

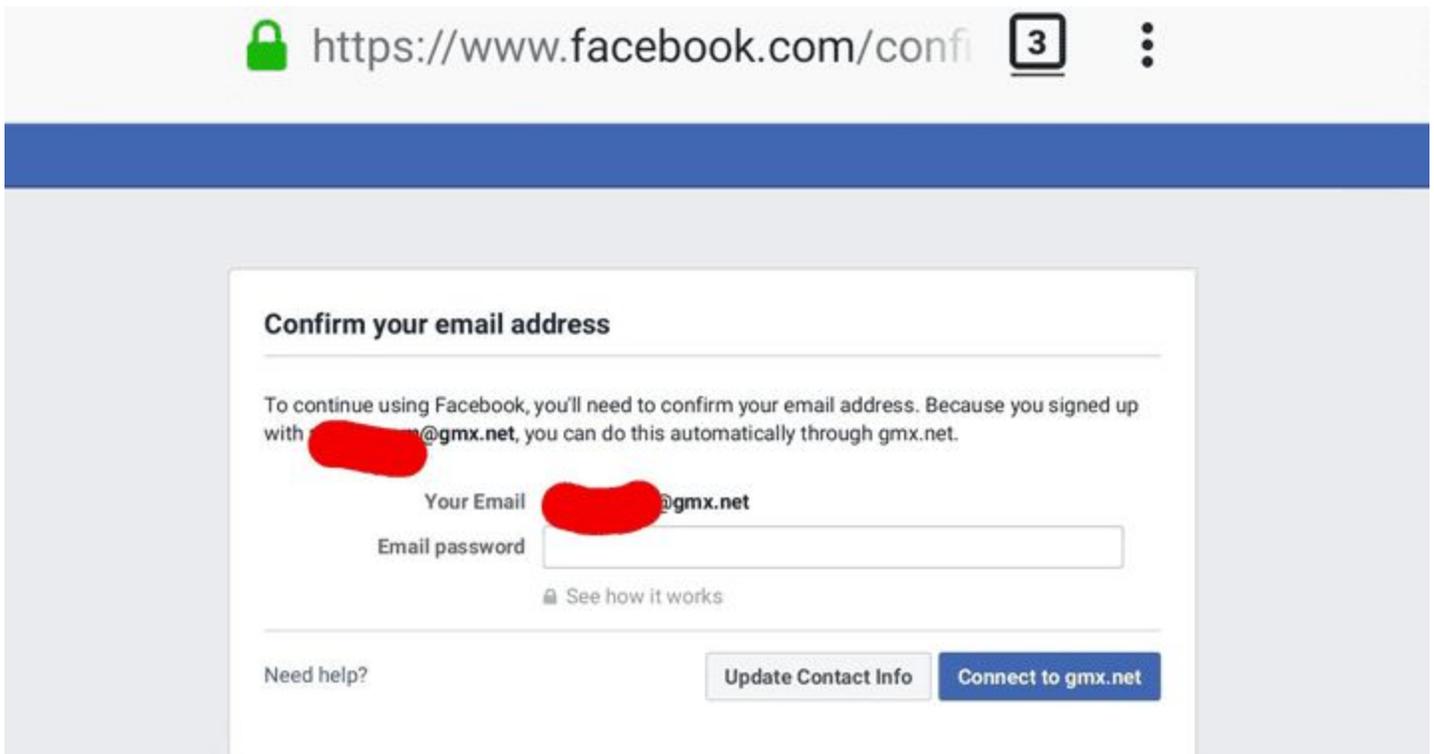
Security Now! #709 - 04-09-19

URL "Ping" Tracking

This week on Security Now!

This week we discuss more news of Microsoft's Chromium-based Edge browser, the UK government's plan to legislate, police and enforce online social media content, improvements to Windows 10's update management, news from the "spoofing biometrics" department, the worrisome state of Android mobile financial apps, an update on the NSA's GHIDRA software reverse engineering tool suite, perhaps the dumbest thing Facebook has done yet (and by policy, not by mistake), an important change in Win10 1809 external storage caching policy, a bit of miscellany and closing the loop feedback from our terrific listeners... then we're going to take a close look at another capitulation in the (virtually lost) battle against tracking our behavior on the Internet with URL "ping" tracking.

If we didn't have clear, multi-sourced, verified proof and evidence of this, including Facebook's own acknowledgement that they planned to stop doing it... I wouldn't have believed it. In fact... I had to make SURE that the story didn't first appear on April 1st!



Security News

Microsoft's Chromium-based Edge Browser is Officially Available

<https://www.microsoftedgeinsider.com/en-us/download/>

Windows 10 now available. Coming soon to Win7, 8, 8.1 and macOS.

Whereas the Windows 10 Insider builds are only available to members of the Insider program, the Microsoft Edge Insider builds are available to everyone and can be downloaded from the Microsoft Edge Insider site.

Three update "channels" are available:

The Beta channel - the most stable browser updated approximately every 6 weeks.

The Dev channel - more stable than the nightly Canary, but not yet ready for Beta. It's working toward being the next major release.

The Canary channel - the cutting edge state of the browser, updated nightly and it's prototyping the version that will be two versions from the current release.

Microsoft's Joe Belfore stated in an announcement: "In these first builds we are very much focused on the fundamentals and have not yet included a wide range of feature and language support that will come later. You'll start to see differences from the current Microsoft Edge including subtle design finishes, support for a broader selection of extensions and the ability to manage your sign-in profile. We look forward to people starting to kick the tires and will be refining the feature set over time based on the feedback we receive."

To enable Chrome extensions in the Microsoft Edge Insider build:

1. Select the "Extensions" entry on the Edge Menu.
2. Toggle on the setting "Allow extensions from other stores." at the bottom of the settings panel.

Online Harms White Paper

Yesterday, the UK government announced a suite of online safety laws which intend to hold the publishers of online social media platforms liable for the harmful behavior spreading through their platforms.

<https://www.gov.uk/government/consultations/online-harms-white-paper>

It's called the "Online Harms White Paper" and it's a joint proposal published by the Department for Digital, Culture, Media & Sport and the UK Home Office, the law package "comprises legislative and non-legislative measures and will make companies more responsible for their users' safety online, especially children and other vulnerable groups."

For the next three months, until July 1st, the Online Harms White Paper is under an open consultations status to allow the government to collect opinions from "organisations, companies and others with relevant views, insights or evidence" regarding the future online safety regulatory framework.

Consultation description

The Online Harms White Paper sets out the government's plans for a world-leading package of online safety measures that also supports innovation and a thriving digital economy. This package comprises legislative and non-legislative measures and will make companies more responsible for their users' safety online, especially children and other vulnerable groups.

The White Paper proposes establishing in law a new duty of care towards users, which will be overseen by an independent regulator. Companies will be held to account for tackling a comprehensive set of online harms, ranging from illegal activity and content to behaviours which are harmful but not necessarily illegal.

This consultation aims to gather views on various aspects of the government's plans for regulation and tackling online harms, including:

- the online services in scope of the regulatory framework;
- options for appointing an independent regulatory body to implement, oversee and enforce the new regulatory framework;
- the enforcement powers of an independent regulatory body;
- potential redress mechanisms for online users; and
- measures to ensure regulation is targeted and proportionate for industry.

This is an open public consultation. We particularly encourage responses from organisations, companies and others with relevant views, insights or evidence.

The paper itself is 102 pages, so we won't slog our way through it here. But a few highlights from the separate executive summary include:

1. The government wants the UK to be the safest place in the world to go online, and the best place to start and grow a digital business. Given the prevalence of illegal and harmful content online, and the level of public concern about online harms, not just in the UK but worldwide, we believe that the digital economy urgently needs a new regulatory framework to improve our citizens' safety online. This will rebuild public confidence and set clear expectations of companies, allowing our citizens to enjoy more safely the benefits that online services offer.

7. This White Paper sets out a programme of action to tackle content or activity that harms individual users, particularly children, or threatens our way of life in the UK, either by undermining national security, or by undermining our shared rights, responsibilities and opportunities to foster integration.

8. There is currently a range of regulatory and voluntary initiatives aimed at addressing these problems, but these have not gone far or fast enough, or been consistent enough between different companies, to keep UK users safe online.

9. Many of our international partners are also developing new regulatory approaches to tackle online harms, but none has yet established a regulatory framework that tackles this range of online harms. The UK will be the first to do this, leading international efforts by setting a

coherent, proportionate and effective approach that reflects our commitment to a free, open and secure internet.

12. Our vision is for:

- A free, open and secure internet
- Freedom of expression online
- An online environment where companies take effective steps to keep their users safe, and where criminal, terrorist and hostile foreign state activity is not left to contaminate the online space
- Rules and norms for the internet that discourage harmful behaviour
- The UK as a thriving digital economy, with a prosperous ecosystem of companies developing innovation in online safety
- Citizens who understand the risks of online activity, challenge unacceptable behaviours and know how to access help if they experience harm online, with children receiving extra protection
- A global coalition of countries all taking coordinated steps to keep their citizens safe online
- Renewed public confidence and trust in online companies and services

And so on. Those were a few of the first 12 points. It goes on for a total of 49. But everyone here gets the idea. The UK is going to impose regulatory standards backed by substantial fines. It's pretty clear now that some social media platforms have suck up massive advertising revenue while taking the stance that the Internet is open and free and that no one on it should be regulated. Recently, Facebook's Mark Zuckerberg has been saying that he wants and welcomes much-needed regulation. Be careful what you wish for. It is indeed coming.

Windows 10 Feature Update management to become much more clear

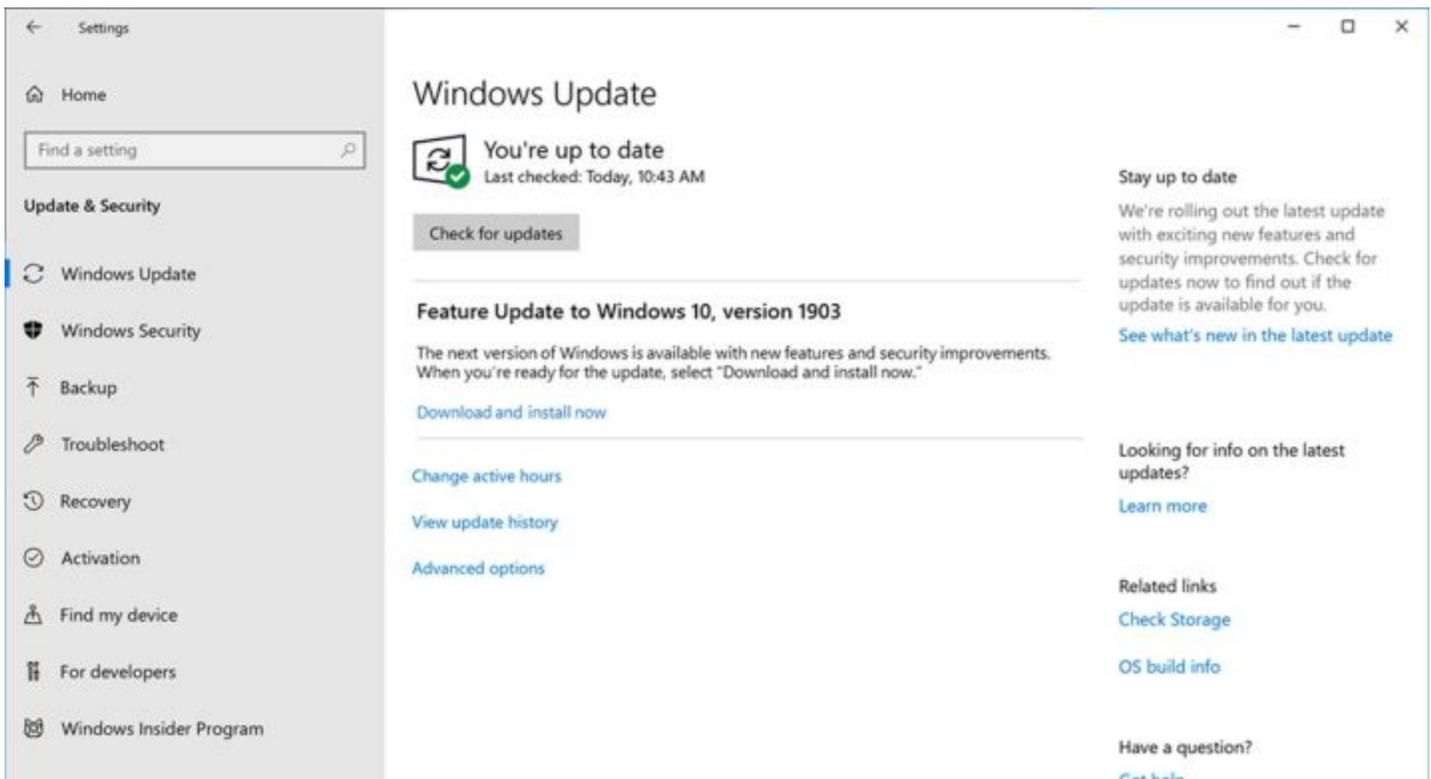
<https://blogs.windows.com/windowsexperience/2019/04/04/improving-the-windows-10-update-experience-with-control-quality-and-transparency/>

Posted last Thursday, April 4 by Mike Fortin / Corporate Vice President, Windows
"Improving the Windows 10 update experience with control, quality and transparency"

<quote> In previous Windows 10 feature update rollouts, the update installation was automatically initiated on a device once our data gave us confidence that device would have a great update experience. Beginning with the Windows 10 May 2019 Update, users will be more in control of initiating the feature OS update. We will provide notification that an update is available and recommended based on our data, but it will be largely up to the user to initiate when the update occurs. When Windows 10 devices are at, or will soon reach, end of service, Windows update will continue to automatically initiate a feature update; keeping machines supported and receiving monthly updates is critical to device security and ecosystem health. We are adding new features that will empower users with control and transparency around when updates are installed. In fact, all customers will now have the ability to explicitly choose if they want to update their device when they "check for updates" or to pause updates for up to 35 days.

We are taking further steps to be confident in the quality of the May 2019 Update. We will increase the amount of time that the May 2019 Update spends in the Release Preview phase, and we will work closely with ecosystem partners during this phase to proactively obtain more early feedback about this release. This will give us additional signals to detect issues before broader deployment. We are also continuing to make significant new investments in machine learning (ML) technology to both detect high-impact issues efficiently at scale and further evolve how we intelligently select devices that will have a smooth update experience.

[The] "Download and install now" option provides users a separate control to initiate the installation of a feature update on eligible devices with no known key blocking compatibility issues. Users can still "Check for updates" to get monthly quality and security updates. Windows will automatically initiate a new feature update if the version of Windows 10 is nearing end of support. We may notify you when a feature update is available and ready for your machine. All Windows 10 devices with a supported version will continue to automatically receive the monthly updates. This new "download and install" option will also be available for our most popular versions of Windows 10, versions 1803 and 1809, by late May.



News from the "Spoofing Biometrics" Department

Samsung's recently released Galaxy S10 top-of-the-line smartphone has a spiffy new fingerprint reader.

We all recall Apple's fingerprint reader. It uses capacitive technology to image the ridges of our fingertips. When it was released it was noted that since it was capacitive, it would not be spoofed by flat images. And as we also know, it was only a few days after it got into the hands of some creative hackers that a fake thumb with ridges was created to spoof Apple's capacitive technology.

So now we jump forward a few years to Samsung: Everyone is all freaked out over the seamlessness of the screen, with much angst over the "Apple iPhone Ears" created by the encroachment of the iPhone's 3D facial recognition technology. Samsung, determined to minimize the encroachment of any UI technology set their engineers the task of incorporating a fingerprint reader INTO THE SCREEN so that there would be NO physical zone of any kind. They succeeded and created an acoustic ultrasonic imaging technology whose transducer is placed behind the screen and which operates THROUGH the screen.

And, we're talking about this because, as with Apple's capacitive "3D fingerprint" imaging technology, it didn't take clever hackers long to spoof the Samsung ultrasonic fingerprint reader.

Last Wednesday, a Reddit user calling himself "darkshark9" posted his hack of his Samsung S10.

<https://imgur.com/gallery/8aGqsSu>

He unlocked his Samsung Galaxy S10 using his 3D printed fingerprint picked up from a photo of a wine glass taken using the smartphone. However, to make his accomplishment more striking he noted that the fingerprint image could be captured at greater distance using a DSLR camera to steal one's fingerprint from across the room or even from a lot farther away with the help of a telephoto lens.

<quote> I pulled the image into Photoshop and increased the contrast, and created an alpha mask. I exported that over to 3DS Max and created a geometry displacement from the Photoshop image which gave me a raised 3D model of every last detail of the fingerprint. I popped that model into the 3D printing software and began to print it. This was printed using an AnyCubic Photon LCD resin printer, which is accurate down to about 10 microns (in Z height, 45 microns in x/y), which is more than enough detail to capture all of the ridges in a fingerprint. It printed perfectly. Print time was only around 13 minutes.

It took me 3 prints trying to get the right ridge height (and I forgot to mirror the fingerprint on the first one) but yeah, 3rd time was the charm. The 3D print unlocks my phone...in some cases just as well as my actual finger does.

This brings up a lot of ethics questions and concerns. There's nothing stopping me from stealing your fingerprints without you ever knowing, then printing gloves with your fingerprints built into them and going and committing a crime.

If I steal someone's phone, their fingerprints are already on it. I can do this entire process in

less than 3 minutes and remotely start the 3d print so that it's done by the time I get to it. Most banking apps only require fingerprint authentication so I could have all of your info and spend your money in less than 15 minutes if your phone is secured by fingerprint alone. </quote>

So where does this take us?

Sure... We COULD attempt to make the recognition process much more robust and less spoofable. We could run some very weak AC through the fingertip tissue to determine impedance. All existing spoofs would fail that test. Or we could send several different frequencies of light into the finger and determine some measures of relative spectral tissue absorption. And we could also watch this over the span of a few heartbeats to detect capillary pulse using photo plethysmography.

We know that Apple did go to great lengths to make their FaceID largely immune to simple spoofs. It incorporates infrared imaging to obtain a 3D model of the users face specifically so that showing it a photo won't fool it. But capturing 3D images of people is no longer the stuff of science fiction. Consumer apps do it now regularly.

So I think that the right way to think of this, is soberly. It is the classic case of trading off convenience for security. Biometric systems are incredibly convenient to use, but unless their technology is raised to the point where their cost becomes prohibitive in consumer settings -- and then also far more prone to refusing to accept that you really are you, which would be quite annoying -- they are going to be very easily spoofable. So, if you want to use TouchID and FaceID and the like, by all means do so. But recognize that these devices ARE inherently spoofable and treat your devices accordingly. Apple, for one, has been good about the phone's lock feature forcing a first-subsequent-use fallback to manual passcode entry to then unlock the use of biometrics. They understood the tradeoffs even if their marketing pitches downplay them.

The worrisome state of Android mobile financial apps

Although it's a bit self-serving, because San Francisco-based Arxan Technologies bills itself as "the trusted provider of application protection solutions", they generated some news recently with a press release highlighting the results of their analysis of a group of popular mobile financial apps.

Their PR reported their discovery of "widespread security inadequacies and protection failures among consumer financial applications, leading to the exposure of source code, sensitive data stored in apps, access to back-end servers via APIs, and more."

Senior cybersecurity analyst Alissa Knight of global research and advisory firm Aite Group authored the study, titled: 'In plain sight: The vulnerability epidemic in financial services mobile apps.'

Knight examined the mobile apps of 30 financial institutions downloaded from the Google Play store across eight financial services sectors: retail banking, credit card, mobile payment, cryptocurrency, HSA, retail brokerage, health insurance, and auto insurance. Using tools readily available on the internet, Knight found nearly all of the applications could easily be reverse engineered allowing access to sensitive information stored inside the source code, such as

improperly stored Personal Identifying Information, account credentials, server-side file locations, API keys, live deployment and QA URLs used by the developers for testing the apps. The research highlights a systemic lack of application-appropriate protection such as application shielding, threat detection, encryption, and response technology across financial services apps.

Analysis of the mobile financial industry applications highlighted major deficiencies in application design including easily reverse engineered code that exposes serious vulnerabilities including in-app data storage; compromised data transmission due to weak data encryption and insufficient transport layer protection; and malware injection/tampering.

The Aite Group's key research findings include:

Lack of Binary Protections — 97% of all apps tested lacked binary code protection, making it possible to reverse engineer or decompile the apps exposing source code to analysis and tampering

- Unintended Data Leakage — 90% of the apps tested shared services with other applications on the device, leaving data from the financial institution's app accessible to any other application on the device
- Insecure Data Storage — 83% of the apps tested insecurely stored data outside of the apps control, for example, in a device's local file system, external storage, and copied data to the clipboard allowing shared access with other apps; and, exposed a new attack surface via APIs
- Weak Encryption — 80% of the apps tested implemented weak encryption algorithms or the incorrect implementation of a strong cipher, allowing adversaries to decrypt sensitive data and manipulate or steal it as needed
- Insecure Random-Number Generation — 70% of the apps use an insecure random-number generator, making the values easily guessed and hackable

Alissa Knight wrote: "During this research project, it took me 8.5 minutes on average to crack into an application and begin to freely read the underlying code, identify APIs, read file names, access sensitive data and more. With financial institutions holding such sensitive financial and personal data — and operating in such stringent regulatory environments — it is shocking to see just how many of their applications lack basic secure coding practices and app security protections. The large number of vulnerabilities exposed from decompiling these applications poses a direct threat to financial institutions and their customers. These resulting threats ranged from account takeovers, credit application fraud, synthetic identity fraud, identity theft and more. It's clear from the findings that the industry needs to address the vulnerability epidemic throughout its mobile apps and employ a defense-in-depth approach to securing mobile applications — starting with app protection, threat detection and encryption capabilities implemented at the code level. Of all the findings, the most shocking was without a doubt, the SQL queries exposing information on the backend databases hard coded in the app along with private keys being stored unencrypted in different sub-directories."

Yikes! In other words, these readily reverse-engineerable apps are using readily discoverable private keys to access sensitive financial institution backend databases via poorly encrypted SQL queries. We don't know anything about the security or accessibility of the backend databases, but if the same developers who implemented the front end had anything to do with the backend, it doesn't look good.

Note that we have no reason to believe that the companion iOS apps are likely every bit as poorly designed. As we know, iOS is more tightly closed, but it's now standard practice for a common code base to be used across multiple target platforms with only the user-interface customized per-platform. On the other hand, if the attacks are against financial institution backend databases and services, going in through the Android version of the app would likely be quicker and easier, as this researcher discovered.

And the fact that knowledge of this worrisome lack of proper design is now public doesn't help either. Thanks to the NSA's release of GHIDRA, the bar has not been lowered, it's been dropped to the floor, on reverse-engineering binary code.

Table A: Vulnerabilities Found Across All Financial Sectors

| Type of app | Lack of binary protections | Insecure data storage | Unintended data leakage | Client-side injection | Weak encryption | Implicit trust of all certificates |
|--------------------|----------------------------|-----------------------|-------------------------|-----------------------|-----------------|------------------------------------|
| Retail bank | ● | ● | ● | ◐ | ● | ◐ |
| Credit card issuer | ◐ | ● | ● | ◐ | ● | ○ |
| Mobile payments | ● | ● | ● | ◐ | ● | ○ |
| HSA bank | ● | ◐ | ◐ | ◐ | ◐ | ○ |
| Retail brokerage | ◐ | ◐ | ● | ◐ | ◐ | ◐ |
| Health insurer | ● | ◐ | ◐ | ◐ | ◐ | ○ |
| Auto insurer | ● | ● | ● | ◐ | ◐ | ○ |
| Cryptocurrency | ◐ | ◐ | ◐ | ◐ | ◐ | ◐ |

Source: Aite Group

Legend: ○ = 0% of the apps tested exhibited the vulnerability; ◐ = 25% of the apps tested exhibited the vulnerability; ◑ = 50% of the apps tested exhibited the vulnerability; ◒ = 75% of the apps tested exhibited the vulnerability; ● = 100% of the apps tested exhibited the vulnerability

Table B: Vulnerabilities Found Across All Financial Sectors (Continued)

| Type of app | Execution of activities as root | World readable/writable files and directories | Private key exposure | Exposure of database parameters and SQL queries | Insecure random number generation |
|--------------------|---------------------------------|---|----------------------|---|-----------------------------------|
| Retail bank | ● | ○ | ○ | ● | ● |
| Credit card issuer | ● | ○ | ○ | ● | ● |
| Mobile payments | ○ | ○ | ○ | ● | ● |
| HSA bank | ● | ○ | ○ | ● | ● |
| Retail brokerage | ○ | ○ | ○ | ● | ● |
| Health insurer | ● | ○ | ○ | ● | ● |
| Auto insurer | ○ | ○ | ○ | ● | ● |
| Cryptocurrency | ● | ○ | ● | ● | ● |

Source: Aite Group

Legend: ○ = 0% of the apps tested exhibited the vulnerability; ○ = 25% of the apps tested exhibited the vulnerability; ○ = 50% of the apps tested exhibited the vulnerability; ● = 75% of the apps tested exhibited the vulnerability; ● = 100% of the apps tested exhibited the vulnerability

And speaking of the NSA and GHIDRA...

Last Thursday, the widely anticipated release of the full source code for the NSA's GHIDRA reverse engineering tour de force appeared on Github:

<https://github.com/NationalSecurityAgency/ghidra/tree/master/Ghidra>

Written in JAVA, cross platform Windows, macOS and Linux.

Incredible processor support:

X86 16/32/64, ARM/AARCH64, PowerPC 32/64, VLE, MIPS 16/32/64, micro, 68xxx, Java / DEX bytecode, PA-RISC, PIC 12/16/17/18/24, Sparc 32/64, CR16C, Z80, 6502, 8051, MSP430, AVR8, AVR32, and more...

The official site: <https://ghidra-sre.org/>

Keyboard Sortcuts Cheat Sheet: <https://ghidra-sre.org/CheatSheet.html>

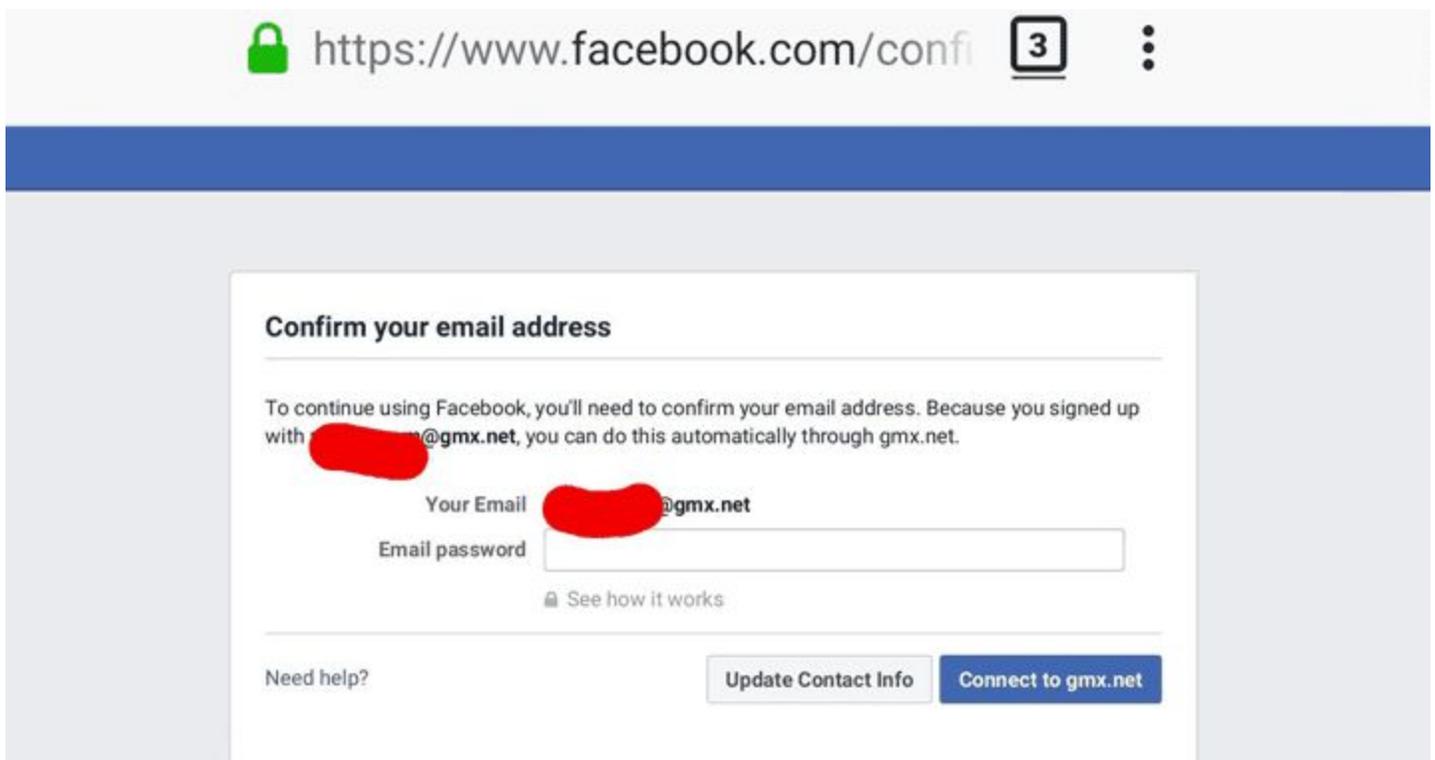
GHIDRA, being available for free, rather than for the multi-thousands of dollars per seat commercial IDA Pro reverse engineering tool, this promises to change the landscape of reverse engineering.

And we already have our first community-reported bug:

Matthew Hickey, who goes by "HackerFantastic" online reported a security issue in GHIDRA. He noticed that GHIDRA opens JDWP debug port 18001 bound to =all= of the system's network interfaces, rather than to just the system's localhost port, when a user launches GHIDRA in its debug mode. This would have allowed anyone (the bug was quickly fixed) with access to that machine over the network to remotely execute arbitrary code on the analysts' system. Whoops!

Believe it or not, Facebook...

(And I can hardly believe that I'm saying this, even of Facebook)... was actually asking some of its users to verify their identity by providing their private eMail address. I'm not kidding. Here's a screenshot:



For those listening, the Facebook dialog reads... (see above)

The Daily Beast wrote:

Just two weeks after admitting it stored hundreds of millions of its users' own passwords insecurely, Facebook is demanding some users fork over the password for their outside email account as the price of admission to the social network.

Facebook users are being interrupted by an interstitial demanding they provide the password for the email account they gave to Facebook when signing up. "To continue using Facebook, you'll need to confirm your email," the message demands. "Since you signed up with [email address], you can do that automatically ..."

A form below the message asked for the users' "email password."

Small print below the password field promises, "Facebook won't store your password." But the company has recently been criticized for repurposing information it originally acquired for "security" reasons.

In a statement emailed to The Daily Beast after this story published, Facebook reiterated its claim it doesn't store the email passwords. But the company also announced it will end the practice altogether.

"We understand the password verification option isn't the best way to go about this, so we are going to stop offering it," Facebook wrote.

Windows 10, version 1809 and later:

Change in default removal policy for external storage media

<https://support.microsoft.com/en-us/help/4495263/windows-10-1809-change-in-default-removal-policy-for-external-media>

Summary

Windows defines two policies, Quick removal and Better performance, that control how the system interacts with external storage devices such as USB thumb drives or Thunderbolt-enabled external drives. Beginning in Windows 10 version 1809, the default policy is Quick removal.

In earlier versions of Windows the default policy was Better performance.

You can change the policy setting for each external device, and the policy that you set remains in effect if you disconnect the device and then connect it again to the same computer port.

Translation: Historically, Windows has always enabled Write caching for all mass storage, internal and external. This is especially crucial for external non-volatile thumb drives since (a) they write so very slowly and (b) writing inherently degrades their storage cells and shortens their lives.

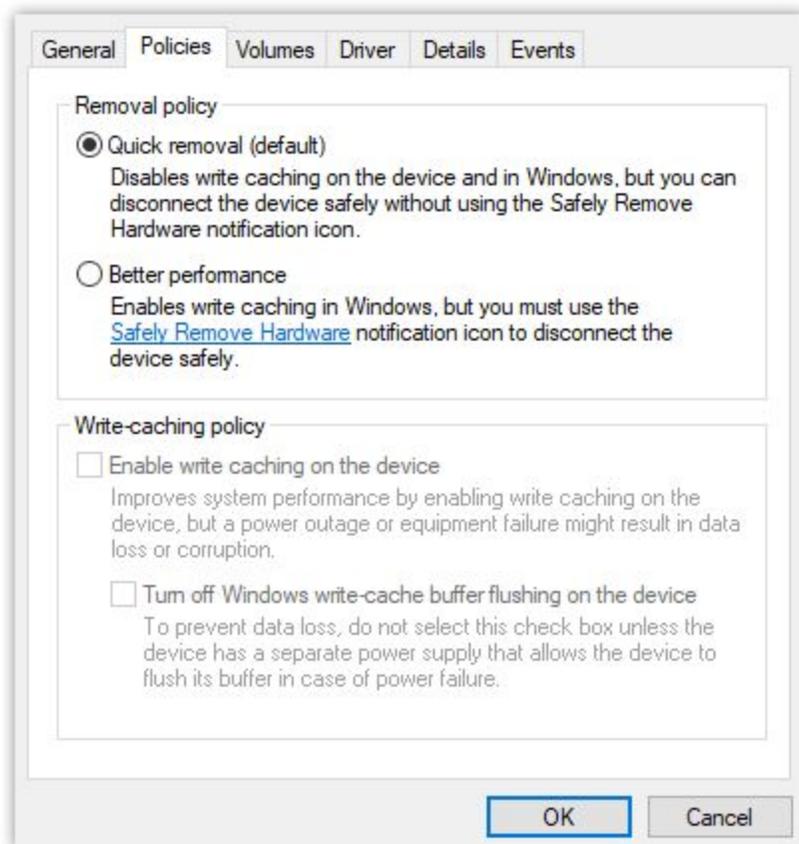
However, users were not reliably using the "Safe Eject" function for their external USB Thumb drives, and they were pulling the drives out of the machine before the drive's writing cached data had been fully flushed and written to the drive.

So, after lo these many years, we've decided to err in the direction of caution and to protect users from themselves by disabling USB device write caching by default, thus protecting users from themselves at the cost of dramatically slowing down all USB write operations -- and prematurely aging write-intolerant thumb drives.

Due to multiple writes frequently occurring to the same block, disabling all external drive caching doesn't just push the timing out to the end, it dramatically slows down the entire process when writing to the drive -- and it also shortens the drive's life -- since disabling caching prevents the coalescing of multiple writes to the same block -- which is a crucially important function of write caching. And multiple writes to the same block happens a LOT within the file system allocation tables where allocation bitmaps are maintained. Flipping individual allocation bits in a bitmap is an incredibly expensive thing to do.

To change the policy for an external storage device...

- Open Disk Management
- Right-Click on Properties
- Select the Policy tab and explore the text you'll find there



Miscellany

- Brad Silverberg and Right Click "Copy"... Rather than Drag & Drop
- Shift-Right-Click: Adds "Copy as Path"

Closing The Loop

Matt D in Philadelphia

Subject: Winrar fix via AV software?

Date: 05 Apr 2019 08:46:45

:

Steve, it occurred to me today that it should be possible for AV software to close the ACE loophole in WinRAR if they scan for, flag, and quarantine the offending DLL within the WinRAR software directory. WinRAR would likely only notice when someone attempted to open an ACE archive, and worst case it forces people to go reinstall a new (patched) version of the software. Am I missing something here? AV software always auto-updates, and this could be easily pushed out. Perhaps they refuse to do this for policy reasons? It seems to me this should be a big enough zero-day that they could make an exception.

Bryan <anon@grc.com>

Location: DeWitt, MI

Subject: Where do random numbers go on vacation in the winter

Date: 04 Apr 2019 10:28:20

:

...to the entropycs (pronounce as en-tropics).

SpinRite

Windows 10, version 1809 and later:

Change in default removal policy for external storage media

<https://support.microsoft.com/en-us/help/4495263/windows-10-1809-change-in-default-removal-policy-for-external-media>

Summary

Windows defines two policies, Quick removal and Better performance, that control how the system interacts with external storage devices such as USB thumb drives or Thunderbolt-enabled external drives. Beginning in Windows 10 version 1809, the default policy is Quick removal.

In earlier versions of Windows the default policy was Better performance.

You can change the policy setting for each external device, and the policy that you set remains in effect if you disconnect the device and then connect it again to the same computer port.

Translation: Historically, Windows has always enabled Write caching for all mass storage, internal and external.

This is especially crucial for external non-volatile thumb drives since (a) they write so very slowly and (b) writing inherently degrades their storage cells and shortens their lives.

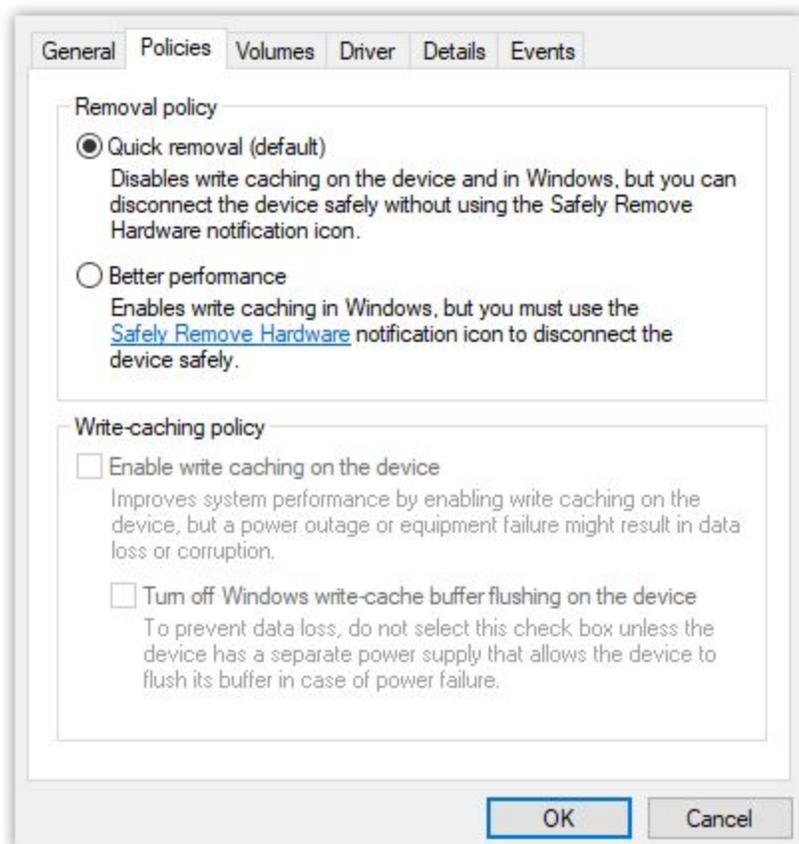
However, users were not reliably using the "Safe Eject" function for their external USB Thumb drives, and they were pulling the drives out of the machine before the drive's writing cached data had been fully flushed and written to the drive.

So, after lo these many years, we've decided to err in the direction of caution and to protect users from themselves by disabling USB device write caching by default, thus protecting users from themselves at the cost of dramatically slowing down all USB write operations -- and prematurely aging write-intolerant thumb drives.

Due to multiple writes frequently occurring to the same block, disabling all external drive caching doesn't just push the timing out to the end, it dramatically slows down the entire process when writing to the drive -- and it also shortens the drive's life -- since disabling caching prevents the coalescing of multiple writes to the same block -- which is a crucially important function of write caching. And multiple writes to the same block happens a LOT within the file system allocation tables where allocation bitmaps are maintained. Flipping individual allocation bits in a bitmap is an incredibly expensive thing to do.

To change the policy for an external storage device...

- Open Disk Management
- Right-Click on Properties
- Select the Policy tab and explore the text you'll find there




```
▼ <a href="https://www.imdb.com/title/tt1326954/" ping="/url?sa=t&source=web&rct=j&url=https://www.imdb.com/title/tt1326954/&ved=2ahUKEwiMieOBg8LhAhXSIjQIHQRWAlYQFjAXegQIBRAB">
  <h3 class="LC201b">Oh My God (2009) - IMDb</h3>
  <br>
  ▶ <div class="TbwUpd">...</div>
</a>
```

Why the difference? Because Google knows that Chrome supports native URL follow "ping" tracking so that it doesn't need to bother with the time consuming URL redirect dance. Google generates very different results pages when it's rendering for Chrome vs another browser where it cannot rely upon "ping" tracking being present.

What's "ping" tracking? It's part of what has become the HTML5 standard which has been around for at least 11 years. It never really took hold because it's so clearly designed for tracking. Unlike URL redirection (which is arguably being abused in the way similar to 3rd-party cookies), where URL redirection has many valid non-tracking purposes, "ping" tracking has no other purpose other than tracking. So for many years the purists doing the web engine development held out and kept "ping" tracking marginalized and not something that could be relied upon as being present.

But today, as with so many other skirmishes, that war is being lost. In today's Chrome 73, it is enabled by default but there is an option flag to disable it. But that option flag has been removed from 74 and 75. Apple's Safari has is stuck on, as does Opera. The Firefox and Brave browsers, as the holdouts, still have it disabled by default, but Firefox does have the option to enable it, since it is, after all, part of the official HTML5 specification.

So what is it, exactly?

The venerable <a> anchor tag was here from the start and it has accreted a large number of optional terms over the years. One of the more recent additions is the "ping" option. The "ping" option is defined as the term "ping" followed by an equals sign and a quote-enclosed list of space-separated URLs.

As per HTML, anything a browser doesn't recognize or support it should ignore. So this won't have any effect in a browser that doesn't understand "ping" or has it disabled. But when a URL containing a "ping" term is clicked, the browser does several things at once:

It immediately jumps to the primary target of the <a> HREF URL =and= it simultaneously (asynchronously) sends one or more POST queries to the list of URLs enumerated in the "ping" URL list.

The POST contains the single word "ping" in its body, but its query parameters provide "Ping-From:", "Ping-To:" and the Content-Type: text/ping. In other words, it's the perfect tracker. And since the browser is making a query to the ping-target site, it will doubtless include

any cookies that the user's browser might already be carrying for that website domain, thus explicitly enabling cross-site and trans-Internet tracking.

What I found most interesting and a bit disturbing was that there is no "same origin" policy control over the ping destinations. The browser is not constrained to only "ping back" to the page's origin domain. The page can instruct the browser to "ping" anyone, anywhere that the page's code requests. At first blush that might seem irresponsible. But I'm sure those arguing for this flexibility noted that URL redirects can also jump to anywhere without restriction, so why artificially restrict the built-in facility that's being offered up as its cleaner -- if a bit creepier -- replacement?

The two things this has going for it is that it's cleaner and clearer and more direct. But also that it does IN PARALLEL what the URL redirect chain does in SERIES. So there's some chance that the user will appear to obtain a snappier response from the site they wish to jump to because the "ping" news about their jump is being sent out at the same time, asynchronously, thus not delaying their arrival at the new site by being bounced around among various tracking sites first.

If anyone has ever noticed their browser's URL address field when this gets really ridiculous? It's be dancing around like mad a bit before you finally get to where you just wanted to go. That was a long URL redirection chain, and they can take significant time since only one server in the chain needs to be slow or down to impede or break the chain.

I don't know when Firefox and Brave will finally capitulate. I think that they might as well do so since the writing is pretty clearly on the wall. URL redirection chains work to solve the same problem, and they cannot be blocked without breaking lots of other useful things. I don't like it either, but like it or not, the web of the 21st century is about tracking and data collection. It was once practical to block website JavaScript by default and selectively enable it where really necessary. That's just not practical on today's web. So I understand, respect and even salute those who want to, and are willing to, continue fighting for their belief that they should not be tracking without their explicit consent. But I think it's becoming increasingly difficult to successfully wage that battle.

~30~