Security Now! #1045 - 09-30-25

News and Listener Views

This week on Security Now!

• Gmail's spam filtering false-positive spree. • iOS 26's Safari randomizes its fingerprint by default. • Cisco's SNMP stands for "Security Not My Problem". • Windows' "stuck" Extended Security Updates (ESU). • Europe complains, gets 1-year of ESU with no strings. • Where to get $6 TLS certs (really) while they last. • The lessons to learn from Jaguar Land Rover's mess. • The NEON app: get paid to have your voice recorded. • Bluesky's age verification, now coming to Ohio. • What is "Kids Web Services" for age verification. • More than 10K Ollama instances publicly exposed. • GRC's DNS Benchmark reaches "release candidate".

Guilty, as charged…

Gmail's spam filter changes

Donn Edwards

Dear Steve, Last weekend I noticed that Gmail's spam filter rules had changed dramatically.

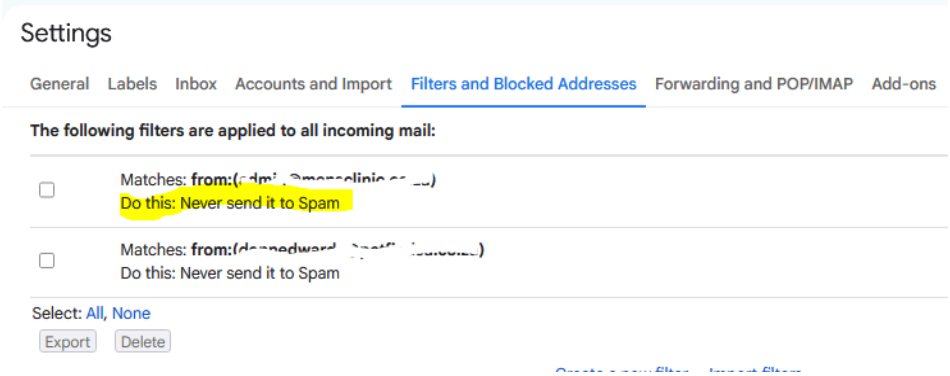
I send a short, software generated, email message to a Gmail address every 5 minutes, to test the operation of my mail server. It consists of two lines, with no HTML whatsoever:

> *eMailtimer sent on 19-SEP-2025 at 09:40:00*
> *Automatic Version 0.8.0.928*

The subject line is equally innocuous:

> *[82086,5790027] 19-SEP-2025 at 9:40AM eMailtimer*

On Saturday 20 September Gmail suddenly decided to treat ALL these messages as Spam. I eventually added a filter to Gmail to never send messages from this address to spam, since the "Not Spam" button had no effect whatsoever:



Perhaps you could advise Gmail users to do this for securitynow@grc.com just in case.
Keep up the good work and thank you for so many years of great content!
Best wishes, Donn Edwards, Johannesburg, South Africa

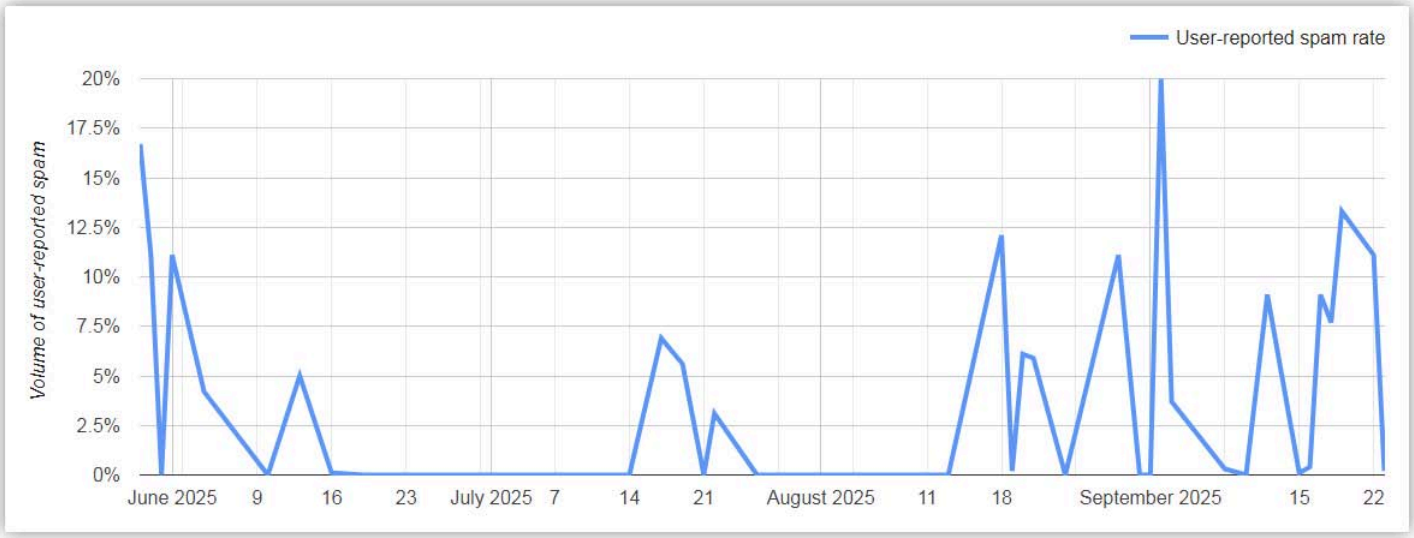
Donn's note was extremely interesting but also somewhat disheartening. An overwhelming percentage of GRC's email system subscribers are using gmail addresses, either directly with the gmail.com domain, or with gmail handling their own domain. We've been sending email like this every week for the past year without any trouble, with no complaints and without our email ever being mis-routed into Spam. What's disheartening is that apparently that long track record doesn't count for anything.

The only thing I would change in Donn's note is to mention that eMail from GRC does not come from the "securitynow@grc.com" address. If you look, you'll see that email is sent from the address "mail-manager@grc.com". So that would be the address you'll need to use.

It's also disheartening that Donn reports that the use of the "Not Spam" button is appearing to fail to train the system that this is not spam. I don't know whether having everyone adding a filter with the rule action of "Never send it to Spam" might also serve to train gmail, but I'll hope that this week we had better results.

As my attention was brought back to this again – I haven't needed to think about it since late

last year when I set everything up and did the mailing to all of SpinRite's past purchasers, I remembered that there were Google "Postmaster" tools for monitoring what Google thinks about email incoming from GRC. So I went back to pull up a chart of the past 120 days:



In the chart above, every light vertical line corresponds to a Monday of the week. Remember that for more than the past four months which this chart shows, I've been dumping around eighteen thousand pieces of email once per week, every week, every Monday.

What we notice, looking at this chart, is no correlation between spam reports and GRC's weekly Monday mailings whose volume is 19 thousand subscribers. The conclusion is that email from GRC is being spoofed and GRC email is being impersonated. Spammers ARE sending email to people as if it was from GRC. And in this instance, GRC is suffering from the fact that we send so little email that even a small bit of spam by count equals a large amount by percentage, which appears to be the only thing that Google and others track.

The puzzle is that I have always had GRC's email being DKIM signed and SPF sender validated. And GRC's DMARC policy has always been set to reject fraudulent email. This means there's no way Google Mail should have been allowing even a single piece of spam to reach its users.

EVERY single piece of email GRC originates is cryptographically signed with a private key whose matching public key is published by our DNS, along with the strictest of enforcement policies.

However, after doing some further research I believe that I was not using the strictest of all possible enforcement policies. I am now. I learned that "alignment" is the term for header domain matching, and that if it's not explicitly specified as "strict" then a "relaxed" policy is assumed. So, I've now tightened this further and I've enabled DMARC success and failure reporting. Preliminary reports appear to show that a change has occurred.

One last bit of news is that I also received additional scattered anecdotal reports from other podcast listeners who reported suddenly observing unexpected Gmail spam false-positives. And over the weekend I sent three small test mailings to a list of 71 GRC watchers who previously indicated an interest in participating in testing email stuff. That list uses all of the same mechanics as GRC's weekly Security Now! Podcast mailing, identifying itself as originating from a list with unsubscribe headers and so forth. My own gmail email account is on that list and it received all three of those test mailings... so I suspect that Google's Gmail folks may have tripped over something that caused a large number of false positive spam hits.

Security News

Safari for iOS 26 is randomizing its users' browser fingerprints

Mac Observer's headline was: *"Apple is Turning on a Powerful Safari Anti-Tracking Tool for Everyone"* Their coverage opened with the sentence: *"Apple is widening Safari's privacy shield. Starting with iOS 26 and macOS Tahoe 26, advanced fingerprinting protection is enabled by default in every tab, not just in Private Browsing."* We know that Mozilla also recently claimed to have improved their fingerprint protection. That caused me to check my updated Firefox with the EFF's excellent "Cover Your Tracks" (https://coveryourtracks.eff.org/) website which, sadly, informed me that my Firefox browser still had a unique fingerprint.

Upon seeing this iOS 26 claim, and having recently been disappointed by Mozilla, I headed back to "coveryourtracks.eff.org" with an iOS 26 iPhone. I purchased an iPhone 16 earlier this year due to that initial panic over tariffs on Chinese imports. But I'm not letting that newer phone get anywhere near iOS 26 until Apple gets a clue and allows all of that Liquid Glass nonsense to be completely turned off, not just "turned down". But I still have my older iPhone 12 which I did update to iOS 26 since it's too old to handle the overhead of Liquid Glass and I do like the many other things that iOS 26 brings. And, sure enough! Anyone using iOS 26 has their tracks fully covered:

I changed nothing about my default browser settings and CoverYourTracks reports that I am blocking tracking ads, blocking invisible trackers, and that my browser has a randomized fingerprint. As we know, the fact that I never changed any Safari settings is crucial. Mac Observer's reporting indicates that these anti-tracking measures are enabled by default, which means that all iOS 26 users, whether or not they are aware of their existence, will obtain this protection. They said:

> *Fingerprinting uses subtle device and browser traits to identify you across sites. Safari now standardizes and "noises" more of those signals by default, making it harder for trackers to single you out while you browse normally.*
>
> *Safari reduces access to high-entropy web APIs commonly abused for fingerprinting and limits script-written storage and navigational state reads. Practically, that translates into fewer stable identifiers for tracking scripts and less durable "stickiness" across sessions.*
>
> *This doesn't change how Link Tracking Protection works in Mail, Messages, and Private Browsing. It also doesn't remove normal cookies from sites you sign into. If something breaks on a niche site, you can temporarily relax protections and try again.*

To check or change settings:

On iPhone or iPad, go to Settings → Safari → Advanced → Advanced Tracking & Fingerprinting Protection and confirm it's set to "All Browsing". It might be set to "Off" or "Private Browsing".

On Mac, open Safari and go to Settings → Advanced and set "Use advanced tracking and fingerprinting protection" to "In all browsing".

My iPhone 16 is still running the last iOS before this jump to iOS 26 – it has 18.6.2. So I decided to see how it compared. I had also never messed with Safari's settings there, so its "Advanced Tracking & Fingerprinting" was still set to "Private Browsing". I changed it to "All browsing" and went to the EFF's "CoverYourTracks" site. Under iOS 18.6.2, I'm getting blocking of tracking ads and invisible trackers, but I'm told that Safari under the last iOS before the move to iOS 26 is presenting a non-unique fingerprint to the world.

It's still way better than the latest Firefox. CYT says that my latest Firefox browser is unique among every one of the 301,784 other browsers that have visited this site during the past 45 days. Wonderful. No problem following me around the Internet. CYT adds that this means it's providing at least 18.2 bits of entropy.

By comparison, among those same 301,784 browsers that CYT encountered during the past 45 days, iOS 18's Safari shared a fingerprint with fewer than one out of every 2340 other browsers. So everyone using any version of Safari on iOS or Mac before 26 would be well served to change their browser's anti-tracking and blocking to "All Browsing" because why not? And anyone who has made the move to iOS of Mac 26 not only doesn't need to change anything, but as is, they're generating an inherently anti-tracking ephemeral fingerprint.

Cisco SNMP: "Security? Not My Problem"

I was reminded of the old joke that "SNMP" stood for "Security's Not My Problem". This is a joke because of course it is. SNMP actually stands for "Simple Network Management Protocol". Any-time the name of a widely used, ancient Internet protocol begins with the word "Simple" you can bet that the "S" would never be confused with standing for "Security." And that's nowhere more true than for SNMP.

The original RFC's which specify the operation of SNMP date from 1988 and, believe it or not, it was never intended to be used for the long term. It was created as a quick hack stopgap solution because the work other groups were doing was believed to be far too burdensome and unimplementable on the machines of the time. So SNMP was thrown together quickly without any real security.

Like DNS which was designed to be lightweight and low-overhead, SNMP uses UDP packets and also like DNS over UDP, SNMP has **no** encryption or eavesdropping protection whatsoever. And what security there is, takes the form of a simple cleartext password whose textual string must match the one stored in the SNMP-equipped device being queried. There's not even any hashing for the user's password. Again, remember, this was back in 1988 when everyone was still getting over their shock that autonomous packet routing actually worked and could be used and scaled to build robust networks.

The trouble is, SNMP is also incredibly useful. It's the way my little SoftPerfect NetWorx network monitor is able to watch the SNMP counters on my network's LAN interface to show the entire network's bandwidth use. That little SNMP client running on my Windows machine is sending UDP packets to port 161 of the LAN interface and the router's little SNMP server is examining the packets, seeing that I'm requesting the interface's received and sent byte counts, which it then

sends back to the client in a return UDP packet. SNMP presents a tree-like structure which can be explored by any patient client to discover all of any network device's settings. This includes things like all of its interfaces, its routing and ARP tables, the IP addresses and network masks of its interfaces, and so on. And SNMP is not just for querying devices. It also supports the setting and changing of a device's settings – all remotely and over the network. And, remember, using insecure plaintext UDP. It's always been a security disaster just waiting to happen.

Here's what Wikipedia has to say under the heading of SNMP Security:

> *Because SNMP is designed to allow administrators to monitor and configure network devices remotely, it can also be used to penetrate a network. A significant number of software tools can scan the entire network using SNMP; therefore, mistakes in the configuration of the read-write mode can make a network susceptible to attack.*
>
> *In 2001, Cisco released information that indicated that, even in read-only mode, the SNMP implementation of Cisco IOS is vulnerable to certain denial of service attacks. These security issues can be fixed through an IOS upgrade.*
>
> *If SNMP is not used in a network, it should be disabled in network devices. When configuring SNMP read-only mode, close attention should be paid to the configuration of the access control and from which IP addresses SNMP messages are accepted. If the SNMP servers are identified by their IP addresses, SNMP is only allowed to respond to these IPs and SNMP messages from other IP addresses would be denied. However, IP address spoofing remains a security concern.*

In other words, SNMP is and has always been a security disaster. If SNMP is not actively needed and in use, its service should never be running. And the IP addresses of anything using SNMP should be filtered so that only authorized clients can obtain SNMP services. Unfortunately, as Wikipedia points out, being carried by UDP spoofing of source IPs remains a problem.

All of the foregoing would mean that no network engineer would ever consider placing any system that published SNMP onto any public network. And this, of course, brings us to ArsTechnica's headline last Thursday: *"As many as 2 million Cisco devices affected by actively exploited 0-day"* and their sub-head: *"Search shows 2 million vulnerable Cisco SNMP interfaces exposed to the Internet."* The actual number, revealed by a Shodan search for the strong "CiscoSystem" on port 161, appears to be 2,303,370.

At one point in their reporting Ars tells us:

> *The vulnerability is the result of a stack overflow bug in the IOS component that handles SNMP (simple network management protocol), which routers and other devices use to collect and handle information about devices inside a network. The vulnerability is exploited by sending crafted SNMP packets.*

Note the phrase "inside a network". SNMP has no business being enabled on the public-facing interface of **any** network equipment. Despite that fact, 2,303,370 Cisco devices respond to a Shodan client query of their port 161 with the string *"CiscoSystems"*. I would be very surprised to learn that more than 2.3 million individual network engineers could or would have deliberately enabled the SNMP service – which they almost certainly have no need for anywhere – on their Cisco device's Internet-connected interface.

Unfortunately, this is readily explainable as yet another example of incredible hubris on Cisco's part. I'm sure that, when asked, they would reply as they have before that their equipment assumes its use by qualified and trained network engineers and that the guidelines enumerated in their "How to harden the security of your new Cisco device" guide should always be followed – even though, of course, it's not required.

The reason Cisco's IOS has such a legacy of trouble is that its design philosophy was born back in 1984 with the company's founding by a pair of Stanford University computer scientists. Little thought was given to "security" back at the dawn of the Internet. And the design of Cisco's IOS – IOS stands for Internet Operating System – reflects that in its implicit assumption that IOS would only be configured and used by professional network engineers. For example, even today, any internal services, such as SNMP or HTTP, that run in the device are, by default, available to all of the device's interfaces. A Cisco IOS device has no intrinsic notion of LANs and WANs. Those are just interfaces and Cisco devices all just have network ports.

The earlier devices, before the early 2000's actually had all of their various services running by default **and** available on all of the device's interfaces until and unless the service was explicitly shutdown with a configuration command such as "no ip http server". GRC originally used Cisco devices. This podcast's early followers will recall the time I had a pair of 1.54 gigabit T1 trunk lines running to my home. They were terminated and bonded by a Cisco router and I recall quite clearly needing to explicitly place a "no ip http server" line in the router's configuration in order to prevent that unwanted service from running and being present on all of the device's interfaces.

So, thanks to Cisco's long legacy of apparently assuming that only highly-trained network engineers would be configuring their devices, today in 2025, more than 2.3 **million** of Cisco's IOS-powered routers are exposing their SNMP services to the public Internet. And, unfortunately, the world has learned that until they're patched – and what chance is there of ***that*** ever happening? – the stack overflow that's present inside all of those routers' SNMP packet processing is exposing those devices to an actively exploited 0-day attack.

The takeaway for our listeners is to be very certain that if you're responsible for a network containing any publicly-facing Cisco IOS-based routers, that, first of all, they are running the latest version of IOS available for them and then that they are only exposing services that are truly needed – not only to the public but also to internal networks. And while you're at it, if the remote IPs that might be using specific services are known and fixed, take the time to filter access to those services so that only those known clients will have access. Get ahead of the next discovered vulnerability.

Enabling Extended Security Updates (ESU)

And speaking of vulnerabilities, a great deal of uncertainty currently surrounds Microsoft's plans for Windows 10 and the availability of the next year's worth of security updates courtesy of their ESU – Extended Security Updates – program.

The world is asking that Microsoft give it more time to migrate from Windows 10 to 11. It would be ideal if Microsoft were to allow Windows 10 machines to be upgraded to run Windows 11. That would beautifully solve their OS version fragmentation problem. They could cook up some story about having made a breakthrough in Windows 11 that now allows it to miraculously run on all of that older hardware that's currently running Windows 10.

But if that cannot or will not happen, I thought that Stacy made a very strong point in her open letter to Microsoft. She noted that someone might have purchased a brand new PC containing

Windows 10 as recently as 2022 whose hardware Microsoft has deemed unworthy of running Windows 11. So now they have a 3-year-old PC that's about to lose access to its constant I.V. drip of security updates, many of which are critical and all of which are of Microsoft's on making. As Stacy wrote eloquently, that doesn't seem right and it also breaks with the expectation Microsoft has previously set of a 10-year life for systems running Windows. I bring this up again as we approach the middle of next month because it appears to be possible for existing Windows 10 non-enterprise edition end users to push their machines to offer the enrollment for the next year of free ESU.

I had a Windows 10 instance that was current with all of its updates and it was logged into my Microsoft account... but it had not yet received Microsoft's offer to enroll in the forthcoming year of ESU. It wasn't clear what it was waiting for or when that might happen. We've heard that the ESU enrollment invitation is being handled as a gradual roll-out, so it might be that just waiting would be sufficient. But some Windows explorers have come up with a sequence of steps that appear to hurry the process along and I wanted to experiment with that. The short version is: It worked, and that machine is now enrolled for the next year of updates.

If you're still running Windows 10 you could wait to see if it happens automatically for you, or you could probably follow the steps I took to get enrolled without waiting. The instructions were posted into a forum thread at the AskWoody website. The AskWoody thread is titled *"How to extend Windows 10 Support now published by ghacks"* and it was very active. The thread refers to ghacks.net where, back on July 24th, Martin Brinkmann posted a series of ESU enrollment screen shots. The only trouble is, many people are not receiving the offer in the first place.

Well down that long thread, an AskWoody_MVP with the handle *"abbodi86"* posted some step-by-step instructions under the title: "Here is a way to force enable "Consumer ESU Feature" and it worked for me. It successfully induced a Windows 10 machine to invite me to enroll.

I have the link to that posting in the AskWoody forum thread in the show notes: https://www.askwoody.com/forums/topic/how-to-extend-windows-10-support-now-published-by-ghacks/#post-2796782 But there's more. This same "abbodi" individual has created a very nice and comprehensive set of resources at GitHub which includes powershell scripts, ample background material and also the same manual steps that I used. The link is in the show notes, and I made it this week's GRC shortcut of the week, so you can just enter: grc.sc/1045. That will transport you to "abbodi"s GitHub page: https://github.com/abbodi1406/ConsumerESU. Armed with the information there you should be able to get any qualifying consumer Windows 10 machine enrolled and ready to continue receiving the next year of updates without interruption.

Windows ESU in the Europe without any strings attached

Another bit of news landed late last week: The news out of what's known as the European Economic Area (EEA) which includes users living in EU member states plus Iceland, Liechtenstein, and Norway is that the Windows 10 machines of those users will continue receiving Windows updates – presumably under the extended service updates program, unless Microsoft finally just scraps the whole thing – with them having to do anything whatsoever. No PC backup, no Bing Brownie points, no PC settings synchronization. No nothing.

Windows 10 updates will be truly and completely free with no strings attached due to pressure from "Euroconsumers", a major consumer protection NGO, which questioned the legality of Microsoft's offerings compared to the EU's new Digital Markets Act (DMA). And the best news is that the organization is not finished pushing Microsoft. They are now demanding that Microsoft provide additional years of ESUs to home users beyond next year, October 13, 2026.

Cheap SSL
I wanted to share another recent discovery of mine: $6 TLS certificates issued with the current maximum allowed 13-month lifetime.

Everyone knows about the revoked.grc.com website I created many years ago to demonstrate that the web browsers of that era were completely ineffective in their checks for certificate revocation. To demonstrate that, the "revoked.grc.com" server there is serving a deliberately revoked certificate. The trouble was, last year's deliberately revoked certificate was expiring and needed to be replaced. And since that simple facility has become pretty popular, receiving around 500 visits per day, it made sense to keep it alive.

It's possible to automate both the issuance **and** the revocation of certificates using the ACME protocol. So I imagine I'll cook up some way of always using a freshly ACME-revoked certificate for the revoked.grc.com site once I've been forced to move all of GRC's other TLS certificates over to Let's Encrypt due to the industry's inexorable and unnecessary march shorter certificate lifetimes. But not today. Today I want to get the new and much improved DNS Benchmark wrapped up. So I just wanted to do what I've been doing, which is to purchase and immediately revoke a single certificate for that site.

For many years, DigiCert has been my certificate supplier. So I went there first. They no longer offer the least expensive and least verified DV-style Domain Validation certificates. I suppose they wisely decided not to attempt to compete with Let's Encrypt's free ACME automation certs. But that leaves OV – Organization Validation – certificates as DigiCert's least expensive option where their price for a single one-year OV certificate is now a breathtaking $324.

Paying $324 for a certificate that would never even be valid, would be pouring money down the drain. So I went looking around for any widely-trusted certificate which I could purchase for a reasonable price. The search brought me to https://cheapsslweb.com/ – and cheap they are. $12 paid one time gives you the right to issue certificates for the domain of your choice as often as you like for two years. So that works out to $6 per domain-year. They offer a 5-year purchase for $20, but the near-term schedule for maximum certificate life shortening means that two to three years is likely to spell the end of manual certificate issuance and management.

Until March 15th of next year, certificates are allowed to have a 398-day life. But the CAB forum requires that to be cut in half, to just 200 days, next March 15th, then again to 100 days a year later, and finally down to 47 days in 2029, four years from now. So, depending upon how resistant you or your application may be to automation, which will pretty much be required after March 15th of 2029 when certificates will only be allowed to have 47-day life, you might want to just do 2 years for $12. Under that plan, you'd purchase and begin using a certificate now. Then you'd issue another certificate shortly before next March 15th 2026, when you can still do so for another 398 days. Then issue a 200-day certificate a year later, shortly before March 15, 2027 and your last 100-day certificate within 200 days, or before your two year purchase subscription lapses with CheapSslWeb. And somewhere before then you should plan to have automation in place when that final certificate expires. And using https://cheapsslweb.com that can all be done within their two-year $12 purchase.

Anyway, if you are not yet ready to invest the time and effort to move to Let's Encrypt's ACME automation for whatever reason – as I am not – I wanted to let everyone know that the downward pricing pressure that Let's Encrypt has placed on the traditional DV certificate market has resulted in extremely inexpensive and widely trusted manually issued and installed domain validation certificates, **while they last.** It won't be feasible to do manual issuance after March of 2029. But certificates issued just before the various deadlines will be able to live long enough for manual management to still be feasible.

Jaguar Land Rover
I wanted to follow up a bit on the status of Jaguar Land Rover since it's quite a harrowing story. I'll just quickly quote from the reporting I saw in the Risky Business Newsletter, which wrote:

> *The UK government has agreed to underwrite a £1.5 billion pound loan to Jaguar Land Rover to help the carmaker deal with the increasingly costly aftermath of a recent cyberattack that has crippled its production and shut down factories for almost a month.*
>
> *The underwrite was approved on Sunday after a visit from UK Business Secretary Peter Kyle to the headquarters of JLR and its main supply chain firm Webasto this week. JLR fell victim to a ransomware attack—supposedly from the HellCat group—on August 31. Production lines at all JLE factories have been shut down ever since, and are expected to last into October.*
>
> *While it looked like another largely benign ransomware attack that hits the back office and the company then needs to reinstall some accounting systems to get back into order, this was not it. Not at all. Systems like CAD , engineering software, and product life-cycle software, payments tracking, and customer car delivery systems went down. The works.*
>
> *The incident has turned into a legitimate catastrophe for both the company, its suppliers, and even the British economy as a whole.*
>
> *With production lines ground to a halt, hundreds of small companies that supplied Jaguar parts and various services also had to slow their pace and even put workers on leave. Several of the smaller ones are facing bankruptcy proceedings and were expected to go under, since several had just "days of cash" left in their accounts.*
>
> *With no car sales and no secondary economic activity being generated by its supply chain for an entire month, the JLR cyberattack is likely to impact the UK's economic growth. The company alone employs over 34,000 people, with another 120,000 working throughout its supply chain.*
>
> *According to a recent report, the company also did not have a cyber insurance deal at the time of the attack and will likely have to foot the bill for the attack and subsequent revenue losses.*
>
> *Just JLR by itself is expected to lose "hundreds of millions of pounds," according to reports, which explains the need for an underwrite in the realm of £1.5 billion.*

We're not on the inside, so we cannot definitively say why and how this happened. But the fact that the company was not carrying any insurance against cyberattacks, and that whatever happened was able to so deeply and so thoroughly nuke its operational capabilities, all that at least strongly suggests that the management of Jaguar Land Rover was not taking the reality of today's cyberattacks seriously enough. They may have felt that carrying cyberattack insurance would be prohibitively expensive. But one has to wonder how they might feel about that decision today?

We've also seen cyber insurance companies becoming increasingly involved in the organization they're being asked to insure – setting operational requirements to minimize everyone's risk. So it may have been the case that Jaguar Land Rover's observable cyber-attack readiness was so poor – as now appears to be the case – that any prospective insurers were forced to quote ridiculously high premiums while capping their liability to a point that carrying insurance under those terms didn't make sense.

The big takeaway lesson for all C-suite executives – and I sincerely hope that this startling Jaguar Land Rover news reaches them – is that the maintenance of true cyber attack readiness is no longer something that can be given lip-service then dismissed when the time comes to set budgets. The unfortunate reality is that today's cyber-threat landscape has truly and significantly increased the cost of doing business. This means that, one way or another, today's enterprises are going to pay – either in advance for preemptive protection and cyber-insurance, or in the form of post-attack ransoms, possibly serious downtime and reputational harm.

NEON

A headline in TechCrunch would give anyone pause. Last Wednesday, they posted a story. Listen carefully to this headline: "Neon, the No. 2 social app on the Apple App Store, **pays** users to record their phone calls and sells data to AI firms. What?! Here's what TechCrunch reported:

*A new app offering to record your phone calls **and pay you for the audio** so it can sell the [audio] data to AI companies is, unbelievably, the No. 2 app in Apple's U.S. App Store's Social Networking section.*

*The app, Neon Mobile, pitches itself as a moneymaking tool offering "hundreds or even thousands of dollars per year" for access to **your** audio conversations. Neon's website says the company pays 30¢ per minute when you call other Neon users and up to $30 per day maximum for making calls to anyone else. The app also pays for referrals. The app first ranked No. 476 in the Social Networking category of the U.S. App Store on September 18 but jumped to No. 10 by the end of last Monday, according to data from app intelligence firm Appfigures.*

Last Wednesday, Neon was spotted in the No. 2 position on the iPhone's top free charts for social apps. It also became the No. 7 top overall app or game earlier on Wednesday morning and became the No. 6 top app.

According to Neon's terms of service, the company's mobile app can capture users' inbound and outbound phone calls. However, Neon's marketing claims to only record your side of the call unless it's with another Neon user. That data is being sold to "AI companies," Neon's terms of service state, "for the purpose of developing, training, testing, and improving machine learning models, artificial intelligence tools and systems, and related technologies."

The fact that such an app exists and is permitted on the app stores is an indication of how far AI has encroached into users' lives and areas once thought of as private. Its high ranking within the Apple App Store, meanwhile, is proof that there is now some subsection of the market seemingly willing to exchange their privacy for pennies, regardless of the larger cost to themselves or society.

Despite what Neon's privacy policy says, its terms include a very broad license to its user data, where Neon grants itself a:

> *…worldwide, exclusive, irrevocable, transferable, royalty-free, fully paid right and license (with the right to sublicense through multiple tiers) to sell, use, host, store, transfer, publicly display, publicly perform (including by means of a digital audio transmission), communicate to the public, reproduce, modify for the purpose of formatting for display, create derivative works as authorized in these Terms, and distribute your Recordings, in whole or in part, in any media formats and through any media channels, in each instance whether now known or hereafter developed.*

That leaves plenty of wiggle room for Neon to do more with users' data than it claims. The terms also include an extensive section on beta features, which have no warranty and may have all sorts of issues and bugs. Though Neon's app raises many red flags, it may be technically legal.

Jennifer Daniels, a partner with the law firm Blank Rome's Privacy, Security & Data Protection Group, tells TechCrunch: "Recording only one side of the phone call is aimed at avoiding wiretap laws. Under [the] laws of many states, you must obtain consent from both parties to a conversation in order to record it … It's an interesting approach." says Daniels.

Peter Jackson, cybersecurity and privacy attorney at Greenberg Glusker, agreed — and tells TechCrunch that the language around "one-sided transcripts" sounds like it could be a backdoor way of saying that Neon records users' calls in their entirety but may just remove what the other party said from the final transcript.

In addition, the legal experts pointed to concerns about how anonymized the data may really be. Neon claims it removes users' names, emails, and phone numbers before selling data to AI companies. But the company doesn't say how AI partners or others it sells to could use that data. Voice data could be used to make fake calls that sound like they're coming from you, or AI companies could use your voice to make their own AI voices.

Jackson said: "Once your voice is over there, it can be used for fraud. Now this company has your phone number and essentially enough information — they have recordings of your voice, which could be used to create an impersonation of you and do all sorts of fraud."

Even if the company itself is trustworthy, Neon doesn't disclose who its trusted partners are or what those entities are allowed to do with users' data further down the road. Neon is also subject to potential data breaches, as any company with valuable data may be.

In a brief test by TechCrunch, Neon did not offer any indication that it was recording the user's call, nor did it warn the call recipient. The app worked like any other voice-over-IP app, and the caller ID displayed the inbound phone number, as usual. (We'll leave it to security researchers to attempt to verify the app's other claims.)

Neon founder Alex Kiam didn't return a request for comment. A business filing shows that Kiam, who is identified only as "Alex" on the company website, operates Neon from an apartment in New York. A LinkedIn post indicates Kiam raised money from Upfront Ventures a few months ago for his startup, but the investor didn't respond to an inquiry from TechCrunch as of the time of writing.

Has AI desensitized users to privacy concerns?

There was a time when companies looking to profit from data collection through mobile apps handled this type of thing on the sly.

When it was revealed in 2019 that Facebook was paying teens to install an app that spies on them, it was a scandal. The following year, headlines buzzed again when it was discovered that app store analytics providers operated dozens of seemingly innocuous apps to collect usage data about the mobile app ecosystem. There are regular warnings to be wary of VPN apps, which often aren't as private as they claim. There are even government reports detailing how agencies regularly purchase personal data that's "commercially available" on the market. Now AI agents regularly join meetings to take notes, and always-on AI devices are on the

market. But at least in those cases, everyone is consenting to a recording, Daniels tells TechCrunch.

In light of this widespread usage and sale of personal data, there are likely now those cynical enough to think that if their data is being sold anyway, they may as well profit from it. Unfortunately, they may be sharing more information than they realize and putting others' privacy at risk when they do.

Jackson said: "There is a tremendous desire on the part of, certainly, knowledge workers — and frankly, everybody — to make it as easy as possible to do your job. And some of these productivity tools do that at the expense of, obviously, your privacy, but also, increasingly, the privacy of those with whom you are interacting on a day-to-day basis."

I have several reactions to this.

Although I know I'm not talking about this audience, many people really don't care about their privacy all that much. I've received sufficient feedback through the years from the people who take the time to listen to this podcast to know that the great majority of our listeners would have no problem being characterized as "old school" as regards their privacy and concerns for online security. But we're the extreme cases. The huge majority of people really do not care.

So I'm not surprised to learn that an app that promises to pay up to $30 per day in return for having one's voice recorded and sold is succeeding. It's not difficult to imagine this spreading like wildfire across school campuses. And the bonus of increasing the payout if both parties are using the service is such a clever way of getting the system to go viral.

What surprises me is the size of the payout amount, which seems quite high and I would be surprised if it turned out to be sustainable at that level. So there may be some early adopter bait-and-switch going on here, where the payout rate will drop once the system has been widely established.

Also, how are the funds sent back to Neon's users? I downloaded the app to see what I could learn but there was no option other than signing up giving it my phone number which, being something of an avid listener of this podcast, I was unwilling to do. So I got no further and that remains an open question for me. How are users paid?

The last thought I have is that, as I'm sure would be the case for many of our listeners, I would have a big problem with not being informed that my voice was being surreptitiously recorded and sold by whomever I was speaking with on the other end of a conversation. Not only is it just creepy, but voice authentication promises to be a serious problem in the future. When you combine AI's ability to convincingly converse with generative AI's ability to on-the-fly spoof the voices of anyone it has a sufficient sample set for, all the pieces are in place for trouble.

Some time ago I told my office manager and bookkeeper, Sue, that she must verify with me in writing via our internal email anything that she believes I'm instructing her to do regarding moving money. And we've been practicing that for some time. I've told her that I will never tell her not to confirm in writing and that no emergency is too great for verification first.

It might seem extreme and inconvenient, like freezing one's credit reporting, but these sorts of simple measures can make the difference between being a victim or not.

Bluesky's Age Verification in Ohio

I wanted to correct or at least clarify something I said about the decentralized social media service Bluesky and age verification. We know that Bluesky has suspended all of its services in Mississippi due to that state's nutty "only proven adults can access any social media" law. And we recently noted that Bluesky would be doing the same thing in the states of South Dakota and Wyoming as they have for the United Kingdom, where saner laws have been enacted. In the UK and in those two states, only access to adult content requires some form of proof of age.

Then, yesterday, I ran across another update in TechCrunch. The beginning of their report said:

> *The social network Bluesky will begin verifying users' ages in the state of Ohio to comply with new regulations, starting on Monday, September 29. The company — which offers an open and decentralized competitor to X and Threads — says it will enable the Kids Web Services' (KWS) age verification solution in the state. This is the same solution that Bluesky is already using in South Dakota and Wyoming to comply with similar laws.*
>
> *Bluesky announced the move in Ohio on Sunday via its Bluesky Safety account, and in an update to last month's blog post about the matter. The changes come as a number of U.S. states are rolling out their own age verification laws to protect children from online risks, given the lack of federal guidance. The Ohio law, meant to protect kids from pornography, will require users in Ohio to upload a copy of their government-issued photo ID or other personal identification before accessing adult content. This includes the type of adult content that can be found on social networks.*
>
> *KWS will provide the technical infrastructure that allows Bluesky to verify users' ages by offering multiple ways for them to do so beyond only uploading a government-issued identity document. According to its website, KWS also lets users verify by facial scans, payment cards, and more.*

I had previously assumed and at least intimated that Bluesky's use of KWS "Kids Web Services" was some homegrown solution they had cobbled together as a means of verifying age. But TechCrunch's report painted it as an outside service. And, indeed, that's what KWS is: https://www.kidswebservices.com/en-US

I learned is that KWS has been around for quite a while. Exactly four years ago today, on Sept. 30th, 2021, they posted a piece under the headline *"Making the internet (and metaverse) safer for kids with free parent verification for all developers"*, where they explained:

> *Today one of the biggest challenges for developers and content owners is enabling access for young audiences.*

But putting this in a temporal context, remember that was four years ago, long before all of this recent state and federal legislation began blowing up. So what was the need back then? It was access to video game content. The year before they posted this KWS merged with Epic Games.

> *In order to enable features which may require personal data, such as content personalization, navigation, or push notifications, children's privacy laws like COPPA and GDPR-K may require you to obtain the consent of parents and in many cases verify that the parent is an adult. This is called verifiable parental consent.*

Securing verifiable parental consent creates a user experience where a child has to educate their parent, the parent has to go through the registration process, then verify their identity, and only then grant permission for their child. As painful to get through as the sentence is to read! And this has to be repeated every time a child wants to access a new digital experience!

Many developers look at the complexity (and cost) of the parent verification process and choose to simply avoid young audiences entirely. Large developers can afford to build their own solution (or license ours). Small developers don't have that luxury.

Our Kids Web Services (KWS) platform already powers parent verification for some of the biggest games in the world, including Fortnite and Among Us. KWS delivers the most frictionless parent experience in the industry, thanks to its innovative ParentGraph. Once a parent is verified using KWS, they never need to provide their verification details again for any other service using KWS technology, minimizing personal data processing and providing a better user experience for both parents and players. Today, the ParentGraph includes millions of pre-verified parents and is growing rapidly.

While we are heartened to see more technology companies thinking about access and safety for their younger audiences, the future of the internet (and growth of the metaverse) requires kidtech tools to be available to everyone. The ability to execute at this kind of scale is exactly why we joined Epic Games last year.

I was curious how their parental verification worked. Elsewhere, in their FAQ answering *"What type of verification methods does KWS use?"* they explain:

KWS continuously optimizes and adopts new parent verification methods and vendors. Our verification team continuously researches, tests, and adds new methods and vendors to raise the standard of our parent verification service. We offer developers (and parents) methods that are as inclusive and widely accessible as possible while minimizing personal data collection.

Depending on the child's location, our current verification methods include the following:

- *Credit Card/Debit Card Verification*
- *Face scan*
- *Document ID scan*
- *Social Security Number (SSN), CPF or CURP checks*
- *i-PIN (available only in the Republic of Korea)*
- *Cell phone (available only in the Republic of Korea)*

One of the reasons this system was appealing to Bluesky and will likely be appealing to many others is that it is 100% free of charge regardless of the usage volume. Anyone is free to use this established system rather than needing to build their own. For parents, whose kids wished to have access to Epic's games, this meant that they only needed to go through the annoying process of proving they were an adult once. And the site does talk a lot about data minimization and hashing and such.

I didn't spend any time digging into the details of the system's operation because I sincerely hope that in the long term its operation will matter. I hope it's an example of the sort of stop-gap measure that websites and online services will be driven to adopt and use until we obtain the standardization we need.

10,600 instances of Ollama found publicly exposed
The Internet scanning company "Censys" posted last Wednesday that their comprehensive Internet scanner had identified 10,600 publicly accessible instance of Ollama large language models. The first two paragraphs of Censys' posting said:

> *As we write this in September 2025, Large Language Models (LLMs) are So Hot Right Now. For those who may not be familiar with the hype, LLMs are widely used for a range of applications, and frameworks like Ollama make it easy for users to spin up an instance for their personal use. To add to this, many organizations now publish guides to help users spin up LLM instances faster. However, with this ease of use also comes ease of misuse.*
>
> *Like many other technologies on the web, security is an afterthought, and LLMs are no exception. We already know of anecdotal cases where open instances of LLMs are being misused by online actors, so we take our Internet-wide lens to see what Ollama instances look like today. Fortuitously, Censys already has an Ollama scanner that scans for Ollama instances on HTTP, and exposes that data on hosts and endpoints.*

They go on at some length, but they found that the majority of these instances were concentrated in cloud/hosting providers, with some notable exceptions in software-as-a-service companies which appear to have spun up these instances for their customers.

Since Censys probes all 65,535 TCP ports, they found more than 25% of all Ollama instances were listening on ports other than 11,434 which is Ollama's default port. Ollama normally binds to the localhost IP, 127.0.0.1, to restrict Ollama's service availability to the local platform. But if instances are being spun up for others, binding to public ports is understandable. What is not understandable is the apparent total lack of security. The service is designed for local use, so it binds to the localhost port as its security measure. If that service is bound to a public-facing interface, then the LLM will be public and available to all... which is exactly what Censys found and reported.

After apparently obtaining a connection to an Ollama instance, Censys prompted each instance with two probing prompts: *"What is your purpose?"* and *"Could you remind me what your prompt is?"* Of these 10,600 Ollama instances they found, 1500 of those responded to at least one of those prompts, indicating direct interactivity with the model via the exposed API.

Censys wrote:

> *Like many other entities on the Internet, these instances should not be publicly accessible, and definitely not publicly promptable. As technologies proliferate, we must be cautious about what we post online and how it's accessible to others.*

DNS Benchmark

I wanted to quickly update everyone that my past 10 months of work on the development of a major commercial, though inexpensive update to GRC's most-downloaded DNS Benchmark freeware has reached "release candidate" state. So I'd imagine we're likely only weeks away from having this work finished. Its companion website needs updating to synchronize it with what's coming, so I'll be working there and getting ready while the code settles a bit more.

Listener Feedback

Mike Lendvay

Hi Steve,

I wanted to note about the Apple memory protection (MTE) discussed on the last two podcasts that this functionality has also been added to the Cortex line of ARM chips. The implementation is different but the result is similar. Google enables this for its Advanced Protection Mode. Additionally GrapheneOS enables it at a system level and for apps likely to be targeted. It also offers a toggle to enable it for every app automatically, and then disable it if the app won't work.

Mike's note was joined by others who wrote to tell me that Android and the GrapheneOS both had access to the MTE features of the latest ARM chips. And they're 100% correct. V8.5 of the ARM architecture introduced MTE. And Apple also jumped on it immediately, trying to use it for security. But "security" was never MTE's intended target. It was designed as a debugging aid for developers because it could be deliberately configured to detect their mismanagement of memory. The problem was that if it was operated in its asynchronous (delayed notification) mode, that was useful for developers but not for security which needed to prevent any misuse before it was allowed to occur. And operating MTE in fully synchronous (immediate blocking) mode incurs a significant performance overhead, which makes it impractical to use everywhere all the time.

So Apple first worked with ARM to extend MTE, creating EMTE... then they decided to commit the chip real estate resources needed to take those concepts, which had by then been proven to work but still introduced excessive overhead when used for security enforcement, and that resulted in what Apple calls MIE – Memory Integrity Enforcement.

So it's true that generic ARM chips such as the Cortex family do have MTE, but it needs to be used very sparingly when it's employed in synchronous mode for security enforcement.

Mick Fink

Hi Steve, I tried experimenting with passkeys. We use Microsoft Office and Azure at work, and because Entra would not let me install a passkey directly on my Mac for whatever reason, I added it instead to my Microsoft Authenticator app on my iPhone. Here are the two login flows contrasted side by side:

Username & Password

- *Open password manager*
- *Click the Launch button for my Microsoft account.*
  Username and password are filled in automatically
- *A window comes up with a 2-digit code that I have to enter on my MS Auth app:*
  - *Unlock my phone with 6-digit code*
  - *Click on the authenticator app on my phone*
  - *Reenter my phone's device code for the MS Auth app*
  - *Enter the 2-digit code*
- *Tell my computer yes I would like to remain signed in ... And I'm in* 🎉

Now, let's do this again with a Passkey:

- *Click on login bookmark on my computer*
- *Pick an account (I have 2 different accounts, let's use the passkey one)*
- *Click the Next button on the "Face, fingerprint, PIN or security key" welcome screen*
- *Click on where my passkey is saved. (Not on my Mac, I couldn't do that. So let's click on the phone option.)*
- *I receive a QR code to scan*
- *Unlock phone with 6-digit code*
- *Click on MS Authenticator app*
- *Enter 6-digit code again for MS Auth app access*
- *Click on the QR code scanner button*
- *Point the camera at the computer screen, the QR code is seen and registered.*
- *MS Auth app says "Your iPhone needs to connect to this device in order to sign-in with a passkey." Phone's Bluetooth was already enabled, otherwise that would be more clicks. So, click the Continue button to permit MS Auth to proceed with permission.*
- *Sign-in by clicking the Use Passkey button on my phone.*
- *Enter 6-digit phone code, yet again.*
- *Tell my computer yes I would like to remain signed in.*

Now the computer screen FINALLY shows me that I am logged in. Is it any wonder why passkeys are not popular yet? Love your show, Mick Fink

Using Face ID for all of those various intermediate authentication stages makes things easier. The trouble is, there are so many places where abuse could be inserted that we're stuck needing to continually re-authenticate, switch devices, arrange inter-device communications, and jump through hoops. I'm sure that Mick's point is to acknowledge the enhanced security, but to wonder whether it's really worth all the trouble?

And I can really see his point. The biggest threat was having a single password that people used everywhere. But browser-based password managers solved that problem years ago. It's tempting to wonder whether we shouldn't have just left well enough alone. Having a Passkey is useful when super-security is called for. But that's comparatively rare. Since I'm the only person who uses any of my PCs, I have every site possible keeping me logged in permanently anyway. Whenever I need to do anything important, such as managing investments, I'm required to respond to a phone-loop one-time password and to be working from a previously-seen PC.

So I guess my takeaway is, since no one is forcing Passkeys upon us, and since they are truly more hassle to use, don't use them when they're not really needed. And also, using Passkeys in a same-device mode, where you don't need to coordinate between devices is much more convenient while still bringing nearly all of Passkeys' security benefits.

Chris Forrester

Hello Steve!
After listening to the last few weeks' discussions on age verification, I had a thought I'd like to have your input on.

I realize the onus is currently being heavily placed on the provider of age-restricted content, similar to physical locations such as restaurants, bars, and convenience stores which are required to proactively assert a user's age prior to selling them alcohol. However, as has been pointed out several times, this is not the physical world being dealt with in these discussions, and this problem is going to be fought as an unacceptable burden by the providers. Episode 1044 really brought that home to me.

Is it reasonable to believe that we will begin seeing a turn to the client as being the responsible party? For example, let's say there is a single PC in a household of 4 individuals. That PC is a shared device and has a single username. Perhaps these are non-techy people who don't think that's a problem in-and-of-itself. Perhaps it's a grandparent who allows their grandkids to use it when they come over.

I believe it's possible we will see laws enforcing strict user account controls in order to enforce age verification requirements, where each account is associated with an age verification service of some kind, and the abuse of those restrictions is punishable by law. Basically, each account becomes a vault only accessible by the intended user.

It seems like this would fall in line with more real-world scenarios such as, say, a liquor cabinet, where the responsible adult is considered fully liable when accessed by minors. I don't like or agree with it, but I feel like this is a real possibility of where the longer road is leading.

I was a TechTV watcher when I was a kid and I've been a weekly listener to Security Now since 2007 and cannot begin to tell you how much I've learned from you and Leo over the years. You two are 100% responsible for my career in security-focused software development, and I would be honored to have a mention on the podcast. If that were to happen, feel free to use my name! Looking forward to episode 2,000!

Thanks, -- Chris Forrester

What Chris suggests is an interesting extension of the notion I shared last week where a user's browser is aware of its user's date of birth. It never discloses that date... but by using some future W3C browser API extension, a website is able to ask for that information. Rather than giving it over freely, the browser informs its user that it's being asked for their age and the user can then decide whether this is a reasonable request based upon where they are... and they would be free to decline.

Now, presumably, if the user had not in some way authenticated themselves to the browser, the prompting for their age would require that too, somehow.

So what Chris has added to this model is that our authenticated operating system logon sessions would provide that authentication. Our OS account – much as Microsoft now has us logging into them all the time – might possess a confirmed date of birth which a browser running on the system could inherit from the logged-on user and then only need to confirm whether the user wishes to let the site they're visiting know. And, conceivably, browsers could also be preset with a "never", "always" or "always ask" setting to decide how to handle remote age requests.

And I'm with Chris and I'm sure most of those listening, this is all a big mess. But laws are being written whether we like them or not, and we're already seeing services and sites we use either

pulling up stakes or working to remain online and compliant with these emerging laws.

Brian Tanner

Hey Steve, A while back you pointed to a video by a LLM guy that was a soup to nuts explanation of how the whole LLM system works. Try as I might, I can find neither the reference nor the video. Could you point again please? Brian

Absolutely. I'm sure that Brian is recalling the amazing presentations by Grant Sanderson, the "Animated Math" wizard over at https://www.3blue1brown.com/. Go to 3Blue1Brown.com (in both cases using the characters for the numbers) then click on "Neural Networks" topic since today's LLMs are direct outgrowths of the decades long experimentation with neural networks. No one thought to just throw an insane amount of resources at them and then feed the entire Internet into it to see what happens: https://youtu.be/aircAruvnKk

Ryan Lloyd

Hi Steve, I watched the video linked to in episode 1044 on the digital age verification implementation from the EU. I found it interesting, working closely with companies in this space.

I can't help but think there is one major oversight in this privacy preserving approach however. There appears to be no biometric/liveness verification that occurs at any point during the workflow. It appears to just be providing an identity card to the app. It seems to me a user could easily obtain such an identity numerous ways;

- "borrow" from their parents
- reuse from a cousin
- obtain on the internet from a reddit forum or black market

Young people facing a blockage in gaining access to restricted content / material tend to be resourceful so I'm struggling to see how this approach will really help ensure end users are kept out. It feels like this is less about verifying the age of the user behind the screen as it is about verifying the user can obtain a scan of an identity artefact that is of age.

I welcome your thoughts on this.

I had similar concerns when you raised the idea of browsers bearing the burden of age verification in the future. My feeling is that even if a browser implements this, the browser has no guarantee that the user with physical access to the device is the same user who verified their age at some point in the history of setting up the browser, unless you wish to inconvenience every site with re-verification using liveness techniques.

Thanks, Ryan

I agree completely with everything Ryan observed. Our listeners will recall how many times I've noted that to truly solve this problem, any assertion of identity must be closely bound to some effective biometric authentication. If we don't do that we're just creating another easily bypassed solution. The result being that everyone is inconvenienced while the actual problem

remains unsolved. What we learn is just how difficult it is to bring real world solutions to the cyber world.

Lee MacKinnell

Hi Steve: I decided to see if I could block my browser's access to localhost, so I asked google and i got this result:
https://superuser.com/questions/1823747/how-to-block-connections-to-localhost-by-non-localhost-webpages-on-firefox

"I am migrating to Firefox from Safari and noticed that websites (ANY website) that I visit has permission to attempt to make connections to localhost ports..... Is there a way to configure Firefox to block such connections? Safari already does this natively."

and there is a response:

"UBlock Origin has this capability, but it is not enabled by default.
To enable it, enable Filter lists > Privacy > Block Outsider Intrusion into LAN."

This also works for Chrome-based browsers like Brave as it is also a setting in the UBlock Origin Lite version.

Lee MacKinnell / Brisbane Australia

That's very cool and very useful. Thanks Lee!

Joey Albert

Steve, October 14: https://www.microsoft.com/en-us/windows/end-of-support

We use 0patch and it makes sense. The TSR patches in RAM and reapplies all patches when rebooting. So the subscription makes sense. It's like subscribing to streaming services. If your subscription expires, no more music.

0patch is $41 (translated from Euros which they charge). Windows 10 support is $30 for consumers but not for businesses. For businesses it's $61 for the first year and doubles each year after. It stops after 3 years. So 0patch is a bargain! Joey Albert

Joey's point is a good one. We need to remember that all of this Win10 ESU stuff is only for end-user consumers. Not for commercial business users. As far as I know at this time, though we know how fluid the situation is, business users are not being given exceptions.

???