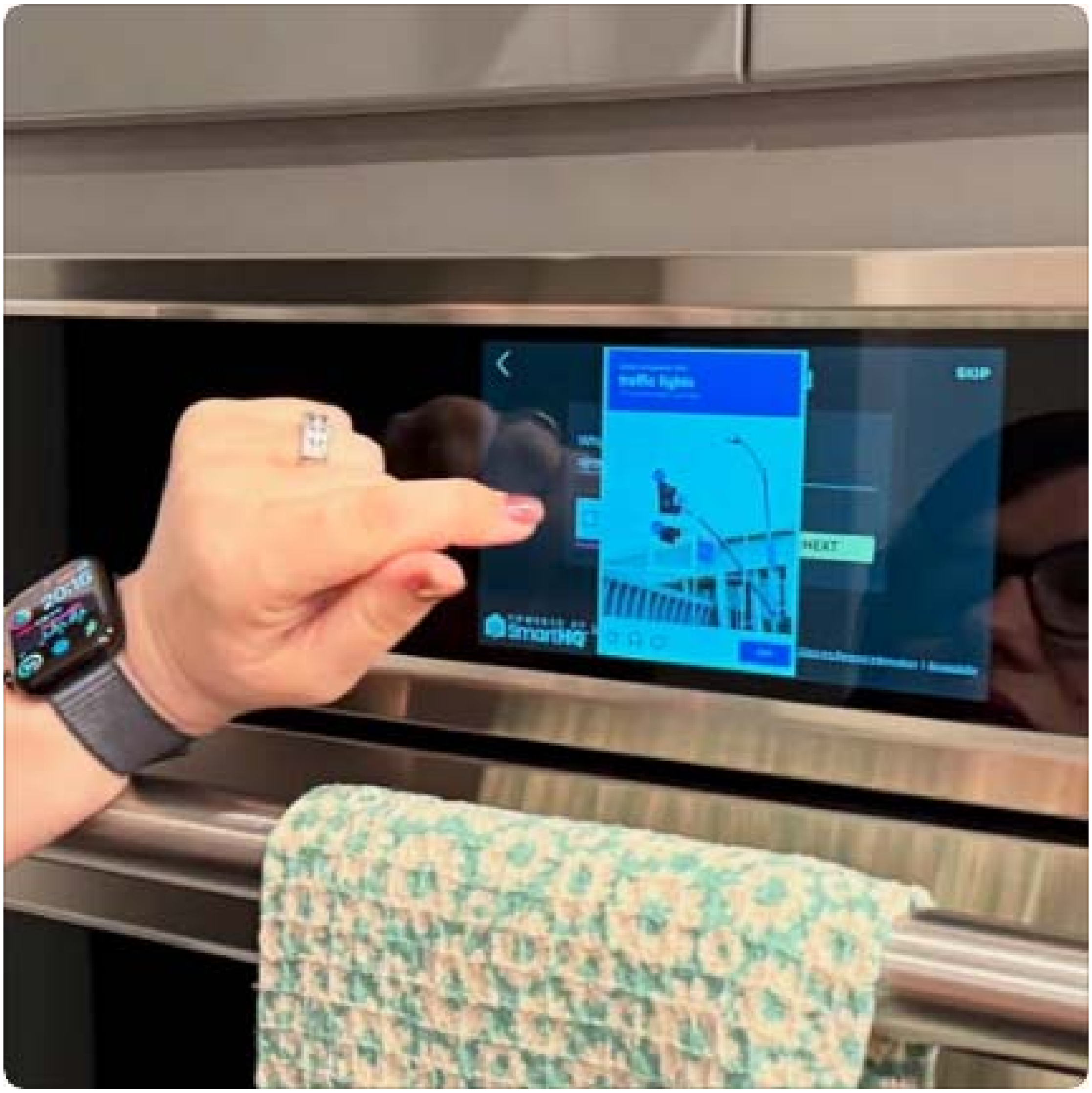
Security Now! #1029 - 06-10-25

The Illusion of Thinking

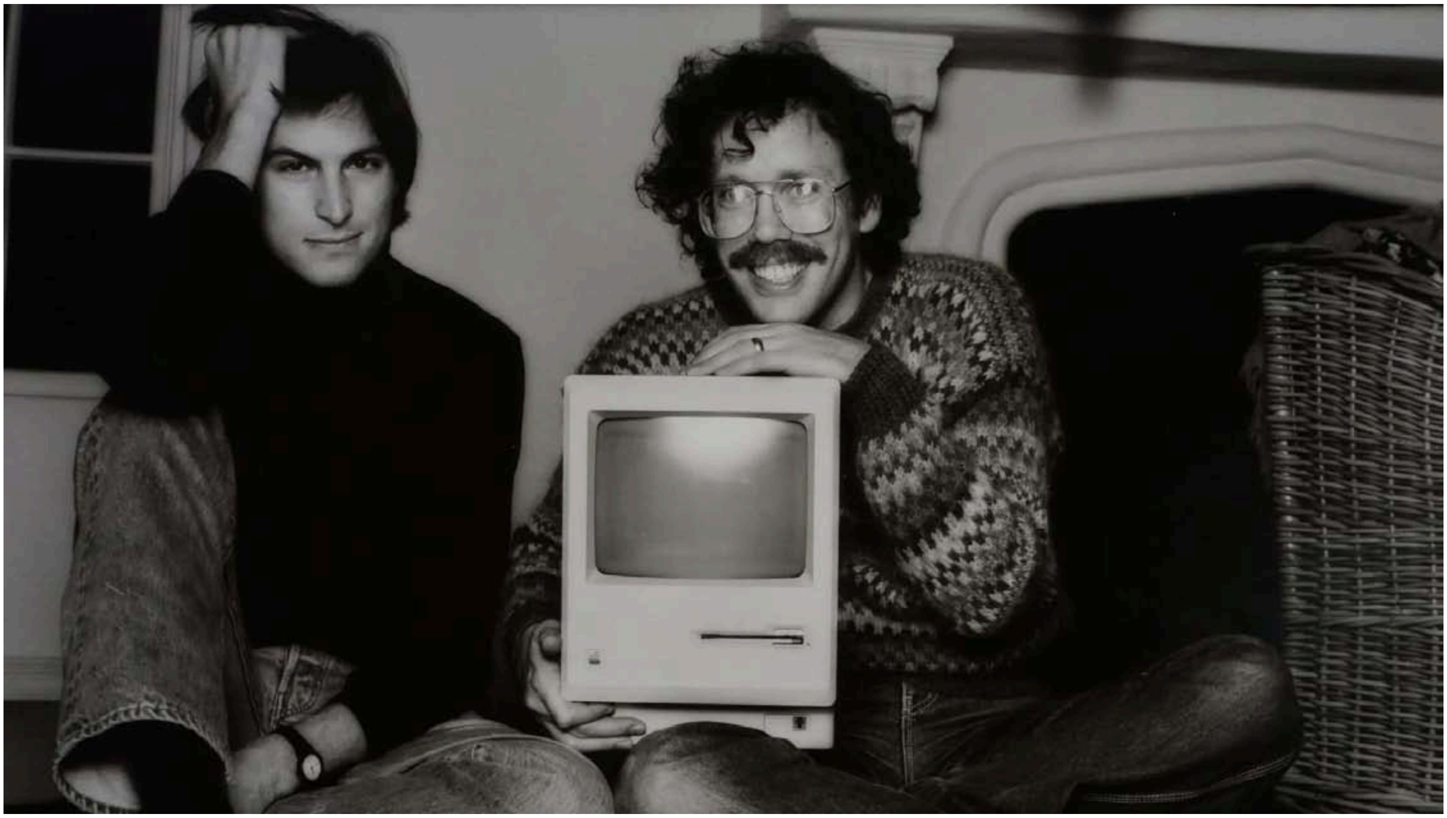
This week on Security Now!

• In memoriam: Bill Atkinson • Meta native apps & JavaScript collude for a localhost local mess. • The EU rolls out its own DNS4EU filtered DNS service. • Ukraine DDoS's Russia's Railway DNS ... and... so what? • The Linux Foundation creates an alternative Wordpress package manager. • Court tells OpenAI it must NOT delete ANYONE's chats. Period! :(• A CVSS 10.0 in Erlang/OTP's SSH library. • Can Russia intercept Telegram? Perhaps. • Spain's ISPs mistakenly block Google sites. • Reddit sues Anthropic. • Twitter's new encrypted DM's are as lame as the old ones. • The Login.gov site may not have any backups. • Apple explores the question of recent Large Reasoning Models "thinking".

If your kitchen oven challenges you to prove you're human, something has gone very wrong somewhere.



Bill Atkinson



I wanted to take a moment to note with sadness the passing, too soon, of Bill Atkinson who died last Thursday, June 5th, after losing his battle with pancreatic cancer. That's also what took Steve Jobs 14 years earlier in 2011 when Steve was only 56. Bill was born in 1951, so he made it to 74.

Bill wrote of himself in the 3rd person for the "About" page of his website: *"Aside from being a nature photographer, he is also well known in the world of software design. Years ago, as a member of the original Macintosh team at Apple, he helped design much of the initial Macintosh user interface and wrote the original QuickDraw, MacPaint, and HyperCard software."* Bill was being modest. He received his undergraduate degree from UC San Diego, where he met the now also famous Apple alumnus Jef Raskin (Apple employee #31) who was one of his professors. Bill continued his studies as a grad student in neurochemistry at the University of Washington. Later, Raskin invited Atkinson to visit him at Apple, where Steve Jobs got his hands on him and persuaded him to forget school and join the company immediately as employee #51.

At Apple, Bill became the principal designer and developer of the GUI for Apple's Lisa, and later became one of the first thirty members of the original Apple Macintosh dev team where he also principally designed the Mac's UI. He was the author of MacPaint which, at the time, was an astonishing piece of work. No one could believe it. MacPaint was built upon the QuickDraw toolbox, which Bill had first written for the Lisa and ported to the Mac. And need I note that QuickDraw was 100% pure Mororola 68000 assembly language? Much of that code is some of the most beautiful high performance bitmapped graphics code anyone has ever seen. Its high performance was essential to the success of the Macintosh and its graphical user interface.

Bill also conceived, designed and implemented HyperCard which gave non-programmers access to programming and database design. Many years later, in 1994, Bill received the EFF Pioneer Award for his contributions to the field of personal computing.

Security News

“Local Mess”

If anyone might be at all unsure about just how badly the likes of Meta are determined to surreptitiously track their users’ movements around the Internet for the purpose of secretly profiling them, the news I have to share about a recent super-sneaky tracking discovery will disabuse anyone of any doubts along those lines. (https://localmess.github.io/)

To quickly lay out what it does and how it work, the write-up of this begins with a quick overview:

> *We disclose a novel tracking method by Meta and Yandex, potentially affecting **billions** of Android users. We found that native Android apps—including Facebook, Instagram, and several Yandex apps including Maps and Browser—silently listen on fixed local ports for tracking purposes.*

I’ll interrupt to note that that’s actually, it’s kind of diabolically brilliant, though it’s not completely new. For example, my own native Windows SQRL client, running in the user’s machine, opens and listens on port 25519 for connections from a SQRL script running on login pages. SQRL login javascript on a website’s login page would send the SQRL client a unique token by opening a TCP connection to the localhost IP where the resident SQRL client app was listening. The SQRL client app would then connect to the remote site at the URL provided by the website which contained a unique token. It would identify its user and use the unique token to perform a secure public- key authentication. Upon authentication success, the remote site would return a URL which the SQRL client would forward to the waiting web browser, which would then jump to the user to the site with the user now logged-on. Thus, “Presto!” without doing anything the user would be logged in with complete security that could not be hacked, spoofed or intercepted.

So the idea of allowing a webpage’s JavaScript to talk to a local native app is not entirely new. But, of course, what SQRL was doing was above board and fully documented as part of the protocol. That is decidedly not the case with Meta and Yandex who were doing this purely for tracking. During the development of SQRL there was some worry about this handy facility disappearing, since Microsoft was aware of the potential for the abuse of this and for a while they tried to shut down browser access to the localhost IP from the web browser. But there are many other legitimate use-cases for this that too many things broke and Microsoft was forced to backpedal and leave the facility in place.

The guys who discovered Meta and Yandex’s abuse of this continued:

> *These native Android apps receive browsers' metadata, cookies and commands from the Meta Pixel and Yandex Metrica scripts embedded on 5.8 million web sites. These JavaScripts load on users' mobile browsers and silently connect with native apps running on the same device through localhost sockets. Since native apps have access to device identifiers like the Android Advertising ID (AAID) or directly handle actual user identities as in the case of Meta apps, this method effectively allows these organizations to link mobile browsing sessions and web cookies to real world user identities, de-anonymizing users visiting sites embedding their scripts.*
>
> *This web-to-app ID sharing method bypasses all typical privacy protections such as clearing cookies, Incognito Mode and all of Android's permission controls. It also opens the door for potentially malicious apps eavesdropping on users’ web activity.*

So what we have here is an interesting and extremely privacy-invasive hack. The concern is that this is **not** leveraging some bug that can be found, fixed and eliminated. As I noted, Microsoft previously tried and failed to eliminate this capability. So that everyone is clear about this, the problem Microsoft had with cutting off their browser from all access to the local machine is that it has always been possible to do this and, as we've often seen, anytime something is possible it will eventually be done. And once applications have become dependent upon some available mechanism it's extremely difficult to take that away. For example, many web developers run local web servers on their machines and test their web code locally on web browsers running on the same machines. It's entirely practical and easier than needing to setup some second external web server. Another example is that web browsers have become so powerful that a local application might run "headless" without any desktop presence of its own. Instead, it will launch the system's web browser for all communication. The user experiences it as a website, but they're communicating with an application that's running in their own local machine. This is done by running a web server on the local machine which the web browser communicates with.

So Meta and Yandex are both abusing this deliberate and formally supported ability of web browsers not only to connect to far away remote servers out on the Internet, but also to little local servers setup and running inside any application on the same machine — and there's no obvious way any user can know this is going on, let alone prevent it from happening.

Since this problem is not going away, let's take a closer look at what these researchers found. They wrote:

> *While there are subtle differences in the way Meta and Yandex bridge web and mobile contexts and identifiers, both of them essentially misuse the unvetted access to localhost sockets. The Android OS allows any installed app with the INTERNET permission to open a listening socket on the loopback interface (127.0.0.1). Browsers running on the same device also access this interface without user consent or platform mediation. This allows JavaScript embedded on web pages to communicate with native Android apps and share identifiers and browsing habits, bridging ephemeral web identifiers to long-lived mobile app IDs using standard Web APIs.*
>
> *The Meta (Facebook) Pixel JavaScript, when loaded in an Android mobile web browser, transmits the first-party _fbp cookie using WebRTC to UDP ports 12580–12585 to any app on the device that is listening on those ports. We found Meta-owned Android apps Facebook and Instagram, available on the Google Play Store, listening on this port range.*

So here's the step-by-step of this in detail:

1. In their normal course of use, the user opens their native Facebook or Instagram app on their device. The app is eventually switched away from, is sent to the background, and creates a background service to listen for incoming traffic on a TCP port (12387 or 12388) and a UDP port (the first unoccupied port in the range 12580-12585). Users must be logged-in with their credentials on the apps.

2. The user opens their web browser and visits any one of 5.8 million websites integrating the Meta Pixel.

3. Websites may ask for consent depending on the website's and visitor's locations.

4. The Meta Pixel script sends the _fbp cookie to the native Instagram or Facebook app using WebRTC protocol.

5. The Meta Pixel script simultaneously sends the _fbp value in a request to https://www.facebook.com/tr (gee, do you think "tr" might be short for "track"?). The URL's query tail contains other parameters such as page URL (dl), website and browser metadata, and the event type (ev) (e.g., PageView, AddToCart, Donate, Purchase).

6. The Facebook or Instagram apps receive the _fbp cookie from the Meta Pixel JavaScript running on the browser. The apps transmit _fbp to https://graph.facebook.com/graphql along with other persistent user identifiers, linking users' fbp ID (web visit) with their Facebook or Instagram account.

The researchers explain:

> *According to Meta's Cookies Policy, the _fbp cookie "identifies browsers for the purposes of providing advertising and site analytics services and has a lifespan of 90 days." The cookie is present on approximately 25% of the top million websites, making it the 3rd most common first-party cookie of the web, according to Web Almanac 2024.*
>
> *A first-party cookie implies that it cannot be used to track users across websites, as it is set under the website's domain. That means the same user has different _fbp cookies on different websites. However, the method we disclose allows the linking of the different _fbp cookies to the same user, which bypasses existing protections and runs counter to user expectations.*

Yep. So just to be clear, this entire surreptitious surveillance system was specifically designed to explicitly and deliberately bypass not only all user-expressible anti-tracking wishes, but also to circumvent all of the work the browser vendors have invested in to limit cross-site tracking. This neatly circumvents all of the explicit 1st-party domain-tied cookie isolation and stovepiping that our web browsers have added specifically to prevent the abuse of the original cookie system.

Let me be very clear about this: There can be no other reason for this. Based upon the behavior of this system which these researchers have observed, there can be no other reason for this. It is entirely indefensible.

So that's what Meta has been up to. How does the Russian search service Yandex compare? These researchers write:

> *Since 2017, the Yandex Metrica script initiates HTTP requests with long and opaque parameters to localhost through specific TCP ports: 29009, 29010, 30102, and 30103. Our investigation revealed that Yandex-owned applications—such as Yandex Maps, Navigator, Search and Browser— actively listen on these ports. Furthermore, our analysis indicates that the domain yandexmetrica.com is resolving to the loopback address 127.0.0.1, and that the Yandex Metrica script transmits data via HTTPS to local ports 29010 and 30103. This design choice obfuscates the data exfiltration process, thereby complicating conventional detection mechanisms.*

In other words, it's quite sneaky to have a public domain like yandexmetrica.com resolving to the local host IP 127.0.0.1 since script code analyzers would likely look for the string "localhost" or the IP "127.0.0.1" – but Yandex embeds a public-appearing domain name to furtherobscure what's actually going on. And their use of HTTPS means that any communications is also obscured and is less easy to intercept, monitor and analyze. And then Yandex gets even trickier.

The researchers explain:

> *Yandex apps contact a Yandex domain (startup.mobile.yandex.net, or similar) to retrieve the* ***list of ports*** *to listen to. The endpoint returns a JSON containing the local port number (e.g., 30102, 29009) and a "first_delay_seconds" parameter which we believe is used to delay the initiation of the service. On one of our test devices, first_delay_seconds roughly corresponded to the number of seconds it took for the Yandex app to begin listening on local ports – which was around 3 days.*

The only POSSIBLE reason for this is to avoid detection and to prevent any researchers from easily discovering this deliberately concealed behavior. It really is despicable. They write:

> *After receiving the localhost HTTP requests from the Yandex Metrica script, the mobile app responds with a Base64-encoded binary payload embedding and bridging the Android Advertising ID (AAID) among other identifiers accessible from Java APIs like Google's advertising ID and UUIDs, potentially Yandex-specific. As opposed to Meta's Pixel case, all of this information is aggregated and uploaded together to the Yandex Metrica server (e.g., mc.yango.com) by the JavaScript code running on the web browser, rather than by the native app. In the case of Yandex, the native app acts as a proxy to collect native Android-specific identifiers, then transferring them to the browser context through localhost sockets.*

In other words, Meta has their native Facebook or Instagram app doing the communicating with the Meta mothership whereas The various Yandex apps run native servers that the Yandex JavaScripts communicate with in order to obtain whatever device-specific information Yandex might wish. That information is then returned to the browser from the little local Yandex servers, which the Yandex JavaScript then forwards to Yandex.

The researchers point out an additional problem under their heading *"Additional risk: Browsing history leak"*, writing:

> *Using HTTP requests for web-to-native ID sharing may expose users browsing history to third-parties. A malicious third-party Android application that also listens on the aforementioned ports can intercept the HTTP requests sent by the Yandex Metrica script and Meta's communication channel by monitoring the Origin HTTP header.*
>
> *We developed a proof-of-concept app to demonstrate the feasibility of this browsing history harvesting by any malicious third-party app. We found that browsers such as Chrome, Firefox and Edge are susceptible to this form of browsing history leakage in both default* ***and private browsing modes.*** *The Brave browser was unaffected by this issue due to their blocklist and the blocking of requests to the localhost; and DuckDuckGo was only minimally affected due to missing domains in their blocklist.*
>
> *While the possibility for other apps to listen to these ports exist, we have not observed any other app, not owned by Meta or Yandex, listening to these ports.*
>
> *Due to Yandex using HTTP requests for its localhost communications, any app listening on the required ports can monitor the website a user visited with these tracking capabilities as demonstrated by the video above. We first open our proof of concept app, which listens to the ports used by Yandex, and send it to the background. Next, we visit five websites across different browsers. Afterwards, we can see the URLs of these five sites listed in the app.*

In other words, once this local system abuse is present there's nothing to prevent other apps from establishing their own competing services and hooking into this illicit extra-browser communications to obtain – for their own purposes – the same Internet-wide tracking and monitoring that the Meta and Yandex apps are deliberately employing.

Summarizing things, they wrote:

> *This novel tracking method exploits unrestricted access to localhost sockets on the Android platforms, including most Android browsers. As we show, these trackers perform this practice without user awareness, as current privacy controls — sandboxing approaches, mobile platform and browser permissions, web consent models, incognito modes, resetting mobile advertising IDs, or clearing cookies — are all insufficient to control and mitigate it.*
>
> *We note that localhost communications may be used for legitimate purposes such as web development. However, the research community has raised concerns about localhost sockets becoming a potential vector for data leakage and persistent tracking. To the best of our knowledge, however, no evidence of real-world abuse for persistent user tracking across platforms has been reported until our disclosure.*
>
> *Our responsible disclosure to major Android browser vendors led to several patches attempting to mitigate this issue; some already deployed, others currently in development. We thank all participating vendors (Chrome, Mozilla, DuckDuckGo, and Brave) for their active collaboration and constructive engagement throughout the process. Other Chromium-based browsers should follow upstream code changes to patch their own products.*
>
> *However, beyond these short-term fixes, fully addressing the issue will require a broader set of measures as they are not covering the fundamental limitations of platforms' sandboxing methods and policies. These include user-facing controls to alert users about localhost access, stronger platform policies accompanied by consistent and strict enforcement actions to proactively prevent misuse, and enhanced security around Android's interprocess communication (IPC) mechanisms, particularly those relying on localhost connections.*

And I'll add that while these guys are only focusing upon mobile platforms this is not a mobile-only problem. My implementation, and others, of this legitimate intra-platform communication for SQRL's use works cross-platform everywhere – on both mobile and desktop. So we know that there are currently no controls for this.

My own feeling is that no browsers should allow this by default. It's just too dangerous to permit out of the box. So the default should be for browsers to block and notify their user when any website they visit attempts to open a backdoor channel to something running – perhaps surreptitiously – on their local machine. Any legitimate use of this, such as for web development, would then expect and permit this. And a browser might offer some configuration. There might be three settings: block and don't notify, request permission, and always allow. And as another option – since, for example, Firefox certainly appears to have no upper limit on the number of fine-grained configuration settings it's able to manage – a user might permit this localhost network communication only over certain ports, such as the standard web ports 80 and 443 to permit local web server access while blocking all other high ports that apps might use.

Technology aside though, this makes one sort of shake one's head. Yandex is Russian so they're not friends of the West and they're certainly not on any friendship trajectory. But Meta is a huge and, we would wish, responsible U.S. corporation that would like to have and deserve the trust of its users.

But the design and installation of these covert backdoors in their apps – which can only have the purpose of communicating with matching user-tracking web scripts spread across 5.8 million Internet sites – really deserves the attention of U.S. authorities.

Meta knows this was wrong because this horrifying behavior was immediately shut down, the same day, after the publication of this research. They got caught bypassing all user choice and anti-tracking browser enforcement and immediately turned it off. They're able to do this since those JavaScripts are all being sourced by their own content delivery network. So it was only a matter of changing the code being sent by the mothership. But their apps will still be opening and listening for any local web browser connections. Who's to say where, when and how they might attempt to resume this behavior in the future?

Okay... so what else has been going on?

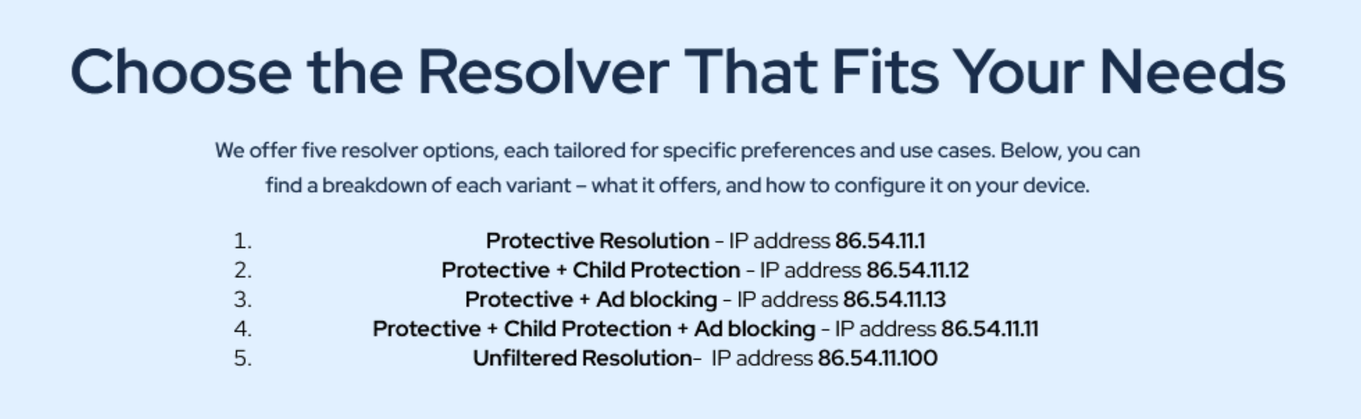
DNS4EU — https://www.joindns4.eu/
Last week the European Union launched its own multi-flavor DNS service. There are flavors for governments, telcos and home users. This new DNS4EU service is designed to provide secure and privacy-focused DNS resolvers for the EU bloc as an alternative to US and other foreign services. The project was first announced back in October 2022 and was built under the supervision of the EU cybersecurity agency ENISA.

It's currently managed by a consortium led by the Czech Republic security firm Whalebone, and members include cybersecurity companies, CERTs, and academic institutions from 10 EU countries. I confirmed the "Whalebone" ownership since I immediately dropped the various DNS resolver IPs into GRC's DNS Benchmark and the Benchmark's ownership tab showed they were all within a network owned by "WHALEBONE S.R.O."

Naturally, these EU DNS resolvers include built-in DNS filters for malicious and malware-linked domains that prevent users from connecting to known bad sites. The lists are managed from a central location by EU threat intel analysts at no cost to users, companies, or any governments that might decide to adopt the service.

The pitch to governments and telcos is that having the EU offer a trusted DNS service can eliminate the costs associated with running their own DNS infrastructure. And to the degree that independent DNS services required security personnel to manage and filter the directory, that can now be offloaded to the dedicated DNS4EU team.

The variations DNS targeted toward home users is available with different filtering profiles will while not resolve malicious domains, adult content, ads, all three, or none:

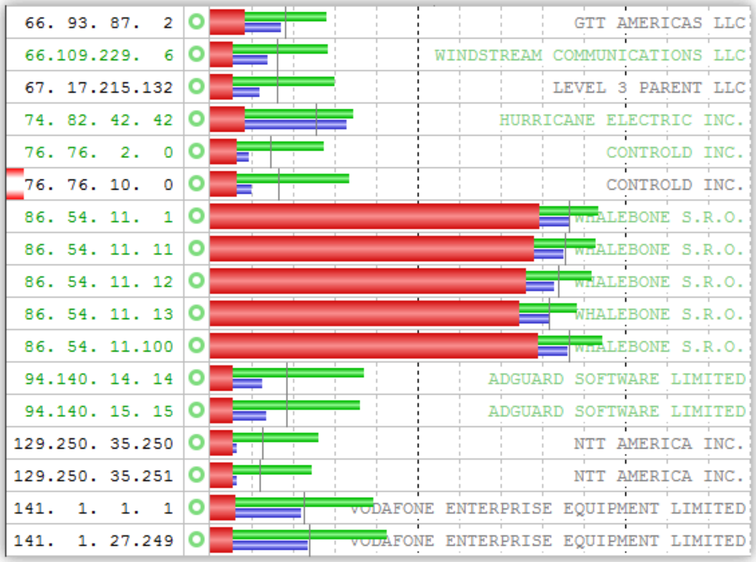


Choose the Resolver That Fits Your Needs

We offer five resolver options, each tailored for specific preferences and use cases. Below, you can find a breakdown of each variant – what it offers, and how to configure it on your device.

1. **Protective Resolution** - IP address **86.54.11.1**
2. **Protective + Child Protection** - IP address **86.54.11.12**
3. **Protective + Ad blocking** - IP address **86.54.11.13**
4. **Protective + Child Protection + Ad blocking** - IP address **86.54.11.11**
5. **Unfiltered Resolution**- IP address **86.54.11.100**

While it would be nice to have government-backed free DNS web content filtering, being the guy behind what has pretty much become the industry-standard DNS benchmark, I wondered how those five resolver IPs list on the DNS4EU website performed:



The chart I’ve included in the show notes brings the word “atrocious” to mind. Their average response time ranged between 163 and 173 milliseconds. Compare that to Cloudflare’s DNS that the same benchmark run averaged at 20 milliseconds. Now to be fair, the European Union is not suggesting that someone located in Southern California should use their DNS at all, let alone expect stellar performance from it. So I posted into the “grc.dns.dev” newsgroup where we’ve all been testing the evolving next-generation DNS benchmark code, asking anyone who’s located in the EU to give the same set of DNS IPs a run. So far I haven’t heard back. But I’m sure I’ll have ample feedback by next week.

These new DNS services are available under IPv4 and IPv6 for DNS over UDP, and DoH or DoT for privacy enforcing secure DNS over TCP/TLS. The DNS Benchmark also reports that all services support the DNSSEC security extensions to offer signed and thus unalterable DNS replies. Although some reporting has raised questions about whether the service could be used to censor Europeans, the use of this DNS obviously cannot be mandatory and the EU said the project's purpose is to ensure digital sovereignty, which seems entirely reasonable.

Anyone in the EU wishing to explore this further should jump their browser over to joindns4.eu for all the information as well as IPv6 and DoH and DoT URLs. Everything is there.

GUR takes down Russian Railways sites

While we’re on the topic of DNS, Ukraine's military intelligence agency claims that it took down the DNS service of the Russian Railways using a 6 Gbps / 2.5 million packets per second DDoS attack. The reporting was in Ukrainian news, in Ukranian, and didn’t dig any further. It’s unclear to me what that accomplishes. As we know, any attack on DNS would need to be sustained until local DNS caches expire. At that point things would begin collapsing. But it also wasn’t clear that would then collapse. Would the trains no longer run? Would scheduling and ticket sales fail?

Using a large number of inexpensive stealthfully inserted autonomous drones to remotely take out many extremely expensive Russian cruise missile launching warplanes. Now THAT’s something to write home about.

The Linux Foundation launches the FAIR Wordpress package manager
Given the astonishing number of websites that use the Wordpress core as their content management system (CMS) I always want to keep our listeners abreast of any important Wordpress-related news. So when the Linux Foundation announces the launch of their replacement for Wordpress.org's own package manager, that news makes the cut.

I haven't kept up to date with the politics surrounding Wordpress and Automatic. But the reporting I saw said *"The new system is a decentralized alternative to the WordPress.org plugin and theme ecosystem developed with help from veteran WordPress developers who were pushed out from the main WordPress project last year during a power grab by Automattic and Matt Mullenweg."* What I do know is that this replacement looks pretty sweet. The "fairpm" page on Github (https://github.com/fairpm) explains:

> *The FAIR Package Manager is an open-source initiative backed by the Linux Foundation. Our goal is to rethink how software is distributed and managed in the world of open web publishing. We focus on decentralization, transparency, and giving users more control. Our community brings together developers, infrastructure providers, and open web contributors and advocates who all share the same mission: to move away from centralized systems and empower site owners and hosting providers with greater independence.*
>
> *FAIR is governed through open working groups and consensus-driven processes, ensuring that its development reflects the needs of the broader community. Whether you're a contributor, a host, or an end user, FAIR invites participation at every level, from writing code and documentation, to community organisation and governance. As a community-led project, we aim to build public digital infrastructure that is both resilient and fair.*
>
> *The FAIR Package Manager is a decentralized alternative to the central WordPress.org plugin and theme ecosystem, designed to return control to WordPress hosts and developers. It operates as a drop-in WordPress plugin, seamlessly replacing existing centralized services with a federated, open-source infrastructure.*
>
> *There are two core pillars of the FAIR system:*
>
> - *API Replacement: It replaces communication with WordPress.org APIs (such as update checks and event feeds) using local or FAIR-governed alternatives. Some features—like browser version checks—are handled entirely within the plugin using embedded logic (e.g., browserslist).*
>
> - *Decentralized Package Management: FAIR introduces a new package distribution model for themes and plugins. It supports opt-in packages that use the FAIR protocol and enables hosts to configure their own mirrors for plugin/theme data using AspirePress or their own domains. While stable plugins currently use mirrors of WordPress.org, future versions will fully support FAIR-native packages.*

This seems like a useful addition to the Internet's #1 web authoring and delivery system.

OpenAI slams court order to save all ChatGPT logs, including deleted chats
I was reminded of my recent discovery and reporting of the privacy-preserving and unfiltered conversational AI "Venice.ai" when I saw ArsTechnica's headline *"OpenAI slams court order to save all ChatGPT logs, including deleted chats"*. With the subhead: *"OpenAI defends privacy of hundreds of millions of ChatGPT users."* Yikes! And when Ars says "all ChatGPT logs" they

mean **all of every user's** ChatGPT logs, not just those of selected users. So this is ***everyone's*** ChatGPT interactions, period. It seems clearly better for ChatGPT to never have any logs to save in the first place, which is one of the features of that "Venice.ai" service. To understand what's going on here, the details are worth sharing. Here's that Ars' reported:

> *OpenAI is now fighting a court order to preserve **all** ChatGPT user logs—including deleted chats and sensitive chats logged through its API business offering—after news organizations suing over copyright claims accused the AI company of destroying evidence.*
>
> *OpenAI explained in a court filing demanding oral arguments in a bid to block the controversial order: "Before OpenAI had an opportunity to respond to those unfounded accusations, the court ordered OpenAI to 'preserve and segregate all output log data that would otherwise be deleted on a going forward basis until further order of the Court (in essence, the output log data that OpenAI has been destroying)."*
>
> *In the filing, OpenAI alleged that the court rushed the order based only on a hunch raised by The New York Times and other news plaintiffs. And now, without "any just cause," OpenAI argued, the order "continues to prevent OpenAI from respecting its users' privacy decisions." That risk extended to users of ChatGPT Free, Plus, and Pro, as well as users of OpenAI's application programming interface (API), OpenAI said.*
>
> *The court order came after news organizations expressed concern that people using ChatGPT to skirt paywalls "might be more likely to 'delete all [their] searches' to cover their tracks," OpenAI explained. Evidence to support that claim, news plaintiffs argued, was missing from the record because so far, OpenAI had only shared samples of chat logs that users had agreed that the company could retain. Sharing the news plaintiffs' concerns, the judge, Ona Wang, ultimately agreed that OpenAI likely would never stop deleting that alleged evidence absent a court order, granting news plaintiffs' request to preserve all chats.*
>
> *OpenAI argued that the May 13 order was premature and should be vacated, until, "at a minimum," news organizations can establish a substantial need for OpenAI to preserve all chat logs. They warned that the privacy of hundreds of millions of ChatGPT users globally is at risk every day that the "sweeping, unprecedented" order continues to be enforced.*
>
> *OpenAI argued: "As a result, OpenAI is forced to jettison its commitment to allow users to control when and how their ChatGPT conversation data is used, and whether it is retained."*
>
> *Meanwhile, there is no evidence beyond speculation yet supporting claims that "OpenAI had intentionally deleted data," OpenAI alleged. And supposedly there is not "a single piece of evidence supporting" claims that copyright-infringing ChatGPT users are more likely to delete their chats.*
>
> *OpenAI argued: "OpenAI did not 'destroy' any data, and certainly did not delete any data in response to litigation events. The Order appears to have incorrectly assumed the contrary."*
>
> *At a conference in January, Wang raised a hypothetical in line with her thinking on the subsequent order. She asked OpenAI's legal team to consider a ChatGPT user who "found some way to get around the pay wall" and "was getting The New York Times content somehow as the output." If that user "then hears about this case and says, 'Oh, whoa, you know I'm going to ask them to delete all of my searches and not retain any of my searches going forward,'" the judge asked, wouldn't that be "directly the problem" that the order would address?*

OpenAI does not plan to give up this fight, alleging that news plaintiffs have "fallen silent" on claims of intentional evidence destruction, and the order should be deemed unlawful.

For OpenAI, risks of breaching its own privacy agreements could not only "damage" relationships with users but could also risk putting the company in breach of contracts and global privacy regulations. Further, the order imposes "significant" burdens on OpenAI, supposedly forcing the ChatGPT maker to dedicate months of engineering hours at substantial costs to comply, OpenAI claimed. It follows then that OpenAI's potential for harm "far outweighs News Plaintiffs' speculative need for such data," OpenAI argued.

"While OpenAI appreciates the court's efforts to manage discovery in this complex set of cases, it has no choice but to protect the interests of its users by objecting to the Preservation Order and requesting its immediate vacatur," OpenAI said.

Millions of people use ChatGPT daily for a range of purposes, OpenAI noted, "ranging from the mundane to profoundly personal."

People may choose to delete chat logs that contain their private thoughts, OpenAI said, as well as sensitive information, like financial data from balancing the house budget or intimate details from workshopping wedding vows. And for business users connecting to OpenAI's API, the stakes may be even higher, as their logs may contain their companies' most confidential data, including trade secrets and privileged business information.

"Given that array of highly confidential and personal use cases, OpenAI goes to great lengths to protect its users' data and privacy," OpenAI argued.

It does this partly by "honoring its privacy policies and contractual commitments to users"—which the preservation order allegedly "jettisoned" in "one fell swoop."

Before the order was in place mid-May, OpenAI only retained "chat history" for users of ChatGPT Free, Plus, and Pro who did not opt out of data retention. But now, OpenAI has been forced to preserve chat history even when users "elect to not retain particular conversations by manually deleting specific conversations or by starting a 'Temporary Chat,' which disappears once closed," OpenAI said. Previously, users could also request to "delete their OpenAI accounts entirely, including all prior conversation history," which was then purged within 30 days.

While OpenAI rejects claims that ordinary users use ChatGPT to access news articles, the company noted that including OpenAI's business customers in the order made "even less sense," since API conversation data "is subject to standard retention policies." That means API customers couldn't delete all their searches based on their customers' activity, which is the supposed basis for requiring OpenAI to retain sensitive data.

"The court nevertheless required OpenAI to continue preserving API Conversation Data as well," OpenAI argued, in support of lifting the order on the API chat logs.

Users who found out about the preservation order panicked, OpenAI noted. In court filings, they cited social media posts sounding alarms on LinkedIn and X. They further argued that the court should have weighed those user concerns before issuing a preservation order, but "that did not happen here."

One tech worker on LinkedIn suggested the order created "a serious breach of contract for

> *every company that uses OpenAI," while privacy advocates on X warned, "every single AI service 'powered by' OpenAI should be concerned."*
>
> *Also on LinkedIn, a consultant rushed to warn clients to be "extra careful" sharing sensitive data "with ChatGPT or through OpenAI's API for now," warning, "your outputs could eventually be read by others, even if you opted out of training data sharing or used 'temporary chat'!"*
>
> *People on both platforms recommended using alternative tools to avoid privacy concerns, like Mistral AI or Google Gemini, with one cybersecurity professional on LinkedIn describing the ordered chat log retention as "an unacceptable security risk."*
>
> *On X, an account with tens of thousands of followers summed up the controversy by suggesting that "Wang apparently thinks the NY Times' boomer copyright concerns trump the privacy of EVERY @OpenAI USER—insane!!!"*
>
> *The reason for the alarm is "simple," OpenAI said. "Users feel more free to use ChatGPT when they know that they are in control of their personal information, including which conversations are retained and which are not."*
>
> *It's unclear if OpenAI will be able to get the judge to waver if oral arguments are scheduled.*
>
> *Wang previously justified the broad order partly due to the news organizations' claim that "the volume of deleted conversations is significant." She suggested that OpenAI could have taken steps to anonymize the chat logs but chose not to, only making an argument for why it "would not" be able to segregate data, rather than explaining why it "can't."*
>
> *Spokespersons for OpenAI and The New York Times' legal team declined Ars' request to comment on the ongoing multi-district litigation.*

So, that's a mess. The bottom line is that for the time being, and since this began, no one's ChatGPT logs have actually been deleted. They've been forced by court order to retain everyone's everything. I don't mean to make more of this than it is. I'm not suggesting that we should be terrified. I have no doubt that ChatGPT will treat them with as much respect as possible. But "deleted" doesn't actually mean truly gone. So if you are someone who cares about maintain as much absolute privacy as possible, you'll want to look at something such as venice.ai whose entire architecture is designed in TNO-mode so that they never have any logs to either keep or delete.

Erlang/OTP SSH library CVSS 10.0!!
CVE-2025-32433 was just posted as I was wrapping up the show notes so not much is known about it. But the attention-grabbing feature is its extremely rare CVSS of 10.0. It earned this due to the fact that it allows for unauthenticated remote code execution on all affected systems. The official CVE description reads:

> *Erlang/OTP is a set of libraries for the Erlang programming language. Prior to versions OTP-27.3.3, OTP-26.2.5.11, and OTP-25.3.2.20, an SSH server may allow an attacker to perform unauthenticated remote code execution (RCE). By exploiting a flaw in SSH protocol message handling, a malicious actor could gain unauthorized access to affected systems and execute arbitrary commands without valid credentials. A temporary workaround involves disabling the SSH server or to prevent access via firewall rules.*

Even though no one talks about using Erlang, apparently it’s out there. And any CVSS of 10.0 in remote SSH authentication deserves a mention. So I’m sure you know if this is something that affects you.

Can Russia “intercept” Telegram messages?

There’s a report that appears to allege that Russia now has some means for intercepting Telegram messages. My most pressing question is whether this applies to 2-party one-to-one messages. Here’s what the reporting says:

> *Russian human rights NGO “First Department” warned on Friday that Russia’s Federal Security Service (FSB) has learned to intercept messages sent by Russians to bots or feedback accounts associated with certain Ukrainian Telegram channels, potentially exposing anyone communicating with such outlets to treason charges.*
>
> *Russia’s principal domestic intelligence agency has gained access to correspondence made with Ukrainian Telegram channels including Crimean Wind and Vision Vishnun, according to First Department, which said that the FSB’s hacking of Ukrainian Telegram channels had come about during a 2022 investigation into the Ukrainian intelligence agencies “gathering information that threatens the security of the Russian Federation” via messengers and social networks including Telegram.*
>
> *The case is being handled by the FSB’s investigative department, though no suspects or defendants have been named in the case, according to First Department. When the FSB identifies individual Russian citizens who have communicated with or transmitted funds to certain Ukrainian Telegram channels, it contacts the FSB office in their region, which then typically opens a criminal case for treason against the implicated person.*
>
> *First Department said “We know that by the time the defendants in cases of ‘state treason’ are detained, the FSB is already in possession of their correspondence. And the fact that neither defendants nor a lawyer are named in the main case allows the FSB to hide how exactly it goes about gaining access to that correspondence.”*
>
> *First Department stressed that their findings highlighted the various security risks inherent in using Telegram for confidential communication, especially in cases where the contents of such private messages could result in criminal charges.*
>
> *Dmitry Zair-Bek, the head of First Division, said that materials from Telegram have already been used as evidence in “a significant number of cases”, adding that “in most cases, they have been accessed due to compromised devices. … However, there are also cases in which no credible technical explanations consistent with known access methods can be identified. This could indicate either the use of undisclosed cyber espionage tools or Telegram’s cooperation with the Russian authorities, obvious signs of which we see in a number of other areas.”*

We’ve been watching Pavel Durov’s previously adamant stance soften somewhat over time, particularly after he was arrested and convicted in France last summer. Has he allowed Telegram to be compromised? It’s certainly not a messaging system that can be trusted. And remember that an audit of its home-grown crypto technology raised additional concerns several months ago.

Spain blocks Google

I had to double-check the date on this news when I read that Spanish ISPs had accidentally blocked Google domains while attempting to crack down on illegal soccer live streams. The double-check was required, of course, because this is not the first time this has happened, nor the first time we've noted what a lame and hairbrained approach it is to force ISPs to locally filter large chunks of the Internet for (only) their subscribers. Maybe someday we'll learn... but I'm not holding my breath.

Reddit sues Anthropic

Reddit has sued Anthropic for scrapping and using Reddit comments to train its Claude AI chatbot.

Twitter's new encrypted DMs aren't better than the old ones

A recent analysis of Twitter's new encrypted "XChat" messaging appears to leave much to be desired. The researcher who looked into it wrote:

> *When Twitter launched encrypted DMs a couple of years ago, it was the worst kind of end-to-end encrypted - technically E2EE, but in a way that made it relatively easy for Twitter to inject new encryption keys and get everyone's messages anyway. It was also lacking a whole bunch of features such as sending pictures, so the entire thing was largely a waste of time. But a couple of days ago, Elon announced the arrival of "XChat", a new encrypted message platform <quote> "built on Rust with (Bitcoin style) encryption, whole new architecture." Maybe this time they've got it right?*
>
> *The TL;DR is: No. Use Signal. Twitter can probably obtain your private keys, and admit that they can MITM you and have full access to your metadata.*

The analysis goes deeper and it might make for a deeper dive on the podcast, so I may return to that next week. In the meantime, I'd follow this investigator's recommendation.

More "ThunderMail"

Meanwhile, "ThunderMail" – the worst named service ever – will have email servers located in the European Union for increased privacy. Yeah. Okay. Fine. Whatever. But could you change the damn name?

Login.gov backup issues

In other happy news, a GAO report incidentally noted that the Login.gov service has no policy to test its backups. So a cyberattack, a mistake, or any other IT issue could completely crash the US government's entire login and identity system for days, weeks, or even months until it is restored.

The Illusion of Thinking

https://ml-site.cdn-apple.com/papers/the-illusion-of-thinking.pdf

A couple of days ago I added an “AI” group to GRC’s long-running text-only NNTP newsgroups. In my inaugural post to that group, I wrote:

> *Everyone, I've learned not to haphazardly create groups that do not have enduring value, since it's more difficult to remove groups than to create them... and endless group proliferation is not ideal. But I think it's WAY beyond clear that Artificial Intelligence is in the process of rapidly changing the world, and I cannot imagine any more important and worthwhile new group to create.*

Then, this past Sunday, upon discovering this just-released research from Apple, thanks to feedback from “Urs Rau”, one of our listeners, I posted the following into this new AI newsgroup:

> *“The Illusion of Thinking” is how the title of their well-assembled paper begins. The entire title is: “The Illusion of Thinking: Understanding the Strengths and Limitations of Reasoning Models via the Lens of Problem Complexity”*
>
> *Is this just sour grapes, engendered by Apple finding themselves behind the rest of the industry in AI deployment? I don't think so. This looks like an exploration that adds to our understanding of what we have today. And it's not suggesting that what we have today is not useful, nor that Apple might not wish they had some of their own. What it's doing is exploring the LIMITS of what we are now calling “Artificial Intelligence” and suggesting what many of us have intuited, which is that while a massive problem space* ***can*** *be solved with powerful pattern matching, when there are not patterns to be matched, today's systems are revealed to not be exhibiting anything like true problem understanding.*

In other words, Leo, your earliest take on all this, which was that AI was little more than fancy spelling correction, carried an essential kernel of truth onto which Apple has just placed a very fine point. Everyone should listen carefully to what Apple’s research paper Abstract explains:

> *Recent generations of frontier language models have introduced Large Reasoning Models (LRMs) that generate detailed thinking processes before providing answers. While these models demonstrate improved performance on reasoning benchmarks, their fundamental capabilities, scaling properties, and limitations remain insufficiently understood.*
>
> *Current evaluations primarily focus on established mathematical and coding benchmarks, emphasizing final answer accuracy. However, this evaluation paradigm often suffers from data contamination and does not provide insights into the reasoning traces’ structure and quality.*
>
> *In this work, we systematically investigate these gaps with the help of controllable puzzle environments that allow precise manipulation of compositional complexity while maintaining consistent logical structures. This setup enables the analysis of not only final answers but also the internal reasoning traces, offering insights into how LRMs “think”. Through extensive experimentation across diverse puzzles, we show that frontier LRMs face a complete accuracy collapse beyond certain complexities. Moreover, they exhibit a counterintuitive scaling limit: their reasoning effort increases with problem complexity up to a point, then declines despite having an adequate token budget.*

> *By comparing LRMs with their standard LLM counterparts under equivalent inference compute, we identify three performance regimes: (1) low-complexity tasks where standard models surprisingly outperform LRMs, (2) medium-complexity tasks where additional thinking in LRMs demonstrates advantage, and (3) high-complexity tasks where* ***both*** *models experience complete collapse.*
>
> *We found that LRMs have limitations in exact computation: they fail to use explicit algorithms and reason inconsistently across puzzles. We also investigate the reasoning traces in more depth, studying the patterns of explored solutions and analyzing the models' computational behavior, shedding light on their strengths, limitations, and ultimately raising crucial questions about their true reasoning capabilities.*

As I've cautioned before, anything and everything that's believed to be known about AI definitely needs to carry a date stamp and also probably a "best if used by" expiration date. What this means for us here is that Apple is showing us some interesting and probably previously under-appreciated features of today's LRMs – Large Reasoning Models. It's worth reminding ourselves that if Apple had written this same paper a year ago, before the appearance of LRMs and only challenging LLMs, the results would have been similar, though significantly less impressive for the AI side.

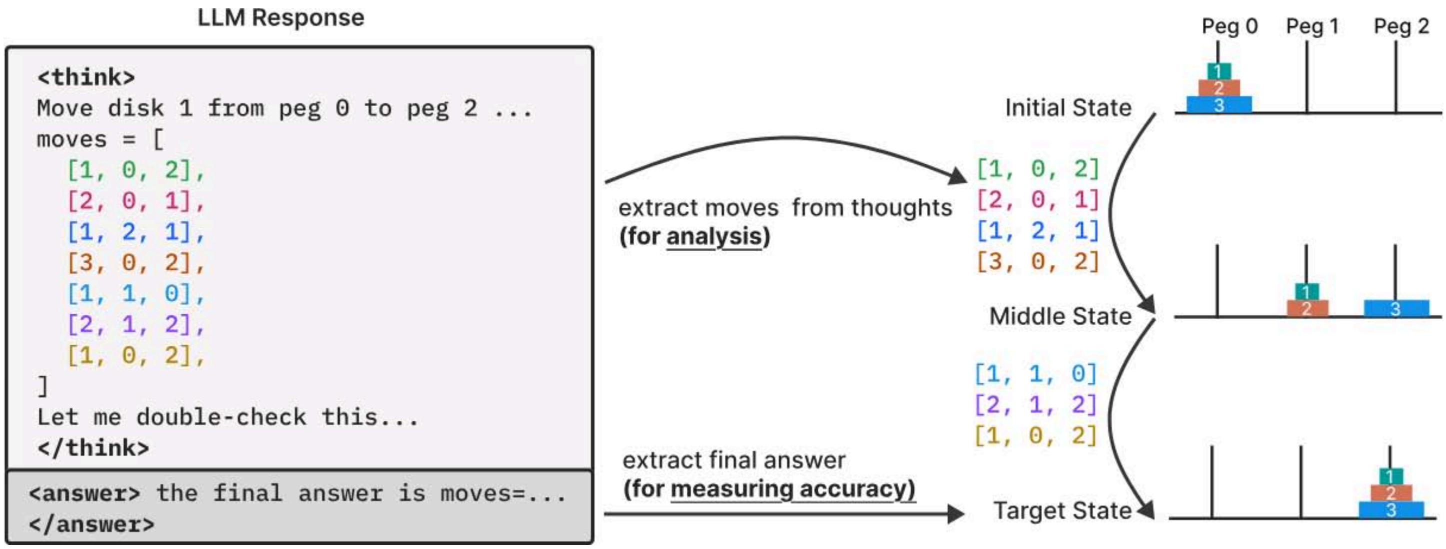
The question, then, is whether, and if so to what degree, even LARGER Reasoning Models in the future will be able to eclipse the performance of today's LRMs? In other words, since what we all want to know today is what's going to happen with AI in the future, to what degree is Apple's research able to speak to any fundamental underlying limitations that might limit any future AI?

To answer that question we need to see what Apple's research discovered. Here's how Apple's researchers set up the question:

> *Large Language Models (LLMs) have recently evolved to include specialized variants explicitly designed for reasoning tasks — Large Reasoning Models (LRMs) such as OpenAI's o1/o3, DeepSeek-R1, Claude 3.7 Sonnet Thinking, and Gemini Thinking. These models are new artifacts, characterized by their "thinking" mechanisms such as long Chain-of-Thought (CoT) with self-reflection, and have demonstrated promising results across various reasoning benchmarks. Their emergence suggests a potential paradigm shift in how LLM systems approach complex reasoning and problem-solving tasks, with some researchers proposing them as significant steps toward more general artificial intelligence capabilities.*
>
> *Despite these claims and performance advancements, the fundamental benefits and limitations of LRMs remain insufficiently understood. Critical questions still persist: Are these models capable of generalizable reasoning, or are they leveraging different forms of pattern matching? How does their performance scale with increasing problem complexity? How do they compare to their non-thinking standard LLM counterparts when provided with the same inference token compute? Most importantly, what are the inherent limitations of current reasoning approaches, and what improvements might be necessary to advance toward more robust reasoning capabilities?*
>
> *We believe the lack of systematic analyses investigating these questions is due to limitations in current evaluation paradigms. Existing evaluations predominantly focus on established mathematical and coding benchmarks, which, while valuable, often suffer from data contamination issues and do not allow for controlled experimental conditions across different*

settings and complexities. Moreover, these evaluations do not provide insights into the structure and quality of reasoning traces. To understand the reasoning behavior of these models more rigorously, we need environments that enable controlled experimentation.

In this study, we probe the reasoning mechanisms of frontier LRMs through the lens of problem complexity. Rather than standard benchmarks (e.g., math problems), we adopt controllable puzzle environments that let us vary complexity systematically—by adjusting puzzle elements while preserving the core logic—and inspect both solutions and internal reasoning.



```
<think>
Move disk 1 from peg 0 to peg 2 ...
moves = [
  [1, 0, 2],
  [2, 0, 1],
  [1, 2, 1],
  [3, 0, 2],
  [1, 1, 0],
  [2, 1, 2],
  [1, 0, 2],
]
Let me double-check this...
</think>
<answer> the final answer is moves=...
</answer>
```

We then see the paper's diagram of one of the puzzle tests Apple's researchers chose, which is the famous Towers of Hanoi. This is a classic puzzle with very simple rules. I received a beautiful wooden version one Christmas as a child from that annoying Aunt of mine who was always trying to stump me. For those who are not familiar, the puzzle consists of three pegs in a line, with one of the end pegs having a stack of discs of decreasing diameter, with the largest disc on the bottom. The challenge is to move all of the discs from the starting peg to the peg at the other end by moving one disc at a time from any peg to any other peg while never placing a larger disc over a smaller disc. It's a truly lovely puzzle because the rules are simple, but the solution requires patience, repetition, and grasping a deeper solution concept.

I should note that the puzzle is also a joy to solve by computer using traditional coding methods and that the most elegant coding solution employs recursion, since this puzzle itself is deeply recursive. For anyone who has an age-appropriate child – or nephew! – Amazon has a large selection of beautifully rendered wooden and colorful versions of this famous puzzle.

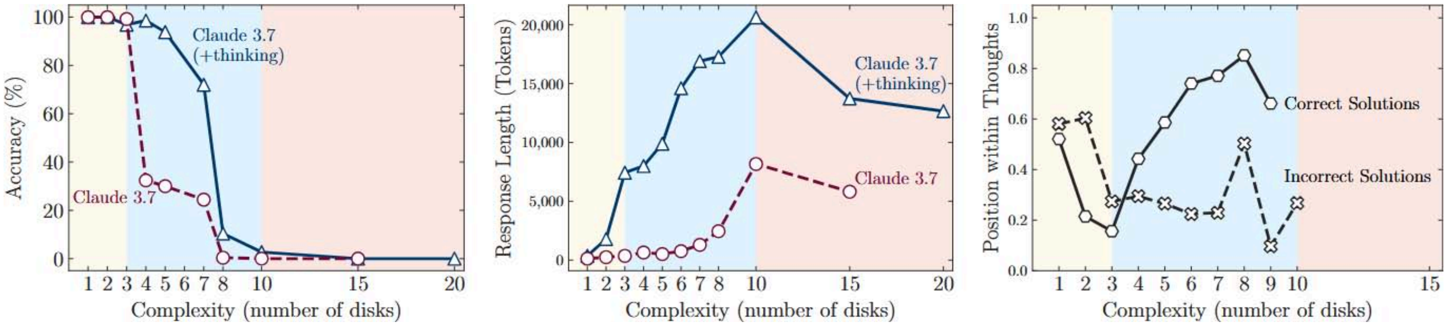
What's so clever about Apple's choice of this puzzle is that its complexity can be uniformly scaled simply by changing the number of discs. If we have one disc, we can simply move it to its destination ped. If we have two, the smaller top disc must first be placed on the middle peg, so that the bottom larger disc can be placed on its destination peg at the other end, then the smaller disc can join the larger disc on the end peg and the 2-disc puzzle is solved. Switching to 3 discs requires a bit more work: Visualize three peg and three discs. The smallest disc temporarily goes onto the 3rd destination peg. The middle disc goes to the middle peg. Now the smallest disc can go on top of the middle disc on the middle peg. This frees up the 3rd peg to receive the largest bottom disc which is now all alone on the original peg. The middle size disc is then moved to the first peg to uncover the middle-size disc, which can now be placed onto the

3rd destination page, and the smallest disc can then join the others to complete the stack and solve the 3-disc puzzle. It’s quite satisfying. And note that the 2 versus 3 disc puzzle may hopefully teach the astute puzzler which peg should first receive the smallest disc, based upon whether the disc count is even or odd. And that would be confirmed by the 4-disc puzzle.

I should mention that if anyone listening is planning to make a gift of one of these, please encourage its recipient to start out this way, rather than just jumping into a very frustrating deep end using all of the eight or ten discs that these puzzles provide. Solving the puzzle with very few discs will provide the encouragement and stamina that will be needed to eventually tackle and solve the very gratifying full puzzle... and that little trick about noticing which peg to start out with will definitely save the day.

So I think that Apple’s choice of The Towers of Hanoi is brilliant by reason of the puzzle’s lovely scalability of difficulty. In all, they used four different somewhat similar sequential combinatorial puzzles: Towers of Hanoi, Checker Jumping, Block World and River Crossing. Here’s what Apple explained:

> *These puzzles: (1) offer fine-grained control over complexity; (2) avoid contamination common in established benchmarks; (3) require only the explicitly provided rules, emphasizing algorithmic reasoning; and (4) support rigorous, simulator-based evaluation, enabling precise solution checks and detailed failure analyses.*
>
> *Our empirical investigation reveals several key findings about current Language Reasoning Models (LRMs): First, despite their sophisticated self-reflection mechanisms learned through reinforcement learning, these models fail to develop generalizable problem-solving capabilities for planning tasks, with performance collapsing to **zero** beyond a certain complexity threshold.*
>
> *Second, our comparison between LRMs and standard LLMs under equivalent inference compute reveals three distinct reasoning regimes.*



> *For simpler, low-compositional problems, standard LLMs demonstrate greater efficiency and accuracy. As problem complexity moderately increases, thinking models gain an advantage. However, when problems reach high complexity with longer compositional depth, both model types experience complete performance collapse (above, left chart).*
>
> *Notably, near this collapse point, LRMs begin reducing their reasoning effort (measured by inference-time tokens) as problem complexity increases, despite operating well below generation length limits (above, middle). This suggests a fundamental inference time scaling limitation in LRMs’ reasoning capabilities relative to problem complexity.*

Finally, our analysis of intermediate reasoning traces or thoughts reveals complexity-dependent patterns: In simpler problems, reasoning models often identify correct solutions early but inefficiently continue exploring incorrect alternatives—an "overthinking" phenomenon. At moderate complexity, correct solutions emerge only after extensive exploration of incorrect paths. And beyond a certain complexity threshold, models completely fail to find correct solutions (above, right).

This indicates LRMs possess limited self-correction capabilities that, while valuable, reveal fundamental inefficiencies and clear scaling limitations. These findings highlight both the strengths and limitations of existing LRMs, raising questions about the nature of reasoning in these systems with important implications for their design and deployment.

Our key contributions are:

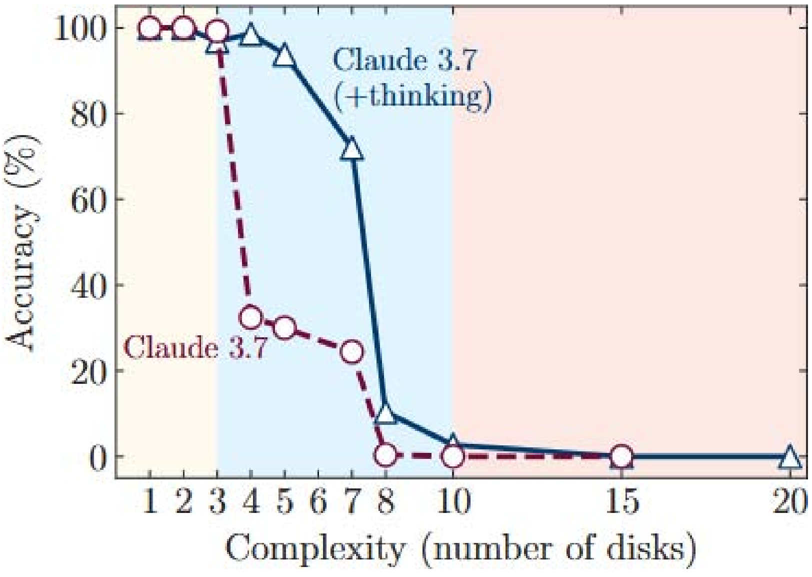
- *We question the current evaluation paradigm of LRMs on established math benchmarks and design a controlled experimental testbed by leveraging algorithmic puzzle environments that enable controllable experimentation with respect to problem complexity.*

- *We show that state-of-the-art LRMs (o3-mini, DeepSeek-R1, Claude-3.7-Sonnet-Thinking) still fail to develop generalizable problem-solving capabilities, with accuracy ultimately collapsing to **zero** beyond certain complexities across different environments.*

- *We find that there exists a scaling limit in the LRMs' reasoning effort with respect to problem complexity, evidenced by the counterintuitive decreasing trend in the thinking tokens after a complexity point.*

- *We question the current evaluation paradigm based on final accuracy and extend our evaluation to intermediate solutions of thinking traces with the help of deterministic puzzle simulators. Our analysis reveals that as problem complexity increases, correct solutions systematically emerge at later positions in thinking compared to incorrect ones, providing quantitative insights into the self-correction mechanisms within LRMs.*

- *We uncover surprising limitations in LRMs' ability to perform exact computation, including their failure to benefit from explicit algorithms and their inconsistent reasoning across puzzle types.*



For those listening to this without the advantage of the performance charts in the show notes, the Claude 3.7 thinking vs non-thinking model performance on the Towers of Hanoi puzzle was interesting.

Both the earlier Large Language Model and the later Large Reasoning Models performed perfectly, returning success 100% of the time when only 1 or 2 discs were used. Both models still did very well after a 3rd disc was added, but interestingly, the fancier thinking model underperformed the simpler LLM by about 4 percent. But when that first peg was stacked with 4 discs, the deeper thinking model's performance was restored whereas the simpler Claude 3.7 LLM collapsed to only finding the solution 35% of the time. As the disc count increases, both models' performance continues to drop, but the LRM holds a huge lead over the LLM until they get to 8 discs. The LLM is never able to solve that one whereas the thinking model finds the 8-disc solution about 1 out of every 10 tries, and 10 discs is beyond the reach of either.

The full research paper has lots of interesting detail about the various models' performance on the four puzzle types. I noted, however, that the nature of the other three puzzles seemed to be pretty much beyond the grasp of any of this so-called "AI".

One of their more interesting findings was the appearance of what they term the three complexity regimes. Paraphrasing from their paper, they wrote:

> *How Does Complexity Affect Reasoning?*
>
> *Motivated by the observations to systematically investigate the impact of problem complexity on reasoning behavior, we conducted experiments comparing thinking and non-thinking model pairs across our controlled puzzle environments. Our analysis focused on matched pairs of LLMs with identical model backbones, specifically Claude-3.7-Sonnet (w. vs. w/o thinking) and DeepSeek (R1 vs. V3). For each puzzle, we vary the complexity by manipulating problem size N (representing disk count, checker count, block count, or crossing elements).*
>
> *Results from these experiments demonstrate that, unlike observations from math, there exists three regimes in the behavior of these models with respect to complexity:*
>
> 1. *In the first regime where problem complexity is low, we observe that non-thinking models are capable of obtaining performance comparable to, or even better than thinking models with more token-efficient inference.*
>
> 2. *In the second regime with medium complexity, the advantage of reasoning models capable of generating long chain-of-thought begin to manifest, and the performance gap between model pairs increases.*
>
> 3. *The most interesting regime is the third regime where problem complexity is higher and the performance of both models have collapsed to zero. Results show that while thinking models delay this collapse, they also ultimately encounter the same fundamental limitations as their non-thinking counterparts.*

I think it's important to address their decision to use puzzles as an evaluation mechanism versus math problems. They gave this a lot of thought, and wrote on the "Math and Puzzle Environments" the following:

> *Currently, it is not clear whether the performance enhancements observed in recent*

> *reinforcement learning (RL)-based thinking models (all of the LRMs we've been talking about) are attributable to increased exposure to established mathematical benchmark data, to the significantly greater inference compute allocated to thinking tokens, or to reasoning capabilities developed by RL-based training?*
>
> *Recent studies have explored this question with established math benchmarks by comparing the upper-bound capabilities of RL-based thinking models with their non-thinking standard LLM counterparts. They have shown that under equivalent inference token budgets, non-thinking LLMs can eventually reach performance comparable to thinking models on benchmarks like MATH500 and AIME24. We also conducted our comparative analysis of frontier LRMs like Claude-3.7-Sonnet (with vs. without thinking) and DeepSeek (R1 vs. V3).*
>
> *Our results confirm that, on the MATH500 dataset, the performance of thinking models is comparable to their non-thinking counterparts when provided with the same inference token budget. However, we observed that this performance gap widens on the AIME24 benchmark and widens further on AIME25.*
>
> *This widening gap presents an interpretive challenge. It could be attributed to either: (1) increasing complexity requiring more sophisticated reasoning processes, thus revealing genuine advantages of the thinking models for more complex problems, or (2) reduced data contamination in newer benchmarks (particularly AIME25). Interestingly, human performance on AIME25 was actually* ***higher*** *than on AIME24, suggesting that AIME25 might be* ***less*** *complex. Yet models perform worse on AIME25 than AIME24—potentially suggesting data contamination during the training of frontier LRMs.*
>
> *Given these non-justified observations and the fact that mathematical benchmarks do not allow for controlled manipulation of problem complexity, we turned to puzzle environments that enable more precise and systematic experimentation.*

So we have the very real problem of data contamination, meaning that the models may have previously encountered the problems during their training and simply memorized the answer. So they're not actually reasoning, thinking and solving new problems, they're pattern-matching and regurgitating. But even puzzles like the Towers and Hanoi and River Crossing exist on the Internet and are also presumably in the training data. The researchers talk about this under the heading "Open Questions: Puzzling Behavior of Reasoning Models", writing:

> *We present surprising results concerning the limitations of reasoning models in executing exact problem-solving steps, as well as demonstrating different behaviors of the models based on the number of moves.*
>
> *In the Tower of Hanoi environment, even when we provide the algorithm in the prompt—so that the model only needs to execute the prescribed steps—performance does not improve, and the observed collapse still occurs at roughly the same point. This is noteworthy because finding and devising a solution should require substantially more computation (e.g., for search and verification) than merely executing a given algorithm. This further highlights the limitations of reasoning models in verification and in following logical steps to solve a problem, suggesting that further research is needed to understand the symbolic manipulation capabilities of such models.*
>
> *Moreover, we observe very different behavior from the Claude 3.7 Sonnet thinking model. In the Tower of Hanoi environment, the model's first error in the proposed solution often occurs*

> *much later, e.g., around move 100 for (N=10 discs), compared to the River Crossing environment, where the model can only produce a valid solution until move 4. Note that this model also achieves near-perfect accuracy when solving the Tower of Hanoi with (N=5), which requires 31 moves, while it fails to solve the River Crossing puzzle when (N=3), which has a solution of only 11 moves. This likely suggests that examples of River Crossing with N>2 are scarce on the web, meaning LRMs may not have frequently encountered or memorized such instances during training.*

This work by Apple's researchers is full of terrific insights that I want to commend to anyone who's interested in obtaining a more thorough understanding of where things probably stand at this point in time. Here's what the researchers conclude:

> *In this paper, we systematically examine frontier Large Reasoning Models (LRMs) through the lens of problem complexity using controllable puzzle environments. Our findings reveal fundamental limitations in current models: despite sophisticated self-reflection mechanisms, these models fail to develop* ***generalizable*** *reasoning capabilities beyond certain complexity thresholds.*

I'm going to repeat that since I think that's the essence of this entire paper:

"Our findings reveal that despite sophisticated self-reflection mechanisms, these models fail to develop ***generalizable*** *reasoning capabilities beyond certain complexity thresholds."*

So, these models are much better at doing what their simpler LLM brethren have been doing, but the difference is fundamentally quantitative not qualitative. Apple continues:

> *We identified three distinct reasoning regimes: standard LLMs outperform LRMs at low complexity, LRMs excel at moderate complexity, and both collapse at high complexity.*
>
> *Particularly concerning is the counterintuitive reduction in reasoning effort as problems approach critical complexity, suggesting an inherent compute scaling limit in LRMs.*
>
> *Our detailed analysis of reasoning traces further exposed complexity-dependent reasoning patterns, from inefficient "overthinking" on simpler problems to complete failure on complex ones. These insights challenge prevailing assumptions about LRM capabilities and suggest that current approaches may be encountering fundamental barriers to generalizable reasoning.*
>
> *Finally, we presented some surprising results on LRMs that lead to several open questions for future work. Most notably, we observed their limitations in performing exact computation; for example, when we provided the solution algorithm for the Tower of Hanoi to the models, their performance on this puzzle did not improve.*
>
> *Moreover, investigating the first failure move of the models revealed surprising behaviors. For instance, they could perform up to 100 correct moves in the Tower of Hanoi but fail to provide more than 5 correct moves in the River Crossing puzzle. We believe our results can pave the way for future investigations into the reasoning capabilities of these systems.*
>
> ***Limitations****: We acknowledge that our work has limitations. While our puzzle environments enable controlled experimentation with fine-grained control over problem complexity, they*

represent a narrow slice of reasoning tasks and may not capture the diversity of real-world or knowledge-intensive reasoning problems. It is notable that most of our experiments rely on black-box API access to the closed frontier LRMs, limiting our ability to analyze internal states or architectural components. Furthermore, the use of deterministic puzzle simulators assumes that reasoning can be perfectly validated step by step. However, in less structured domains, such precise validation may not be feasible, limiting the transferability of this analysis to other more generalizable reasoning.

So, in other words, the only thing this is, is what it is. It may or may not be more widely applicable and it may not even have any meaning or utility beyond the scope of these problems. There's not a great deal of real world need for stacking discs on poles. But for what it's worth, it does track with the intuition many of us have about where the true capabilities of today's AI falls.

Using terms like "comprehend" or "understand", or even "reason" clearly do not apply. They're used by AI fanboys. Maybe they're a lazy shorthand, but I don't feel they're helpful. In fact, I think they're anti-helpful. So what I think we need is some new anti-anthropomorphic terminology to accompany this new technology.

There is absolutely zero question that scale-driven computation has changed the world forever. Everyone is asking ChatGPT, and other consumer AI, more and more questions everyday, and that's only going to accelerate as the benefits of this become more widely known. AI does ***not*** need to become AGI or self aware to be useful and, frankly, I would strongly prefer that it didn't.

To that end, I doubt that we have anything to worry about anytime soon, and perhaps not even for the foreseeable future ... thus the title of today's podcast *"The Illusion of Thinking"* because I believe that the fairest conclusion is, that's all we have today. It's useful, but it's not *"thought."*