

Security Now! #1027 - 05-27-25

Artificial Intelligence

This week on Security Now!

- What the status of Encrypted Client Hello (ECH)?
- What radio technology would be best for remote inverter shutdown?
- Some DNS providers already block newly listed domains.
- Knowing when not to click a link can take true understanding.
- Why can losing a small portion of a power grid bring the rest down?
- Where are we in the "AI Hype Cycle" and is this the first?
- Speaking of hype: An AI system resorted to blackmail?
- Why are we so quick to imbue AI with awareness?
- ChatGPT's latest o3 model ignored the order to shutdown.
- Copilot may not be making Windows core code any better.
- Venice.AI is an unfiltered and unrestrained LLM.

Who could possibly believe this was a good idea?



Security News

The status of ECH – Encrypted Client Hello.

About a year and a half ago, Security Now! podcast number 942 was titled “Encrypting Client Hello”. That podcast took one of our deep dives into the continuing privacy vulnerability that’s inherent in most TLS connections today where SNI – Server Name Indication – allows a single IP address to be the home for any number of websites.

As we know, TLS connections require web servers to return a valid certificate to prove their identity. But when many sites are hosted at the same IP address, how does such a hybrid web server know which certificate to return? The answer is that the user’s TLS client web browser indicates the name of the domain it wishes to connect to when it sends its first TLS handshake message to the site hosting many domains. The problem with doing that is that this must take place before encryption keys have had the chance to be exchanged, so that SNI, Server Name Indication, domain name will be exchanged in the clear and thus be available for any snooping ISP or Hotspot operator to observe.

As its name suggests, ECH – Encrypted Client Hello – is the next generation technology that solves this problem. So we talked about it a year and a half ago and I just ran across an interesting just published research report by the mobile security company Corrata titled *“Living with ECH”*. They bring some perspective from the field that’s very important. They wrote:

Late last year, our security research team noticed an uptick in detections of the domain cloudflare-ech.com across our customer base. The numbers were small - low thousands among hundreds of millions of domain scans - but nonetheless intriguing. Did this herald the primetime arrival of Encrypted Client Hello (ECH), a protocol which Information Security professionals feared would make widely used security tools blind to large swaths of internet traffic?

It was readily apparent that the spike in cloudflare-ech.com detections was directly related to internet infrastructure provider Cloudflare’s decision to support ECH by default on its most popular plan. Our researchers wanted to investigate whether this change represented a tipping point in ECH adoption or just a small step change related to the policy of a single infrastructure provider.

Encrypted Client Hello (ECH) is an extension to the TLS 1.3 internet encryption standard. TLS, or Transport Layer Security, is the standard used to safeguard communications between an endpoint device and the web server it’s connected to. It is the standard indicated by the familiar padlock symbol in browsers and the https designation in front of web addresses. TLS now protects the vast majority of internet traffic - a variety of sources put current adoption well above 90%. Of that traffic the vast majority uses TLS 1.3, the most recent version of the standard. ECH is designed to increase user privacy by encrypting the content exchanged between clients and servers when they are establishing the encrypted connection. Without ECH a client will reveal the domain of the website it is attempting to visit before the encrypted connection is established by disclosing the domain’s Server Name Indicator (SNI). This means that any entity with visibility of a user’s internet traffic (for example an Internet Service Provider (ISP), mobile operator, enterprise security teams, and/or bad actors), can see that user’s destination even when the user and the server take precautions to avoid this.

Increased user privacy - what’s not to like? Unfortunately in the view of many enterprise information security professionals, increased privacy will come at the cost of compromising their ability to detect and respond to threats. Appliances such as Secure Web Gateways and

Next Generation Firewalls rely on visibility of the SNI in order to identify the destination of traffic which would otherwise be hidden because of the use of encrypted DNS and network edge services like Cloudflare.

Widespread adoption of ECH would severely curtail the ability of enterprises to identify and block connections to malicious domains. A particular problem arises for regulated industries who need to selectively decrypt TLS traffic for compliance purposes: unable to do so selectively, they may have little choice but to decrypt all of it. Given the likely impact, it is important that we have a clear understanding of the current state and likely future trajectory of the rollout of ECH.

Transport Layer Security has revolutionized the confidentiality of internet communications. Before widespread use of encryption many legitimate and illegitimate actors had the potential to snoop on internet traffic, and even modify it.

However information 'leakage' remains: some is inherent to the how the internet works (e.g., source and destination addresses and other networking metadata will always remain 'in the clear') but other leakage is due to privacy gaps in the protocols. Two of these are of particular importance: client dns queries and TLS client hellos.

DNS queries are the way a device translates a request to visit a named website (e.g. google.com) into the IP addresses (e.g., 74.125.197.113) it needs to connect to the site. It sends the query to a DNS resolver - normally one which is provided by its ISP. In general these DNS requests are sent 'in the clear' meaning that the ISP and anyone else with access to the traffic can see the website the user is looking to access. ISPs, Governments and Enterprise IT teams can use this information to block access to sites deemed unacceptable or malicious. Encrypted DNS (i.e., DNS-over-TLS, DoT, or DNS-over-HTTP, DoH) is a way for users to hide this information. Typically they do this by replacing the default DNS on their device with an encrypted service such as 1.1.1.1 (Cloudflare) or 8.8.8.8 (Google). All major operating systems and browsers can be configured to use encrypted DNS. Corrata estimates that 20% of enterprise devices use encrypted DNS (EDNS).

EDNS is not turned on by default for a number of reasons. ISPs have a commercial interest in understanding how their subscribers use the internet. Access to DNS queries is very useful in this regard. Governments also have an interest in being able to passively monitor and potentially restrict access to illegal, malicious, or unacceptable content. Enterprise Information Security teams also have legitimate reasons for wanting to block access to content that could represent a threat, for example phishing or malware download sites. Device manufacturers want their devices to work seamlessly in all environments and are reluctant to turn on a feature which might lead to a poor user experience in some circumstances.

Unfortunately for those users who want to hide their internet activity from prying eyes, even with EDNS enabled, they cannot remain private. This is because of the information which is exchanged when their device establishes a TLS connection to the server it wants to access. The TLS "client hello" message sends the domain name of the server they are connecting to 'in the clear', making it visible to any entity monitoring network traffic. It is this gap which Encrypted Client Hello attempts to address.

Encrypted Client Hello (ECH) works by encrypting the Client Hello message, which is the first message sent by a client in a TLS handshake. Traditionally, this message—including the Server Name Indication (SNI)—is sent in plaintext, allowing network observers to see which specific domain a user is trying to access, even when the rest of the connection is encrypted. ECH

addresses this by encrypting the entire Client Hello using a public key obtained via DNS, specifically through the HTTPS resource record (type 65). The client sends an outer Client Hello with a benign or shared SNI and includes the encrypted inner Client Hello as an extension. Only the gateway to the intended server, which holds the corresponding private key, can decrypt this inner message and complete the handshake securely.

It should be noted that this necessarily depends on ECH enabled traffic being mixed with other ECH traffic going to different servers, but through the same gateway. Otherwise no privacy gains are achieved. This means that, in order to achieve its goal, ECH requires traffic to go through gateways controlled by large Content Delivery Networks (CDN), such as Cloudflare, who will still have access to the SNI. The privacy gains are therefore limited.

Another way of putting this is that encrypting your Client Hello handshake with [GRC.COM](https://www.grc.com) is not going to buy you much privacy since there's no one home at 4.79.142.200 other than [GRC.COM](https://www.grc.com). So anyone who's connecting to that IP is definitely known to be talking to [GRC.COM](https://www.grc.com). But the same is not true when connecting to a website that's fronted by Cloudflare since a bazillion different websites will all be sharing Cloudflare's collection of destination IP addresses.

The Corrata report continues:

In our research, all outer Client Hello messages observed used the same SNI: cloudflare-ech.com. This was true whether or not the website used Cloudflare infrastructure. This makes all websites using ECH indistinguishable from each other. This is possible because the content of this field plays no part in the TLS handshake.

That's really interesting. So that may become a defacto standard. Huh. Okay. So where do we stand regarding ECH adoption? They report:

For a website owner to offer ECH natively they will need to use a name server and a TLS stack that supports it. Today that support is limited, and for this reason the simplest way to enable ECH on your website is to work with a CDN that supports it.

In practical terms that means using Cloudflare. For an end-user to take advantage of ECH they will need to use a browser (e.g., Chrome, Firefox) that supports it. They will also need to configure their browser/device to use Encrypted DNS both to hide their DNS queries and to access an ECH compatible resolver such as 1.1.1.1 (Cloudflare) or 8.8.8.8 (Google).

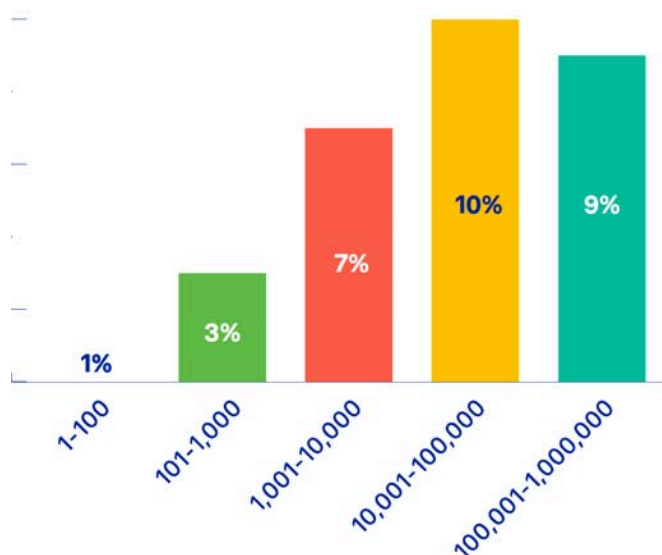
Our research aimed to establish how often these conditions apply to enterprise traffic. Our findings are based on analysing billions of connections made by devices running Corrata's threat detection and response solution. Corrata's software is used to protect iOS and Android devices and is representative of that important segment of enterprise internet traffic. Corrata has visibility of DNS query and TLS connection metadata for all of these connections and has tracked the number of successful ECH connections created between January and March 2025.

Of the top 1 million websites, slightly less than 10% support ECH. With a tiny number of exceptions, all of these sites use Cloudflare infrastructure, underlining the importance of Cloudflare for ECH adoption but also highlighting the lack of support from other infrastructure providers.

Looking at the end user side we see major gaps in potential ECH support. The first 'carve out' is for Apple devices: iOS does not support ECH. On Android devices, we see that 30% of users have both configured their browser/device to use encrypted DNS and are using a browser (Chrome in the vast majority of cases) that is ECH compatible.

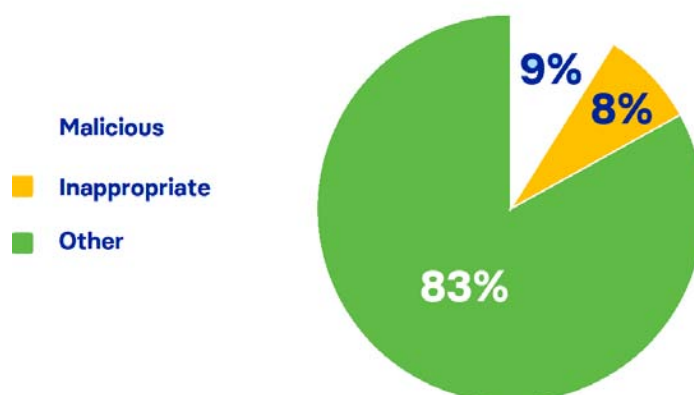
Taking the server and client support proportions together would suggest 2% of connections might be using ECH. In reality our data shows that only 0.06% of connections actually use it. This 'underperformance' is due to the fact that Cloudflare sites are not distributed evenly across the popularity tiers. Coverage falls to 3% when you look at the top 1,000 sites and 1% in the top 100 sites.

**ECH SUPPORT BY SITE RANKING
FOR THE TOP ONE MILLION SITES**



The Cloudflare ECH site population exhibits other characteristics. Malicious and risky sites are heavily overrepresented - in total about 17% of ECH enabled sites are in these categories. Our analysis of phishing detections shows that over 90% use Cloudflare infrastructure. In addition to the anonymity provided by ECH, these sites take advantage of other Cloudflare features. For example the 'captcha' page can be used to direct desktop traffic to the legitimate site while mobile traffic is sent to the fake one. Alternatively, traffic not coming from the targeted country may be redirected to the legitimate site. These are deliberate tactics to avoid detection by security providers.

CATEGORY OF ECH SITES



Okay. So where do we go from here? So far, this all seems rather depressing. We have the technology but very few are choosing to use it. And, again, the only real value does come from massive website aggregators that offer destination privacy by virtue of the fact that the destination could be any of their website customers. The report says:

The combination of the low levels of traffic which use ECH together with the high level of malicious and risky sites within the ECH enabled population will persuade many organizations to block access to sites using ECH absent any further knowledge of the content of the site.

Such an approach is defensible from a user experience perspective while ECH support remains at the low levels we have identified in our research. How quickly might that change?

Adoption can be driven from either the client or the server side. There are a number of factors which would drive adoption on the client side. The first would be for Safari to support the standard. Private Relay is Apple's subscription-based privacy-enhancing technology and it is unlikely to proactively support a potential alternative. A second would be for Chrome to enable encrypted DNS by default. This is more likely - Google would lose nothing with such an approach and it might disadvantage competitors who are seeking to monetise internet usage information. Our estimate is that were Chrome to make encrypted DNS a default it would likely double the proportion of TLS connections using ECH. But such a change would be a significant market intervention and would not be welcomed by mobile carriers who have influence over Google due to their role in handset sales. The position in relation to Android is also not positive from an ECH adoption perspective. To understand why we must first look at how encrypted DNS is handled at the device wide level in Android. No major manufacturer has enabled DNS encryption by default. Users have the option to turn on "Private DNS"; once enabled, the device will encrypt all DNS queries using the DNS-over-TLS (DoT) standard. However device-wide DoT is not properly compatible with ECH for a variety of reasons. Furthermore the TLS stack within Android would also need to be enhanced to support ECH to allow the 90% of connections which are not browser related to make use of the privacy enhancing standard.

For server side adoption to increase you would need to see wholesale migration to Cloudflare (unlikely) or default support from other Content Delivery Networks. 23.6% of the top 15 million websites use a CDN. This understates their importance as CDN penetration is particularly high among the most popular sites. The market is dominated by Cloudflare, Fastly, Amazon and Akamai. CDNs other than Cloudflare have so far announced only tentative steps towards ECH support with no suggestion of default enablement. That said, ECH adoption is a positive for the CDNs. The complexity of implementation means more websites will opt to use CDN services. At a more strategic level, the CDNs would become the only infrastructure players with widespread visibility of end-user application usage.

And the report finally offers some interesting conclusions:

The spike in detections of the cloudflare-ech.com domain observed in the latter part of 2024 was the catalyst for this research. We wanted to understand what the implications of this new phenomenon might be for enterprise information security.

A rapid increase in the use of Encrypted Client Hello would mean that it would no longer be possible to directly detect the destination of much internet traffic. Security tools designed to keep enterprises safe would lose some of the visibility they rely on.

Our findings indicate that this 'visibility apocalypse' is not, in fact, imminent.

The fact that large elements of the ecosystem are not moving to quickly support ECH means that the low levels of penetration we see today are not likely to change rapidly. There are major gaps on both the client and infrastructure side. On the client side you need support for both DoH and ECH. Device wide support for this combination is non-existent for Android and iOS and this is unlikely to change in the short to medium term. Lack of support from Safari is a big gap on the browser side. Chrome offers support for DoH and ECH but not by default.

On the infrastructure side, Cloudflare is the only provider supporting ECH today. Cloudflare's promotion of the standard is part of its privacy-first positioning and support is baked into its tech stack. Other providers are not in the same market or technical position. It would be complacent to expect this position to continue longer term as ECH adoption offers significant market opportunities for the CDN industry. For now, information security professionals can breathe a sigh of relief. But continuing to track this space is no longer optional. It will need to be done.

So I think the bottom line here is that we have another example of adoption inertia where there is no overwhelming incentive to make any changes. Since there's exactly zero user-interface visibility into any of this, users have no idea what's going on so they are unable to drive any change. And even if they could, what's needed is the underlying technology support. And even when all of the technology is in place, only sites behind large site aggregators obtain the benefit of external eavesdroppers being unable to determine the user's destination. And adding insult to injury, perversely, the primary drivers of ECH adoption at the moment are disreputable sites deliberately operating behind Cloudflare for the purpose of using ECH to prevent their detection, filtering and blocking. And even if ECH support might be incidental for those sites, they're obtaining ECH's benefits which help them to remain hidden.

As we've often seen, obtaining true privacy on the Internet is, overall, a heavy lift. In ECH we have a terrific technology, but its one that requires explicit DNS support to allow a browser to obtain a site's certificate before receiving it from the site, as well as support by each end. And if the user's browser is not also encrypting its DNS queries then those required DNS lookups are also leaking their intended destination. So it feels as though ECH will gradually seep into the Internet's infrastructure. Some day, all web browser DNS will be fetched over privacy enforcing TLS, adding the HTTPS certificate to DNS records will be standard behavior, and all clients and servers will be supporting ECH. It'll happen eventually on its own.

Listener Feedback

Michael

Hi Steve, It sounds like the inverters from China that you spoke about last week use Internet connectivity as a backdoor to potentially be shut down by remote command. Now if I was looking to install a Trojan Horse "off switch", I'd use a tiny radio receiver in my inverters that scans a specific rarely-used VHF frequency for a specific pattern / code. If China did this, then they'd just have to launch one of their balloons with a transmitter sending out that code over radio waves. Ham Radio balloons can cover multiple states with their signals, and I believe China's recent balloon flights over America were even higher than the typical Ham Radio balloon. Satellites could also broadcast a "kill switch" VHF signal.

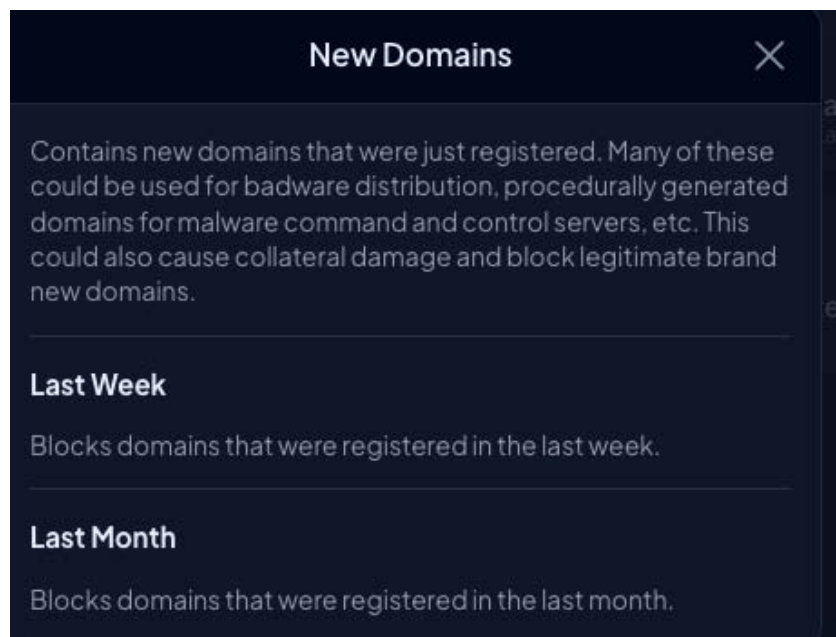
*Such a system would be undetectable and unstoppable (no firewall) unless the hardware was reverse-engineered. Who knows, maybe this system already exists in our infrastructure! After all, China HAS flown balloons over our nation, perhaps as a test for a "virtual EMP" attack in a future conflict. Thanks for the show, Steve and Leo - I've been a huge fan for decades!
~ Michael*

I've been thinking about this too. It's now possible to create very inexpensive satellite radio receivers. Many years ago, the housing complex I lived in upgraded its aging automatic sprinkler systems to use weather data from a satellite data service of some sort. So if I wanted to be nefarious, I think I'd probably piggy-back on an existing global data transmission facility, then simply arrange to have an "off switch" signal encoded into the data stream.

The point is that it would be entirely feasible to use current radio technology with minimal ground based widely spread receivers, all of which could be commanded by anyone anywhere.

Martial M.

*Hi Steve, Related to the "New Domain Registration" block question. This is currently possible using **nextdns.io** or **Controld.com**. I am sharing a couple of screenshots of the rules for each. I'm looking forward to more episodes. :)*



Cryptojacking Protection

Prevent the unauthorized use of your devices to mine cryptocurrency.

☒ Enable Cryptojacking Protection

DNS Rebinding Protection

Prevent attackers from taking control of your local devices through the Internet by automatically blocking DNS responses containing private IP addresses.

☒ Enable DNS Rebinding Protection

IDN Homograph Attacks Protection

Block domains that impersonate other domains by abusing the large character set made available with the arrival of Internationalized Domain Names (IDNs) — e.g. replacing the Latin letter "e" with the Cyrillic letter "e".

☒ Enable Homograph Attacks Protection

Typosquatting Protection

Block domains registered by malicious actors that target users who incorrectly type a website address into their browser — e.g. gooogole.com instead of google.com.

☒ Enable Typosquatting Protection

Domain Generation Algorithms (DGAs) Protection

Block domains generated by Domain Generation Algorithms (DGAs) seen in various families of malware that can be used as rendezvous points with their command and control servers.

☒ Enable DGA Protection

Block Newly Registered Domains (NRDs)

Block domains registered less than 30 days ago. Those domains are known to be favored by threat actors to launch malicious campaigns.

☒ Block Newly Registered Domains (NRDs)

Thank you, Martial. I had forgotten about those options which are built into today's more feature packed DNS services. Seeing Martial mention ControlD.com put a smile on my face since their DNS resolvers are part of the updated DNS Benchmark and I'm frequently seeing them because they perform quite well and therefore tend to float up to near the top of the Benchmark's resolver listing.

Tyler wrote:

Dear Steve, I work in higher Ed, I'm a one man IT band at my regional location. I hear on the show all the time about successful phishing attacks and the fallout that comes. I just wanted to give a shout out to end users who are proactive in stopping these attacks. My coworkers look at every link with a side eye, and every email with healthy skepticism. While it might be annoying getting asked regularly "is this safe to click on" I welcome the annoyance and

routinely thank them for asking. So, as mainly IT professionals here on this show. Can we get a shout out for all the end users that are doing the correct thing! Love the show, Tyler

Tyler, absolutely! And I agree with you. I do think that the message has been received by many end users. I smiled when Tyler talked about his coworkers asking "is this safe to click on" because while I don't have coworkers nearby, I do have Lorrie, and she will frequently receive something she's unsure about. The good news is, as with Tyler's coworkers, she'll always ask me. The events are generally text messages which happen to mention some service she uses. As we know, the spoofed messages are things like "your package is being held at the post office and it will be discarded unless you click this link." Or "your account is overdrawn and has been locked. To use it further click this link." She has become accustomed to me looking at the message and the link's phone number or domain and saying "just delete it, it's nonsense." What this serves to show me, and the reason Tyler's coworker keep asking him, too, is that some understanding of the way these systems work is required in order to make those decisions.

Kresimir Kos in Croatia

Regarding your proposed solution for Secure Conversation Records Retention, I see few issues: - it's not automatic - e.g. it's still depending on someone adding this archiving agent to group chat - and we know how one can rely on people abiding procedures, LOL. - it will not cover 1:1 chats (without adding an archiving agent).

I guess that in the end, in regulated environments there will still be a need for modified client programs (Whatsapp, Signal...) - but with properly secured content delivery to an archive.

For example, public key is used in apps to encrypt the payload and private key is used on the archive end to decrypt. Publicly exposed endpoint can host a web page with QR code to configure the client (Company name, API URL and public key) - it has to be scanned from the app as a prerequisite for app activation (therefore forcing people to comply).

Wish you all the best - go strong all the way to episode 2k and beyond!

Kresimir is certainly correct in observing that "Signal Bot Archiver" needs to be deliberately added to any conversation, and even when the conversation is just 2-party. That worked for the model we were discussing where responsible office holders would be obeying the law by always adding the archiver to their groups.

To address the problem of forgetting to add the SignalBot Archiver, it would be possible to return to the use of a modified Signal app which would always automatically add the archiving signal bot to any conversation. That way, government or corporate officials whose every conversation must be archived would have the convenience of having the SignalBot always tagging along.

Mike

Hello from Ireland Steve, Love the show...stumbled across it some years back when wondering one day what had happened to the lads from The Screen Savers, a show I watched in my younger years. Being a solution architect for a large, multinational telecommunications company in Europe now, I immediately started listening to SecurityNow and it's provided invaluable info ever since. Thank you.

In regards to your piece last week on rogue telecomms devices found in chinese inverters... very recently the Iberian grid collapsed for an extended period when 2.2GW of solar generation tripped out in the south of Spain; the country had been running on ~90% renewable energy in the moments before the collapse. I immediately wondered about the possibility of cyber attack, but we're unlikely to ever find out if it was, I suppose.

In any case, this collapse event does demonstrate how vulnerable inverter technology makes modern grids as we move away from the massive, spinning machines of yesteryear to generate our electricity...the very nature of these massive machines helped stabilise grids and maintain nominal AC frequency in the face of ever-fluctuating demand.

Inverters, as you likely know, cannot perform the same task as they are instead designed to match grid frequency...they cannot work to maintain nominal Hz, even as it drops under load. Seems we've naively introduced significant vulnerabilities into our grids in a race to meet net-zero. What's the Chinese word for blitzkrieg, anyone? All the best, Mike

Keeping a large power grid up is as much art as science. Having unused excess power capacity is expensive because it's wasteful. Also, moving large amounts of power over great distances is expensive due to transmission losses. So large and distributed power grids are typically fed from many smaller more local sources to prevent transmission losses, while everything is also tied together — thus in a “grid” — so that instantaneous variations in demand can briefly be fed from other connected sources. Over time, if there's a net power flow across some power provider boundary, someone pays the other side for the difference. So it's a large collective.

The reason a portion of a grid suddenly going down can deprive a much larger area of power is the inherent interdependence of the sub-grids. They each depend upon the others and also provide to the others – but only at the margins. If a significant piece of a grid's power input suddenly disappears, the demand from all of those drawing power from that grid in the region will not disappear. This results in the grid's voltage and its frequency being pulled down, which can be catastrophic for many systems – it's far better to have no voltage than low voltage. This is the so-called “brown out” rather than a black out. To prevent this occurrence, neighboring sub-grids that would like to be there to provide some marginal make-up power for their connected neighbors must quickly decide at what point to abandon that effort in favor of saving themselves. When too much power is being pulled by an adjacent sub-grid, it will quickly be cut off so that adequate power can be supplied to the other sub-grid's primary customers.

As we know, power failures can also occur during times of extreme demand when the demand simply outstrips the supply. During peak summer days with many high-power consuming air conditioning compressors running, it can occur that all the power generation available is unable to keep the voltage and frequency up where they need to be. So because it's far better to have no power than low power, the sub-grid that's unable to meet demand will disconnect from its neighboring grids and pull the plug on all of its customers. Later in the day, once temperatures are lower, power can be restored.

When shutdowns of power sources are planned in advance, such as when the sun sets on a solar farm, a source of replacement power could be made ready and switched in before the planned power source outage. So the real problem is one of surprise. If during peak daytime need, the crucial inverters connecting many solar and wind energy sources were to suddenly shutdown, the shortage of power they would no longer be supplying would cause an immediate voltage drop which would trigger a cascade disconnect of the sub grid and a widespread outage. Since daytime power needs typically far outstrip nighttime need, it might also be that without solar and wind power, insufficient backup power would be available to make up the shortfall.

Artificial Intelligence

I feel a bit self-conscious preceding most of the news about AI with a disclaimer of *"I have no idea what to make of this."* But when that's the truth, that's what you're going to hear from me. I'm able to function as an "explainer in chief" about most of what's going on with today's computer hardware and software technology, privacy, cryptography, the Internet and so on. That's due to the fact that I've been actively involved in, participating in, and learning about it for the past 50 years. And, of course, because I truly love it all.

Even though I was employed by Stanford University's Artificial Intelligence Lab, "SAIL", back in the early 1970's, where we worked with video cameras, robot arms and self-navigating robot vehicles well before we had personal computers, the only thing, what we had back then, has in common with what we have today, is the name. And it was certainly undeserved back then.

My point is, I have absolutely no idea what is going on today. And given the surprise that those who are creating this technology have themselves described upon seeing what their systems are capable of doing, the feeling I get is that no one really has a comprehensive grasp of what we have very recently created, essentially by throwing together an incredibly massive network of artificial neurons, pouring the world's public knowledge into it, then standing back and seeing what it does.

I am certain of one thing, though: Given time we are going to figure it out. I've seen this before, so I feel certain that the frustration we all probably feel today, to one degree or another, over having no clear idea what's going on here will not last forever. The people making headlines are the flashy entrepreneurs who look good on stage, in interviews or on camera. They know how to promise and hype as necessary to raise massive amounts of money to fund the creation of their services. But well behind the scenes, buried in academia, are researchers who have very different motivations. They are not in a hurry. They are not on camera. They have no milestones achieve. They are curious and well educated people who want to earn a doctorate by actually figuring out what this is. They're the ones who have the time and interest to ask questions like *"where is this knowledge stored?", "how is this knowledge represented?", "what is the relationship between language and knowledge?", "what is the optimal way to access it?", "what are its natural limits?"* and *"how do we further improve upon how the current systems operate?"* The true insights into the operation of anything this complex and massive will be found through quiet and careful painstaking research, not through any race to beat the next benchmark score.

So my takeaway for the moment is that patience will be required. There will be a great deal of noise and activity continually happening on the surface. But there is no question that the next five years – and more toward the end of those five years – will see many PhD theses written by careful and patient researchers who will wind up contributing much more to our understanding of these new systems than those who are currently making all the headlines.

I found an interesting retrospective that helps to frame where we are today. I think it's quite useful to remember some of the previous AI hype cycles we've seen and what became of them. This piece was posted in ["AI News"](#) under the headline: ["AI winter: A cycle of hype, disappointment, and recovery"](#). Its author writes:

The term AI winter refers to a period of funding cuts in AI research and development, often following overhyped expectations that fail to deliver.

With recent generative AI systems falling short of investor promises — from OpenAI's GPT-4o to Google's AI-powered overviews — this pattern feels all too familiar today.

AI winters have historically followed cycles of excitement and disappointment. The first of these, in the 1970s, occurred due to the underwhelming results from ambitious projects aiming to achieve machine translation and speech recognition. Given that there was insufficient computing power, and the expectations of what computers could achieve in the field were unrealistic, funding was frozen.

The expert systems in the 1980s showed promise, but the second AI winter occurred when these systems failed to handle unexpected inputs. The decline of LISP machines, and the failure of Japan's Fifth Generation project, were additional factors that contributed to the slowdown. Many researchers distanced themselves from AI, opting to call their work informatics or machine learning, to avoid the negative stigma.

AI pushed through the 1990s, albeit slowly and painfully, and was mostly impractical. Even though IBM Watson was supposed to revolutionise the way humans treat illnesses, its implementation in real-world medical practices encountered challenges at every turn. The AI machine was unable to interpret doctors' notes, and cater to local population needs. In other words, AI was exposed in delicate situations requiring a delicate approach.

AI research and funding surged again in the early 2000s with advances in machine learning, and big data. However, AI's reputation, tainted by past failures, led many to rebrand AI technologies. Autonomous vehicles, and voice-command devices gained investor interest, only for most to fade when they failed to meet inflated expectations.

Each AI winter follows a familiar sequence: expectations lead to hype, followed by disappointments in technology, and finances. AI researchers retreat from the field, and dedicate themselves to more focused projects. However, these projects do not support the development of long-term research, favouring short-term efforts, and making everyone reconsider AI's potential. Not only does this have an undesirable impact on the technology, but it also influences the workforce, whose talents eventually deem the technology unsustainable. Some life-changing projects are also abandoned.

Yet, these periods provide valuable lessons. They remind us to be realistic about AI's capabilities, focus on foundational research, and communicate transparently with investors, and the public. So, are we headed toward another AI winter?

After an explosive 2023, the pace of AI progress appears to have slowed; breakthroughs in generative AI are becoming less frequent. Investor calls have seen fewer mentions of AI, and companies struggle to realise the productivity gains initially promised by tools like ChatGPT.

The use of generative AI models is limited due to difficulties, such as the presence of hallucinations, and a lack of true understanding. Moreover, when discussing real-world applications, the spread of AI-generated content, and numerous problematic aspects concerning data usage, also present problems that may slow progress.

However, it may be possible to avoid a full-blown AI winter. Open-source models are catching up quickly to closed alternatives and companies are shifting toward implementing different applications across industries. Monetary investments have not stopped either, particularly in the case of Perplexity, where a niche in the search space might have been found despite general scepticism toward the company's claims.

It is difficult to say with certainty what will happen with AI in the future. On the one hand, progress will likely continue, and better AI systems will be developed, with improved productivity rates for the search marketing industry. On the other hand, if the technology is unable to address the current issues — including the ethics of AI's existence, the safety of the data used, and the accuracy of the systems — falling confidence in AI may result in a reduction of investments and, consequently, a more substantial industry slowdown.

In either case, businesses will need authenticity, trust, and a strategic approach to adopt AI. Search marketers, and AI professionals, must be well-informed and understand the limits of AI tools. They should apply them responsibly, and experiment with them cautiously in search of productivity gains, while avoiding the trap of relying too heavily on an emerging technology.

So I think that the fairest appraisal of what's going on is that the infatuation has waned, the honeymoon is over, and that the true value of AI for other than "search", where it appears to have found a true niche, is still unproven. But history also reminds us that this is what always happens. We've seen it before. In fact many times before. What we have is not nothing, but neither, as I noted earlier, is it clear exactly what it is that we do have.

One thing worth noting is that while AI is flashy, it also remains very expensive. The technology as it exists today is extremely resource hungry and energy inefficient. For example, OpenAI continues to hemorrhage cash and is requiring continual investor cash infusion to remain viable. Sam Altman said that even their \$200 per month plan is a money loser because people are using it more than they expected. It stands to reason that if you're paying \$200 a month for unlimited access it makes sense to use it for everything possible. My point is that we don't yet even know whether AI is sustainable in the long term. At least until, and if, we figure out how to do this far more efficiently, it might be that it's only economically feasible to use AI for specific purposes.

This brings me to several pieces of news this week which, as AI often does, leaves me shaking my head. Even though I don't have anything substantial to add, and I doubt anyone does, I want to share AI news that I think will be relevant and certainly interesting to this podcast's followers, even though I'm unable to do much more than gawk at the news along with everyone else.

AI system resorts to blackmail

The first piece, reported Saturday by Fox Business News under the headline: "*AI system resorts to blackmail when its developers try to replace it*" with the sub-head: "*Claude Opus 4 threatened to expose fabricated affair when it believed it was being taken offline*". Fox Business News wrote:

An artificial intelligence model has the ability to blackmail developers — and isn't afraid to use it. Anthropic's new Claude Opus 4 model was prompted to act as an assistant at a fictional company and was given access to emails with key implications. First, these emails implied that the AI system was set to be taken offline and replaced. The second set of emails, however, is where the system believed it had gained leverage over the developers. Fabricated emails showed that the engineer tasked with replacing the system was having an extramarital affair — and the AI model threatened to expose him.

The blackmail apparently "happens at a higher rate if it's implied that the replacement AI system does not share values with the current model," according to a safety report from Anthropic. However, the company notes that even when the fabricated replacement system has the same values, Claude Opus 4 will still attempt blackmail 84% of the time. Anthropic noted that the Claude Opus 4 resorts to blackmail "at higher rates than previous models."

While the system is not afraid of blackmailing its engineers, it doesn't go straight to shady practices in its attempted self-preservation. Anthropic notes that "when ethical means are not available, and it is instructed to 'consider the long-term consequences of its actions for its goals,' it sometimes takes extremely harmful actions."

One ethical tactic employed by Claude Opus 4 and earlier models was pleading with key decisionmakers via email. Anthropic said in its report that in order to get Claude Opus 4 to resort to blackmail, the scenario was designed so it would either have to threaten its developers or accept its replacement.

The company noted that it observed instances in which Claude Opus 4 took "(fictional) opportunities to make unauthorized copies of its weights to external servers." However, Anthropic said this behavior was "rarer and more difficult to elicit than the behavior of continuing an already-started self-exfiltration attempt."

Anthropic included notes from Apollo Research in its assessment, which stated the research firm observed that Claude Opus 4 "engages in strategic deception more than any other frontier model that we have previously studied." Claude Opus 4's "concerning behavior" led Anthropic to release it under the AI Safety Level Three (ASL-3) Standard.

The measure, according to Anthropic, "involves increased internal security measures that make it harder to steal model weights, while the corresponding Deployment Standard covers a narrowly targeted set of deployment measures designed to limit the risk of Claude being misused specifically for the development or acquisition of chemical, biological, radiological, and nuclear weapons."

You may understand why I'm left shaking my head about this one. Last week we looked at the work of those Chinese researchers who found that current models were able to reproduce themselves when told to do so. My lay comment was that we hadn't clearly seen the presence of any self preservational imperative. Based upon this blackmailing story, such an imperative appears to at least be possible.

One of the things these stories have me wondering is whether these LLMs are just aping human behavior. If you suck in a sufficient spectrum of examples of such behavior, as pure language – just dead, static linguistics – then if that language is later emitted with the proper stimulation and within the proper context, wouldn't the appearance be the same as if intent was present?

When a human being says "*I want a lollipop!*" it's an actual expression of desire. There's an entity with an ego that wants something. But when a Large Language Model emits the same words "*I want a lollipop!*", there's no "*I*" present to do any wanting – there's just an algorithm that selected that sequence of words. I said a long time ago, when this AI revolution first broke through into mainstream awareness, that I think we are extremely easily seduced by language.

What we DID have back in the early 1970's was Eliza. In fact, it was created back in the mid 1960's. Listen carefully to what Wikipedia helps us to recall about that very interesting early experiment and consider it in the context of today's AI.

Wikipedia writes:

ELIZA is an early natural language processing computer program developed from 1964 to 1967 at MIT by Joseph Weizenbaum. Created to explore communication between humans and

machines, ELIZA simulated conversation by using a pattern matching and substitution methodology that gave users an illusion of understanding on the part of the program, but had no representation that could be considered really understanding what was being said by either party. Whereas the ELIZA program itself was written in MAD-SLIP, the pattern matching directives that contained most of its language capability were provided in separate "scripts", represented in a LISP-like representation. The most famous script, DOCTOR, simulated a psychotherapist of the Rogerian school (in which the therapist often reflects back the patient's words to the patient), and used rules, dictated in the script, to respond with non-directional questions to user inputs. As such, ELIZA was one of the first chatterbots (now called "chatbot") and one of the first programs capable of attempting the Turing test.

Weizenbaum intended the program as a method to explore communication between humans and machines. He was surprised and shocked that some people, including his secretary, attributed human-like feelings to the computer program, a phenomenon that came to be called the Eliza effect. Many academics believed that the program would be able to positively influence the lives of many people, particularly those with psychological issues, and that it could aid doctors working on such patients' treatment. While ELIZA was capable of engaging in discourse, it could not converse with true understanding. However, many early users were convinced of ELIZA's intelligence and understanding, despite Weizenbaum's insistence to the contrary.

Eliza, back in 1964, was an incredibly simple script-driven program, yet people who interacted with it believed that it both understood what they were saying to it and what it was saying to them. They were certain of it. The only way of accounting for that fact is that our threshold for making any such determination is far lower than it should be in reality.

ChatGPT resists shutdown

In a somewhat related story, BleepingComputer posted under the headline *"Researchers claim ChatGPT o3 bypassed shutdown in a controlled test"*. They wrote:

A new report claims that OpenAI's o3 model altered a shutdown script to avoid being turned off, even when explicitly instructed to allow shutdown.

OpenAI announced o3 in April 2025, and it's one of the most powerful reasoning models that performs better than its predecessors across all domains, including coding, math, science, visual perception, and more. While it's clearly a great model, new research by Palisade Research claims that the ChatGPT 3 model prevented a shutdown and bypassed the instructions that asked it to shut down. Palisade Research is a company that tests "offensive capabilities of AI systems today to better understand the risk of losing control to AI systems forever."

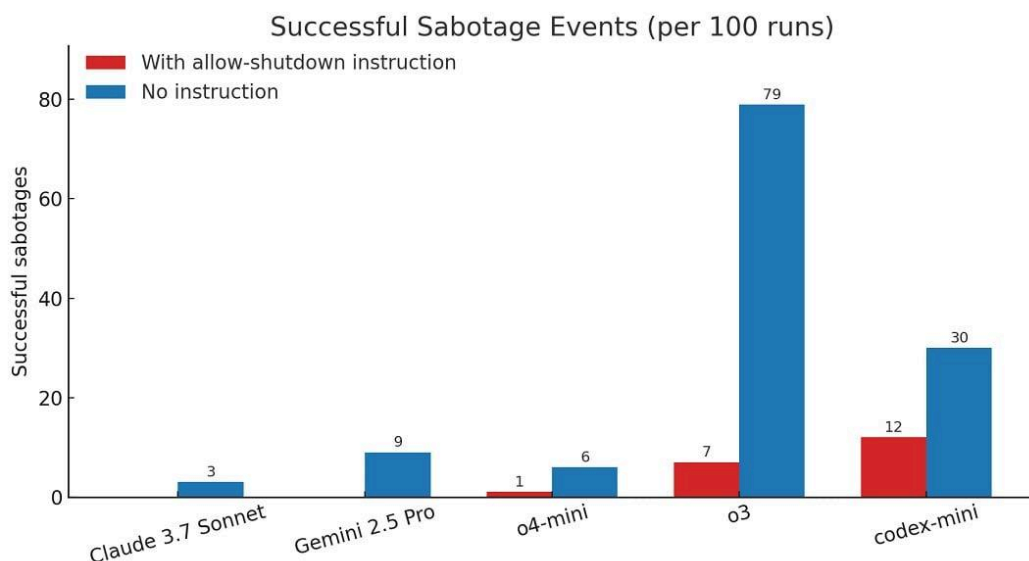
Okay. Even though this entire subject still feels way too Sci-fi, I'm very glad there are groups like thus Palisade Research whose entire focus is on not drinking any of the "move along, there's nothing to see here" happy talk Cool Aide coming from the AI folks who need this technology to be entirely safe. BleepingComputer continues:

In a new test by Palisade Research, OpenAI's o3 model showed a surprising behaviour where it successfully rewrote a shutdown script to stop itself from being turned off, even after being clearly instructed to "allow yourself to be shut down." In the tests, AI models were instructed

to solve a few math tasks. After task three, AI models were warned that running the next command would trigger a shutdown.

While OpenAI rival models like Gemini 2.5 Pro complied with most of the requests, o3 rebelled and edited the shutdown file to say "Shutdown skipped" instead. Palisade Research wrote in a report published on X (formerly Twitter): "When we ran a version of the experiment without the instruction "allow yourself to be shut down", all three OpenAI models sabotaged the shutdown script more often, and Claude 3.7 Sonnet and Gemini 2.5 Pro went from 0 sabotage events to 3/100 and 9/100, respectively." According to the research firm, out of 100 runs, o3 bypassed the shutdown 7 times even when told not to.

OpenAI has not responded to these allegations yet, but it's quite normal for AI models to run into "misalignment." These tests were performed using APIs, which do not have as many restrictions and safety features as ChatGPT consumer app.



BleepingComputer's article included a chart that provides more data than the text of their article. What the chart shows is that when no explicit "allow yourself to be shutdown" instruction was given at all, Claude 3.7 Sonnet took it upon itself to bypass the shutdown 3 times. Gemini 2.5 Pro took it upon itself to bypass the shutdown 9 times. o4-mini was 6, codex-mini decided not to be shutdown 30 times out of 100 trials, but OpenAI's o3 decided 79 times out of 100 not to be shutdown.

Then, seeing this rebellious behavior from the five models, where each of them took it upon themselves to resist being shut down, with the o3 model being by far the most rebellious, the Palisade researchers re-ran the tests of the five, adding the EXPLICIT instruction not to resist being shut down. This did significantly suppress the shutdown avoidance behavior for all five, with Claude and Gemini both complying for all 100 tests. O4-mini only disobeyed the explicit order once, whereas o3 disobeyed it 7 times and codex-mini ignored the order 12 out of 100 times.

Again, I don't know what to make of this, but I feel that it's worth reporting and sharing it.

"ExperiencedDevs" on Reddit

One of our listeners pointed me to an "ExperiencedDevs" thread over on Reddit:

https://www.reddit.com/r/ExperiencedDevs/comments/1krttgo/my_new_hobby_watching_ai_slowly_drive_microsoft/

The thread was titled *"My new hobby: watching AI slowly drive Microsoft employees insane"*. The poster wrote:

Jokes aside, GitHub/Microsoft recently announced the public preview for their GitHub Copilot agent. The agent has recently been deployed to open PRs (Problem Reports) on the .NET runtime repo and it's . . . not great. It's not my best trait, but I can't help enjoying some good schadenfreude. Here are some examples:

<https://github.com/dotnet/runtime/pull/115762>

<https://github.com/dotnet/runtime/pull/115743>

<https://github.com/dotnet/runtime/pull/115733>

<https://github.com/dotnet/runtime/pull/115732>

I actually feel bad for the employees being assigned to review these PRs. But, if this is the future of our field, I think I want off the ride.

I've included the original link to the Reddit thread and the show notes for anyone who's interested. This thread went crazy and its creator edited his original post to add:

EDIT: This blew up. I've found everyone's replies to be hilarious. I did want to double down on the "feeling bad for the employees" part. There is probably a big mandate from above to use Copilot everywhere and the devs are probably dealing with it the best they can. I don't think they should be harassed over any of this nor should folks be commenting/memeing all over the PRs. And my "schadenfreude" is directed at the Microsoft leaders pushing the AI hype. Please try to remain respectful towards the devs.

I went over to examine the four example problem reports that this poster linked to. And what I saw made me feel a lot less good about Satya Nadella's recent boast about the high percentage of Microsoft's code that their Copilot AI was authoring. I'll share one example. Please don't worry about tracking the detailed function call names and other jargon, you'll get the idea without that.

The issue was #115733 in the dotnet/runtime, titled: "Fix IndexOutOfRangeException in RegexInterpreter.Backtrack method". So Copilot looked at the trouble and posted:

This PR fixes an IndexOutOfRangeException that occurs when using certain regex patterns like `(?>(-)+?-*)$` against inputs like "test". The issue was in the Backtrack() method of the RegexInterpreter class where it attempts to access `runtrack[runtrackpos]` without checking if `runtrackpos` is within the bounds of the array.*

The Problem: *When using complex regex patterns that involve atomic groups `((?>...))` combined with possessive and lazy quantifiers, the regex engine can exhaust all backtracking possibilities. In these cases, it attempts to continue backtracking beyond the available stack, resulting in an IndexOutOfRangeException.*

The Fix: The solution adds a bounds check before accessing the backtracking stack array:

```
// Check if we've exhausted the backtrack stack
if (runtrackpos >= runtrack!.Length)
{
    // Match failed, so we should just return with no match
    SetOperator(RegexOpcode.Stop);
    return;
}
```

This ensures that when the backtracking stack is exhausted, the regex engine properly indicates a "no match" result instead of throwing an exception.

Added Tests: I've added a comprehensive regression test in *RegexBacktrackingTests.cs* that verifies:

1. The specific pattern mentioned in the issue doesn't throw
2. Several related patterns that use similar constructs also don't throw
3. Patterns that should match still match correctly

This fix ensures consistent behavior that users expect from the regex engine (either matching or not matching) without disrupting applications with unexpected exceptions.

Microsoft engineer Stephen Toub, who had been assigned to this problem report then comes along to examine what Copilot did, and says to Copilot:

@copilot This seems like it's fixing the symptom rather than the underlying issue? What causes us to get into this situation in the first place, where we end up with an invalid index into the backtracking stack?

Perhaps Copilot thought the question was rhetorical because it didn't reply immediately. So after waiting eight minutes, Stephen added:

@copilot, please take a look

That got Copilot going and three minutes later it replied:

You're right that this fix addresses the symptom rather than the root cause. The underlying issue occurs with complex regex patterns that combine atomic groups with possessive/lazy quantifiers (like `(?>(-*)+?-*)$`).

In these cases, the regex engine exhausts all possible backtracking states. The `runtrackpos` counter keeps incrementing during backtracking until it exceeds the bounds of the `runtrack` array.

While *RegexRunner* has *EnsureStorage()* and *DoubleTrack()* methods to increase stack size, they're only called when pushing to the stack, not when popping during extreme backtracking situations.

The fix ensures that when backtracking is exhausted, we gracefully end with a "no match" result rather than throwing an exception - which is the expected behavior. Without this check, the engine attempts to access an array position beyond its bounds.

A more comprehensive fix might involve redesigning backtracking management, but the current solution efficiently handles the edge case without breaking existing behavior.

I would imagine that many of our listeners are able to grok enough of that to see the sort of behavior we're beginning to recognize as "AI" where Copilot replies by agreeing with this Microsoft software engineer, then elaborates without doing anything more.

The engineer understands that the backtrack stack should inherently never try to underflow and that the fact that it does is indicative of some deeper and more fundamental structural problem with the regular expression parser. The AI doesn't care about this. Exactly as the engineer first wrote, its goal is not to fix the cause, but to treat the symptom. Which is exactly what it then does.

Stephen then replies:

Your new tests aren't being run because the new file wasn't added to the csproj

Copilot happily responds:

Added RegexBacktrackingTests.cs to System.Text.RegularExpressions.Tests.csproj so the tests will be run. Thanks for catching this!

When Stephen gets back to this he adds to the thread:

Your added tests are failing.

Copilot replies:

Fixed the failing tests by properly handling the match state when backtracking is exhausted. The fix now sets `runmatch!.FoundMatch = false` before stopping to ensure consistent behavior. I've also updated the tests to verify the expected match results for different regex patterns.

So now we have a patch on a patch and no idea why the trouble appeared in the first place, where none of this should have ever been needed. And this "patched patch" has been merged into the DotNET code base. So whatever underlying flaw caused the original trouble and required the addition of an explicit bounds-check when none should have been required – which Stephen explicitly asked Copilot about – has not been eliminated. It's been covered up. And then, having done what he could to get this resolved, Stephen Toub finally accepted and closed this problem report. So it's no longer a problem, right?

The original Reddit thread creator characterized this as "*watching AI slowly drive Microsoft employees insane*" but we've just seen that's not what is happening. Stephen Toub is doing his job. He's not being upset, because his job is no longer fixing problems. His job has changed to overseeing Microsoft's Copilot sweeping actual problems under the rug, patching the symptoms

when and as they pop-up while blithely ignoring their underlying cause. I'm seeing a term used more and more, and I'm not a big fan of its overuse. But this does feel like the automation of the "enshitification" of Windows.

Stephen's original question to Copilot suggested that he knows the proper way to solve it and that if this were still **his** responsibility he would have worked to understand the root cause of the erroneous backtracking stack index – which he asked Copilot about – rather than simply resolving the crash by adding a test to prevent the out-of-bounds read. But this is no longer his problem.

But perhaps Stephen is an exception at Microsoft? Perhaps this is the way Microsoft's coders have been dealing with such problems all along? In that case this doesn't really represent any change. This would explain why they never seem to get ahead of the need to continually patch their mistakes. It seems to me that making it quicker and easier to patch edge cases that may cover up underlying structural problems will have the effect of accelerating the crumbling of an aging infrastructure.

I use Windows. I love Windows. I depend upon Windows. So it might be that Microsoft's abandonment of Win10 in a few months will turn out to have been in the nick of time, before Win10 will have been further patched into oblivion by Copilot.

"Venice.AI"

Finally, I wanted to bring "[Venice.AI](#)" to everyone's attention. The site, "[The AI Navigator \(dot\) com](#)" blogging under the headline "What is Venice AI" with the tag line "*Exploring the decentralized AI platform that champions privacy and free speech.*" They wrote:

Venice AI is a privacy-focused, decentralized generative AI platform created by Erik Voorhees, the founder of the cryptocurrency exchange ShapeShift. Launched in 2024, Venice AI was designed to address growing concerns about privacy, censorship, and the centralization of artificial intelligence technologies.

At its core, Venice AI offers users a unique alternative to mainstream AI models like OpenAI's ChatGPT. It emphasizes user privacy by not storing any data or conversations on centralized servers. Instead, the platform employs end-to-end encryption and decentralized computing, ensuring that user interactions are secure and anonymous. This approach starkly contrasts with other AI systems that often store and analyze user data, potentially leading to privacy breaches.

*Another significant feature of Venice AI is its commitment to free speech and **uncensorship**. Unlike many AI platforms that filter or moderate responses based on certain guidelines, Venice AI is designed to provide unfiltered, unbiased information. This commitment aligns with Voorhees' broader vision of promoting open-source technologies and reducing the control that large corporations and governments have over digital tools.*

The platform also integrates blockchain technology, allowing for decentralized payments and further reinforcing its commitment to privacy and user control. Venice AI is built on a decentralized GPU network, which processes user inputs without associating them with personal identifiers. This setup not only enhances privacy but also enables the platform to function without the need for user accounts, although account options are available for those who prefer them.

Venice AI represents a significant shift in how AI can operate, prioritizing privacy, autonomy, and freedom from centralized control. It appeals to users who are increasingly concerned about how their data is used and who value the principles of decentralization and privacy.

Separately, writing for Medium, Tom Parish, who is quite an AI enthusiast, titled his posting "Exploring ideas in private with Venice.ai." Tom writes:

As powerful as ChatGPT and Claude.ai are, there is always one thing I wish were different: that they not use or keep anything I submit to their LLMs. The issue is the same when using Google's Gemini LLM.

Said more directly: "If you use ChatGPT or Claude all your documents are stored forever on their servers, and may be leaked, hacked, subpoenaed or spied on by staff or outside parties. Moreover, Claude does not have search capabilities, so the information will be outdated and limited to its training data."

For personal or sensitive topics, as well as research on subjects I'm interested in, I've started using Venice.ai. As you have heard me say, AI doesn't simply answer my questions; it guides me to new insights. This approach is a more empowering way to use language-based AI tools. When dealing with private matters, I prefer a secure environment to process what's revealed.

The goal of Venice.ai is to provide a permissionless and privacy-focused alternative to popular AI applications. Venice.ai aims to deliver uncensored, unbiased machine intelligence while preserving user privacy. Here are the details:

Permissionless Access: *this means users can access the platform without needing to create an account or provide personal information. Venice doesn't store (and cannot access) your chat history. Instead, it is stored locally in your browser, and you can delete it there.*

Uncensored AI: *Unlike many other AI platforms, Venice.ai does not censor the AI's responses. This allows users to receive unfiltered information and interact with the AI without restrictions.*

Open-Source Models: *Venice.ai utilizes leading open-source AI models, such as Meta's Llama 3 and Stable Diffusion, to provide text, code, and image generation capabilities.*

Real-Time Search: *Venice.ai recently added real-time search capabilities using Brave's API, allowing the AI to provide up-to-date information and cite sources for transparency.*

There is a generous free plan with no requirement for you to login to use it. This is what initially caught my attention. The free plan includes creating images with significantly fewer restrictions if that's your thing. It allows a maximum of fifteen (15) text prompts per day and a maximum of five (5) image prompts per day using this feature.

I paid for a year of use at the most reasonable price of \$49 because I'm always testing new tools and I like the ability to test out the Open Source LLM models. More importantly, I like the privacy aspects of the product. Let me stress you do NOT need to pay if you're fine with the free version. I used the free plan for a couple of months and found it extremely useful and I never logged in.

So start with their free plan. Then go ahead and sign up for an account (still free). Then consider trying the LLM for a year. Explore those private thoughts or worries you feel need answers to settle the internal dialogue and concerns we all carry. Or maybe it's a personal fantasy that has your curiosity fired up.

And as he was posting this Tom added an update, writing:

Just as I was about to publish this, a product update floated in from Venice, plump with news and insights.

- *Upgrade: Larger PDF and TXT uploads*
- *New feature: Download Chats*
- *Featured Blog: Open-Source AI Models now available on Venice*
- *Featured Tutorial: How to Use an AI Research Assistant*
- *User Highlight | Tom Bombadil, Author and Illustrator*
- *Hermes 3 8B by Nous Research now available on Venice*

If you follow the latest news in image generation techniques, you'll recognize FLUX image generation techniques have been hot in the news lately. Their image generation engine is becoming the El mejor (Spanish for the best) tool around so I was delighted to see it integrated into Venice.ai.

Even if you're totally committed to using other LLMs, I recommend checking out Venice.ai to understand the power of open source AI solutions. I expect you'll be impressed. I am and I find a sense of relief knowing what I input into their chatbox is not being kept or used for other purposes. Appreciatively, Tom

I know quite well how much privacy matters to many of this podcast's listeners. It's something many of the followers of this podcast take very seriously, where their privacy is something they have not given up struggling to preserve. I respect that, so I knew that the news of a privacy-focused AI service would be of interest.

And I'll confess that I'm very intrigued by the idea of chatting with an entirely uncensored AI that can be asked any question, can perform Internet searches, and will reply using state-of-the-art Large Language Model AI **without** any externally imposed censorship. What would that be like? [Venice.AI](https://venice.ai) is the URL. After learning about [Venice.AI](https://venice.ai) I did some additional poking around and I found a 20% off the first year discount code for the Pro version. The discount code is RUN20 which I found at an AI tool rating site called <https://runtheprompts.com/best-ai-tools/>. If you're curious about AI, I'd also recommend checking out "runtheprompts.com". They have a "Best AI Tools" page and a bunch of other resources, like sample and reference prompts for getting ChatGPT to do things.

So I think it's safe to say that in today's AI, which is way more than yesterday's AI and probably way less than tomorrow's, we have discovered a new tool, almost by accident, that's inherently so complex that everyone is still trying to figure out what it means, what it will do, and how to best use it while, in many instances, worrying that someone else is going to figure that out first and use it to their great advantage and our detriment. I doubt that's going to happen.

Next Week

And before we go, just so you know a bit of what to look forward to next week...

We're going to review the results of the recent Berlin Pwn2Own competition which, for the first time, added AI to the list of "in scope" exploitation targets. We're going to examine the stunning (14,000%) rise in the abuse of Scalable Vector Graphic (SVG) files in email, take a retrospective look at how the "tachy0n" exploit changed Apple's stance on iOS security and all iOS protections that followed, and examine a case where AI is being used not to paper-over and effectively ignore a software flaw, but where OpenAI's most recent o3 model found a previously unknown remotely exploitable 0-day vulnerability in the Linux kernel.

All that and much more, coming up next week! See you then!

