode.

Firefox's built-in login credential manager for storing and retrieving login username and password will now also be available in Private Browsing mode.

Mozilla is also getting ready to test-release their WebRender rendering engine to a small group of users. This engine will utilize the system's GPU for 2D rendering of web pages to further improve page rendering speed. This feature will initially be rolling out to Windows 10 users with NVIDIA graphics cards. As its functionality solidifies and any issues are resolved it will gradually become available to additional Firefox users throughout the year.

Abusing Code Signing for Profit
https://medium.com/@chroniclesec/abusing-code-signing-for-profit-ef80a37b50f4
https://gist.github.com/Blevene/6455fd7a898425d0546206d4be61fc68

Signing a Windows executable file was originally conceived as a mechanism to guarantee the authenticity and integrity of a file published on the internet. Since its inception, the process of cryptographically signing a piece of code was designed to give the Operating System a way to discriminate between legitimate and potentially malicious software. Unfortunately, this system is built on a problematic core tenet: Trust.

The chain of trust is relatively straight-forward: certificates are signed (issued) by trusted certificate authorities (CAs) , which have the backing of a trusted parent CA. This inherited trust model is taken advantage of by malware authors who purchase certificates directly or via resellers. Whether purchased directly or indirectly, due diligence into customers appears to be lacking. Revoking a certificate, the process by which a CA says the certificate is no longer trustworthy, is unfortunately the only real tool available to combat certificate abuse. This process introduces a delay in which malware with a certificate may be considered “trusted”.

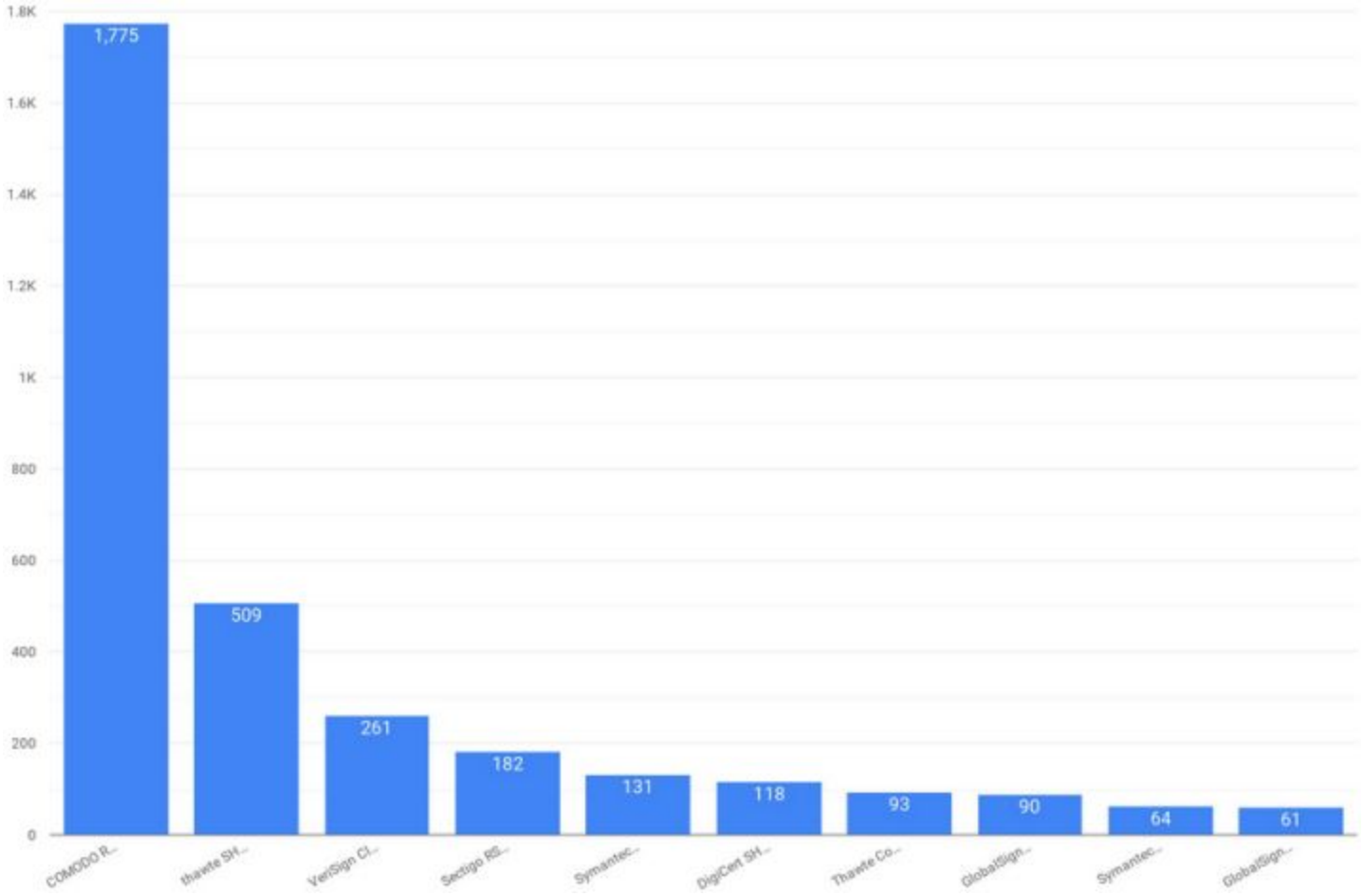
Chronicle researchers hunted within VirusTotal to gain a deeper understanding of this issue. For this investigation researchers only included Windows PE Executable files, filtered out samples with less than 15 aggregate detections, aggressively filtered out grayware files, and calculated the distinct number of samples each signing CA was responsible for (note: the samples may have different certificates, the focus is on the signing CA only). Data was collected within a 365 day span with an initial start date of May 7th, 2019.

So... what did they find??

In total, 3,815 malware samples met the filtering criteria. In other words, there's a LOT of true malware now being signed by valid certificates being issued by trusted Certificate Authorities.

And just guess who #1 is on the list... That's right, our old friends Comodo. And 4th on the list is Comodo under it's new name "Sectigo" -- which we recently discovered after some malware had been found signed by them. They've only been renamed since November 1st of last year, so they haven't had much time to catch up, but it seems clear they will shortly.

Of the 38,15 qualifying malware samples, just one Certificate Authority, Comodo+Sectigo provided the certificates that signed 1,957 of those - more than half, 51.3% ... all signed by the certificates issued by a single CA.



As the Chronicle guys note, there's a precipitous drop off in the numbers (and they didn't even combine Comodo and Sectigo for their chart)... "The CA with the most samples has nearly 3.5x more samples than the next highest, which in turn has almost 2x more than the next highest. The pattern quickly falls off as we move down the line of the top 10 CAs issuing abused certificates.

But they wrote that there is some hope!... "When evaluating this data we determined that 21% of samples had their certificates revoked at the time of writing (May 8th, 2019). This indicates that CAs are taking some action. Note that for the revocation of a certificate to be reflected in the VirusTotal dataset, the sample must be rescanned following the revocation request by the responsible CA.

What Does This Mean Going Forward?

While malware abusing trust is not a new phenomenon, the popular trend of financially motivated threat actors buying code signing certificates illuminates the inherent flaws of trust based security. Signed payloads are no longer solely within the domain of nation-state threat actors stealing code signing certificates from victims; they are readily accessible to operators of crime focused malware. The impact is amplified by the scope and scale of typical crimeware campaigns. Expect to see signed malware reported more frequently.

All hope is not lost. Certificate authorities are actively revoking certificates from malware executables that are identified in the wild. This indicates that CAs do take their responsibilities seriously, though more diligence around buyers may help prior to the proverbial cat being out of the bag.

=====

http://signedmalware.org/

RDP: Really Do Patch!

I chose to discuss the RDP problem further, since it's been quite awhile 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. By all appearances this one is not going to disappoint.

RDP is under the expected assault
The "wormable" Windows Remote Desktop Protocol (RDP) vulnerability that is bad enough that two weeks ago Microsoft reached all the way back to WinXP to patch is now officially named "BlueKeep" (CVE-2019-0708).

Last Thursday, on May 23rd, Dan Goodin for ArsTechnica wrote: "It has 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 the vulnerability was severe. Then five researchers from security firm McAfee reported 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, 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, 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 researcher at Check Point: Via Twitter: Eyal Itkin / @EyalItkin / The last 3 days were intense, but with help from the @_CPResearch_ team, we now have a working BSOD PoC for CVE-2019-0708. Time to catch some sleep.

Security researcher interested in reverse engineering, vulnerabilities, exploits, embedded (game consoles in particular). Kaspersky Lab: Boris Larin / @oct0xor / We analyzed the vulnerability CVE-2019-0708 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
The founder of Zerodium / Chaouki Bekrar / @cBekrar / We've confirmed exploitability of Windows Pre-Auth RDP bug (CVE-2019-0708) patched yesterday by Microsoft. Exploit works remotely, without authentication, and provides SYSTEM privileges on Windows Srv 2008, Win 7, Win 2003, XP. Enabling NLA mitigates the bug. Patch now or GFY!

#malware analyst with 20+ years of experience Low and high programmer. All opinions here are mine. / Valthek / @ValthekOn / I get the CVE-2019-0708 exploit working with my own programmed POC (a very real dangerous POC).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 dont care.

Lead Scientist- Sr. Principal Engineer McAfee: CVE-2019-0708 #BlueKeep - After many hours @ValthekOn was able to get a working PoC for this. We are not going to reveal technical details or release code. We urge everyone to PATCH - it is really nasty.. @Raj_Samani @John_Fokker @Seifreed @fr0gger_ @w3knight pic.twitter.com/W0aGXj2KTa
— Christiaan Beek (@ChristiaanBeek) May 18, 2019

https://securingtomorrow.mcafee.com/other-blogs/mcafee-labs/rdp-stands-for-really-do-patch-understanding-the-wormable-rdp-vulnerability-cve-2019-0708/

During Microsoft’s May Patch Tuesday cycle, a security advisory was released for a vulnerability in the Remote Desktop Protocol (RDP). What was unique in this particular patch cycle was that Microsoft produced a fix for Windows XP and several other operating systems, which have not been supported for security updates in years. So why the urgency and what made Microsoft decide that this was a high risk and critical patch?

According to the advisory, the issue discovered was serious enough that it led to Remote Code

Execution and was wormable, meaning it could spread automatically on unprotected systems. The bulletin referenced well-known network worm "WannaCry" which was heavily exploited just a couple of months =AFTER= Microsoft released MS17-010 as a patch for the related vulnerability, in March 2017. [In other words... The fact that a patch for WannaCry had been released months before didn't matter.] McAfee Advanced Threat Research has been analyzing this latest bug to help prevent a similar scenario and we are urging those with unpatched and affected systems to apply the patch for CVE-2019-0708 as soon as possible. [Unfortunately, as we know, the trouble lies with the systems put online by the people who are NOT reading this sort of security news.] It is extremely likely malicious actors have weaponized this bug and exploitation attempts will likely be observed in the wild in the very near future.

Vulnerable Operating Systems:

- Windows 2003
- Windows XP
- Windows 7
- Windows Server 2008
- Windows Server 2008 R2

Looking inside the RDP protocol...

[I'm going to share the description of the problem. It uses a bunch of terms and presumptions that we have defined, and there's really no point in bothering to get down into the nitty gritty. But it's use to get a FEEL for the problem, and that's something we can get without understanding every nuance. So here goes...]

The Remote Desktop Protocol (RDP) enables connections between a client and endpoint, defining the data communicated between them in virtual channels. Virtual channels are bidirectional data pipes which enable the extension of RDP. Windows Server 2000 defined 32 Static Virtual Channels (SVCs) with RDP 5.1, but due to limitations on the number of channels further defined Dynamic Virtual Channels (DVCs), which are contained within a dedicated SVC. 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 within the _IcaBindVirtualChannels and _IcaRebindVirtualChannels functions in the RDP driver termdd.sys. As can been seen in figure 1, the RDP Connection Sequence connections are initiated and channels setup 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.

The vulnerability is due to the "MS_T120" SVC 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 named "MS_T120."

However, during GCC Conference Initialization, the Client supplies the channel name which is not whitelisted by the server, meaning an attacker can setup another SVC named "MS_T120" on a channel other than 31. It's the use of MS_T120 in a channel other than 31 that leads to heap

memory corruption and remote code execution (RCE).

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 (1Fh) in the _IcaBindVirtualChannels and _IcaRebindVirtualChannels functions within termdd.sys.

[So... translating this... There was some flexibility in the protocol that was being used in its default expected way. But when that flexibility was used in a different, though technically proper fashion, a corruption of memory resulted which a sufficiently clever attacker could leverage to run code they supply to the remote server. And, since the corrupted heap pool is allocated by the rdpwp.sys driver, which resides in the kernel, a remote kernel exploit would be the result.]

<McAfee...> After we investigated the patch being applied for both Windows 2003 and XP and understood how the RDP protocol was parsed before and after patch, we decided to test and create a Proof-of-Concept (PoC) that would use the vulnerability and remotely execute code on a victim's machine to launch the calculator application, a standard litmus test for remote code execution.
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. Network Level Authentication should be effective to stop this exploit if enabled; however, if an attacker has credentials, they will bypass this step.

As a patch is available, we decided not to provide earlier in-depth detail about the exploit or publicly release a proof of concept. That would, in our opinion, not be responsible and may further the interests of malicious adversaries.

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.

[I'm a bit annoyed that McAfee effectively suggested changing the RDP port, suggesting that this would provide useful security. No one should rely upon port obfuscation for security in this case. The ONLY way to use RDP safely is behind a properly configured OpenVPN server to control access.]

So, as of TODAY, we have Zerodium, McAfee, Kaspersky, Check Point, MalwareTech, and Valthek all having confirmed that they have successfully developed exploits for BlueKeep. None of them is publishing, but there have been cross-verifications. The trouble is... This strongly suggests that the vulnerability is decidedly NOT difficult to weaponize.

=====

RiskSense security researcher Sean Dillon has created a tool to allow companies to test whether their PC fleets have been correctly patched against the BlueKeep flaw.

RiskSense's is packaged in a Docker container:
https://github.com/zerosum0x0/CVE-2019-0708

There's also a Metasploit module: https://github.com/rapid7/metasploit-framework/pull/11869

And Robert Graham (ErrataRob) as trimmed it down and produce a nice higher-speed "C" scanner: https://github.com/robertdavidgraham/rdpscan

Rob writes: "rdpscan for CVE-2019-0708 bluekeep vuln"

This is a quick-and-dirty scanner for the CVE-2019-0708 vulnerability in Microsoft Remote Desktop. Right now, there are about 700,000 machines on the public Internet vulnerable to this vulnerability, compared to about 2,000,000 machines that have Remote Desktop exposed, but are patched/safe from exploitation. Many expect that in the next few months a devestating 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 vulnerable machines.

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.

This tool is based entirely on the rdesktop patch from https://github.com/zerosum0x0/CVE-2019-0708. 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.

This is only a couple days old and experimental. However, I am testing it by scanning the entire Internet (with the help of masscan, so I'm working through a lot of problems pretty quickly. You can try contacting me on twttier(@erratarob) for help/comments.

2019-05-27 - Windows and macOS binaries released (click on badges above). You Linux peeps get only source as usual. It seems to be working well on all three platforms.

=====

And, now... BlueKeep scans started over the weekend.

GreyNoise Intelligence / @GreyNoiseIO / GreyNoise is observing sweeping tests for systems vulnerable to the RDP "BlueKeep" (CVE-2019-0708) 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.

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 have only observed 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.

But, as I've noted, with at least six research groups announcing that they've come up with private BlueKeep exploits, and with at least two very detailed write-ups on the BlueKeep vulnerability details available online (McAfee's and "WazeHell's"), it's only a matter of time -- and probably not very much time -- until bad guys come up with their own exploits as well.

The Tor-originating scans that GreyNoise is currently seeing -- and which Morris told ZDNet are still ongoing -- are a first sign that things are about to get worse. Really worse!

And since then, Robert Graham has confirmed from his own scan that 700,000 machines are CURRENTLY -- right now -- vulnerable. So whomever is scanning out through TOR is cataloging the same 700,000 machines.

And then, just this morning, Robert has posted to his own ErrataSecurity Blog:
https://blog.erratasec.com/2019/05/almost-one-million-vulnerable-to.html

"Almost One Million Vulnerable to BlueKeep Vuln (CVE-2019-0708)"

Microsoft announced a vulnerability in it's "Remote Desktop" product that can lead to robust, wormable exploits. I scanned the Internet to assess the danger. I find nearly 1-million devices on the public Internet that are vulnerable to the bug. 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.

To scan the Internet, I started with masscan, my Internet-scale port scanner, looking for port 3389, the one used by Remote Desktop. This takes a couple hours, and lists all the devices running Remote Desktop -- in theory.

This returned 7,629,102 results (over 7-million). However, 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 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, 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:

1,447,579	UNKNOWN	- receive timeout
1,414,793	SAFE	- Target appears patched
1,294,719	UNKNOWN	- connection reset by peer
1,235,448	SAFE	- CredSSP/NLA required
923,671	**VULNERABLE**	**- got appid**
651,545	UNKNOWN	- FIN received
438,480	UNKNOWN	- connect timeout
105,721	UNKNOWN	- connect failed 9
82,836	SAFE	- not RDP but HTTP
24,833	UNKNOWN	- connection reset on connect
3,098	UNKNOWN	- network error
2,576	UNKNOWN	- connection terminated

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. A lot of others are Windows machines, sometimes vulnerable and 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 "pached". 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 are safe from this vulnerability. There are 1,414,793 of them.

The next result to look at are those marked SAFE due to CredSSP/NLA failures, of which there are 1,235,448. This doesn't mean they are patched, but only that we can't exploit them. They require "network level authentication" first before we can talk Remote Desktop to them. That means we can't test whether they are patched or vulnerable -- but neither can the hackers. They may still be exploitable via an insider threat who knows a valid username/password, but they aren't exploitable by anonymous hackers or worms.

The next category is marked as SAFE because they aren't Remote Desktop at all, but HTTP servers. In other words, in response to our Remote Desktop request they send an HTTP response. There are 82,836 of these.

Thus, out of 7.6-million devices that respond to port 3389, we find 3.5-million that reliably talk the Remote Desktop protocol, of which 0.9-million are vulnerable, and the rest are not.

But, since a lot of those "unknowns" are due to transient network errors, then in theory I should be able to rescan them and get some more results. I did this and go the following update:

28,182	SAFE	- Target appears patched
19,991	**VULNERABLE- got appid**	
17,560	SAFE	- CredSSP/NLA required
695	SAFE	- not RDP but HTTP

A third rescan got the following results:

9,838	SAFE	- Target appears patched
7,084	SAFE	- CredSSP/NLA required
6,041	**VULNERABLE- got appid**	
2,963	UNKNOWN	- network error
45	SAFE	- not RDP but HTTP

Some of these rescans are likely overcoming transient errors that preventing getting results the first time. However, others are likely ISPs with Windows machines moving around from one IP address to another, so that continued rescans are going to get distorted results rather than cleaning up the previous results.

The upshot is that these tests confirm that roughly 950,000 machines are on the public Internet that are vulnerable to this bug. Hackers are likely to figure out a robust exploit in the next month or two and cause havoc with these machines.

There are two things you should do to guard yourself. The first is to apply Microsoft's patches, including old Windows XP, Windows Vista, and Windows 7 desktops and servers.

More importantly, for large organizations, is to fix their psexec problem that allows such things to spread via normal user networking. You may have only one old WinXP machine that's vulnerable, that you don't care if it gets infected with ransomware. But, that machine may have a Domain Admin logged in, so that when the worm breaks in, it grab those credentials and uses them to log onto the Domain Controller. Then, from the Domain Controller, the worm sends a copy of itself to all the desktop and servers in the organization, using those credentials instead of the vuln. This is what happened with notPetya: the actual vulnerability wasn't the problem, it was psexec that was the problem.

For patching systems, you have to find them on the network. My rdpscan tool mentioned above is good for scanning small networks. For large networks, you'll probably want to do the same masscan/rdpscan combination that I used to scan the entire Internet. On GitHub, rdpscan has precompiled programs that work on the command-line, but the source is there for you to compile it yourself, in case you don't trust I'm tryin to infect you with a virus.

=====

A non-native English write-up of the "DIFFing" process used to reverse-engineer this patch.
https://wazehell.io/2019/05/22/cve-2019-0708-technical-analysis-rdp-rce/

This world has a real problem that needs to be fixed RIGHT NOW.

Code Red was not the last worm.
Nimda was not.
MS Blast was not.
WannaCry was not.
notPetya was not.

Nor will whatever nightmare we name this BlueKeep worm bee the last.

A 22-instruction in-RAM "micropatch" has been developed by the 0Patch guys that doesn't even require a reboot of the affected system. NOTHING other than laws that do NOT make sense in this situation prevents Microsoft from doing the same thing as Robert Graham and 0patch ... to scan the entire Internet and fix this mistake in RAM.

If this were permissible Microsoft could have closed this hole BEFORE announcing it to the world and fixing it correctly.

We would need to have some control over it. Some oversight body composed of security industry and appropriate government leaders to whom Microsoft would explain the problem and make their case.

~30~