Security Now! #1001 - 11-19-24

Artificial General Intelligence (AGI)

This week on Security Now!

How Microsoft lured the US Government into a far deeper and expensive dependency upon its cybersecurity solutions. Gmail to offer native throwaway email aliases like Apple and Mozilla. Russia to ban several additional hosting companies and give its big Internet disconnect switch another test. Russia uses a diabolical Windows flaw to attack Ukrainians. The value of old Security Now episodes. TrueCrypt's successor. Using Cloudflare's Tunnel service for remote network access. How to make a local server appear to be on a remote public IP. How to share an 'impossible to type' password with someone. How to find obscure previous references in the Security Now podcast. What are the parameters for the expected and widely anticipated next generation Artificial General Intelligence (AGI)? What do those in the industry and academia expect? And is OpenAI's Sam Altman completely nuts for predicting it next year? Is it just a stock ploy?

The bottom of this staircase may have been blocked.
But these future hackers are not deterred!

Security News

Free Candy

Some recent reporting by ProPublica raises some interesting questions...

> *In the summer of 2021, President Joe Biden summoned the CEOs of the nation’s biggest tech companies to the White House. A series of cyberattacks linked to Russia, China and Iran had left the government reeling, and the administration had asked the heads of Microsoft, Amazon, Apple, Google and others to offer concrete commitments to help the U.S. bolster its defenses. Biden told the executives gathered in the East Room: “You have the power, the capacity and the responsibility, I believe, to raise the bar on cybersecurity.”*
>
> *Microsoft had more to prove than most. Its own security lapses had contributed to some of the incursions that had prompted the summit in the first place, such as the SolarWinds attack, in which Russian state-sponsored hackers stole sensitive data from federal agencies, including the National Nuclear Security Administration. Following the discovery of that breach, some members of Congress said the company should provide better cybersecurity for its customers. Others went further. Senator Ron Wyden, who chairs the Senate’s finance committee, called on the government to “reevaluate its dependence on Microsoft” before awarding it any more contracts.*

Good thought, Ron. However, what happened next was not, I’m pretty certain, the sort of “reevaluation” Ron had in mind...

> *In response to the president’s call for help, Microsoft CEO Satya Nadella pledged to **give** the government $150 million in technical services to help upgrade its digital security. On the surface, it seemed a political win for the Biden administration and an instance of routine damage control from the world’s largest software company. But the result of ProPublica’s subsequent investigation suggests that Microsoft’s seemingly straightforward commitment to provide a bunch of free technical services belied a more complex, profit-driven agenda. As time has since revealed, Microsoft’s apparent generosity was a calculated business maneuver designed to bring in **billions** of dollars in new revenue, lock competitors out of lucrative government contracts and even further tighten the company’s grip on federal business.*

If I didn’t know better I would think that Gates was still around since this turned out to be a recognizably classic “Bill move.”

> ***The White House Offer**, as it was known inside Microsoft, would dispatch Microsoft consultants across the federal government to install Microsoft’s cybersecurity products — which, as a part of the offer, were provided free of charge for a limited time.*

That’s right! What a bargain! What’s wrong with this picture?

> *Well, how about once the consultants installed the upgrades, federal customers would be effectively locked in, because shifting to a competitor after the free trial would be cumbersome and costly, according to former Microsoft employees involved in the effort, most of whom spoke on the condition of anonymity because they feared professional repercussions. At that point, the customer would have little choice but to pay for the higher subscription fees.*

> *In fact, two former sales leaders involved in the effort likened it to a drug dealer hooking a user with free samples. “If we give you the crack, and you take the crack, you’ll enjoy the crack,” one said. “And then when it comes time for us to take the crack away, your end users will say, ‘Don’t take it away from me.’ And you’ll be forced to pay me.”*
>
> *Former salespeople said that Microsoft wanted more than those subscription fees. The White House Offer would lead customers to buy other Microsoft products that ran on Azure, the company’s cloud platform. This carried additional charges based on how much storage space and computing power the customer used. These former salespeople said that the expectation was that the upgrades would ultimately “spin the meter” for Azure, helping Microsoft take market share from its main cloud rival, Amazon Web Services.*
>
> *In the years after Nadella made his commitment to Biden, Microsoft’s goals became reality. The Department of Defense, which had resisted the upgrades for years due to their steep cost, began paying for them once the free trial ended, laying the groundwork for future Azure consumption. So did many other civilian agencies. Former Microsoft salesperson Karan Sondhi who had knowledge of the deals said that “The White House Offer” got the government hooked on Azure, “And it was successful beyond what any of us could have imagined.”*
>
> *While Microsoft’s gambit paid off handsomely for the company, legal experts told ProPublica the White House Offer should have never come to pass, as they sidestep or even possibly violate federal laws that regulate government procurement. Such laws generally bar gifts from contractors and require open competition for federal business.*
>
> *Eve Lyon, an attorney who worked for four decades as a procurement specialist in the federal government said that accepting free product upgrades and consulting services collectively worth hundreds of millions of dollars is not like a free sample at Costco, where I can take a sample, say, ‘Thanks for the snack,’ and go on my merry way. Here, you have changed the IT culture, and it would cost a lot of money to switch to another system.”*
>
> *Microsoft, for its part, defended its conduct. Steve Faehl, the security leader for Microsoft’s federal business, said in a statement: “The company’s sole goal during this period was to support an urgent request by the Administration to enhance the security posture of federal agencies who were continuously being targeted by sophisticated nation-state threat actors. There was no guarantee that agencies would purchase these licenses and they were free to engage with other vendors to support their security needs.”*
>
> *Pricing for Microsoft’s security suite was transparent, he said, and the company worked “closely with the Administration to ensure any service and support agreements were pursued ethically and in full compliance with federal laws and regulations.” Faehl said in the statement that Microsoft asked the White House to “review the deal for antitrust concerns and ensure everything was proper and they did so.”*

There’s only one little problem with this. It really is surprisingly difficult to switch vendors. And, of course, it gets worse...

> *The White House summit ushered in a new form of concentrated reliance, as well as the kind of anticompetitive behavior the Biden administration has pledged to stamp out. Former Microsoft salespeople told ProPublica that during their White House Offer push, they advised federal departments to save money by dropping cybersecurity products they had purchased from competitors. Those products, they told them, were now “redundant.” Salespeople also*

> *fended off new competitors by explaining to federal customers that most of the cybersecurity tools they needed were included in the free upgrade bundle.*
>
> *Today, as a result of the deals, vast swaths of the federal government, including all of the military services in the Defense Department, are more reliant than ever on a single company to meet their IT needs. ProPublica's investigation, supported by interviews with eight former Microsoft employees who were involved in the White House Offer, reveals for the first time how this sweeping transformation came to be — a change that critics say leaves Washington vulnerable, the very opposite of what Biden had set out to achieve with his summit.*
>
> *"How did Microsoft become so pervasive of a player in the government?" asked a former Microsoft sales lead? "Well, the government let themselves get coerced into Microsoft when Microsoft rolled the stuff out for free."*

https://www.propublica.org/article/microsoft-white-house-offer-cybersecurity-biden-nadella

The ProPublica article goes into much greater detail and I have a link to it in the show notes for anyone who wants more. But I'm sure that all of our listeners get the idea. At one point Microsoft was asked to provide this enhanced security support to the federal government at no change indefinitely – which they flatly declined. Then, of course, it became a negotiation over how long the services would be free. What adds even more salt to this wound is that for many years these same federal and military agencies had been steadfastly refusing to go with Microsoft solutions due to their cost. But they couldn't say "no" to "free"... so this allowed Microsoft to get their solutions in the door, to remove any previous "reasonably priced" competitive solutions, and then, once the free offer expired, the choice was either pay up or go without.

It's at least mildly disgusting. And what's more, this didn't just fall into Microsoft's lap. Former insiders made clear that this was the intention all along, from the beginning. Microsoft's CEO Satya Nadella knew exactly what he was doing. It was a Trojan horse.

Gmail to add "Shielded Email"

Apple has "Hide My Email" and Mozilla offers "Firefox Relay". These are email services that create throwaway aliases for a user's primary account. The recent news is that Google is reportedly working on something they plan to call "Shielded Email" which will do the same for their 2 Billion Gmail users. As with the other services, users can generate random-looking usernames for use in online forms which will hide their real email addresses. Then, if those aliases ever start receiving spam, those temporary addresses can be deleted.

I've noticed that a large percentage of the subscribers to GRC's emailing lists are gmail domain users. So I imagine this will come as a welcome new service.

Meanwhile, over in Russia

Russian officials have recently announced via Telegram that they plan to expand Russia's ban on foreign web hosting providers that are hosting content that discredits the glorious Russian Army (their words). So Akamai and CDN77 may soon find themselves added to the ban list.

Overall, Russia appears to feel that the Internet is at best a mixed blessing. It's unclear to me how it's possible to function within today's globalized economy without it. But Russia seems poised to at least explore that. To which end, Russia's illustrious Internet watchdog, none other than Roskomnadzor, has announced its plan to conduct another test next month of Russia's big

Internet disconnect switch which, when pulled, does what it says: It severs all ties between Russia and the rest of the global Internet. One wonders about Starlink, though.

And while we're on the topic of Russian antics:
One of the 0-days (CVE-2024-43451) that Microsoft patched this week was used in a Russian hack of Ukrainian organizations earlier this year. According to the security firm ClearSky, the 0-day was part of an exploit chain that exposed NT LANMAN credential hashes when victims interacted with .URL files that were received in phishing emails.

But here's the part that really caught my attention: ClearSky said that right-clicking, deleting, or moving the file established a connection with the attacker's server, exposing authentication data. The report suggests that the campaign also used social engineering to convince victims to run executables.

Now hold on here. Right-clicking on a file to display its context menu and examine its properties, deleting it or dragging it to another directory was all that's needed to cause the victim's machine to establish a remote connection to a malicious server? What?! So I went over to ClearSky to see what was up. https://www.clearskysec.com/0d-vulnerability-exploited-in-the_wild/

The ClearSky Research Team posted their write-up last Wednesday. They wrote:

> *A new zero-day vulnerability, CVE-2024-43451, was discovered by ClearSky Cyber Security in June 2024. This vulnerability affects Windows systems and is being actively exploited in attacks against Ukrainian entities. The vulnerability activates URL files containing malicious code through seemingly innocuous actions:*
>
> - *A single right-click on the file (in all Windows versions).*
> - *Deleting the file (in Windows 10 or 11).*
> - *Dragging the file to another folder (in Windows 10 or 11 and some Windows 7/8/8.1.)*
>
> *The malicious URL files were disguised as academic certificates and were initially observed being distributed from a compromised official Ukrainian government website.*
>
> *The attack begins with a phishing email sent from a compromised Ukrainian government server. The email prompts the recipient to renew their academic certificate. The email contains a malicious URL file. When the user interacts with the URL file by right-clicking, deleting, or moving it, the vulnerability is triggered. This action establishes a connection with the attacker's server and downloads further malicious files, including SparkRAT malware.*
>
> *SparkRAT is an open-source remote access trojan that allows the attacker to gain control of the victim's system. The attackers also employed techniques to maintain persistence on the infected system, ensuring their access even after a reboot.*

Okay. so the culprit here is a .URL file, which is a Windows Internet URL shortcut file. It's a text file in the format of an original .INI file with [sections] called out in square brackets and various name=value pairs. The key is that file contains a URL= line where the scheme of the URL is "file://" followed by the IP of the malicious remote server. In Windows, the FILE:// scheme is handed by SMB where SMB stands for "Server Message Blocks" – which underlies Windows' original file and printer sharing. So that's where NTLM credential hashes come in. Apparently upon even extremely innocuous contact with these files in Windows – and it's worse in more recent Windows 10 and 100 – Windows Explorer will, without prompting, reach out to the file server that's indicated in the shortcut even without its recipient executing the shortcut. The researchers wrote:

> *When examining the URL file, ClearSky's team exposed a new vulnerability: Right clicking the file establishes a connection to an external server. In addition, execution in a sandbox raised an alert about an attempt to pass the NTLM (NT Lan Manager) Hash through the SMB protocol. After receiving the NTLM Hash, an attacker can carry out a Pass-the-Hash attack to identify* ***as*** *the user associated with the captured hash without needing the corresponding password.*

In other words, the credential hash that NTLM's SMB protocol sends out to identify its Windows user can simply be captured and subsequently used to impersonate that user as if they were logged in. The researchers wrote:

> *Further investigation yielded that in Windows 10 and 11 operating systems, the action of dragging the file from one folder to another, or deleting the file, causes the file to communicate with the target server and only then be deleted or moved. Under Windows 7, 8, and 8.1, the file did not initiate communication when dragged or deleted, unless the target folder was open at the time of dragging (this did not happen on the first attempt but was observed only after 2-3 attempts). That is, the newly detected vulnerability is more exploitable on Windows 10 and 11 operating systems.*

I'm sure that it must be a bit unnerving to those old pros among our listeners to learn that the actions that any of us might take to dispose of something we may have inadvertently received could themselves lead directly to a compromise of our machines. Yikes!

Microsoft reportedly patched and closed this flaw in last Tuesday's patch updates. So that's good. But it should serve to remind us that those of us using Windows are using an extremely complex operating system that's still dragging a ton of legacy code forward. That code was written, and its protocols were designed, long before the world had an appreciation for just how secure our future systems would need to be.

The classic example of this was the original design of the Windows metafile format. Windows draws on the screen through a series of drawing primitives – circle, rectangle, line, and so forth. A Windows metafile (WMF) is just the capture of those drawing primitives. Then later, when that metafile is opened, those primitives are replayed onto a new blank canvas to recreate the original drawing. So, the metafile contents are interpreted. But the designers of the original metafile format thought: What if we want to do something that's more than just replaying something that was previously recorded? Why can't the file contain some code that's executed? And remember, this was Windows 3.0. So among all of the interpreted tokens they specified a META_ESCAPE code that would cause the system to execute the code contained within the Windows metafile starting with the bytes immediately following the special escape code.

And so it sat there in the metafile specification for years until much later, in the era of NT and networking and the Internet it was suddenly "rediscovered" and labeled as a horrible exploitable flaw. When I calmly stated that it was obviously there all along by design, many people misunderstood me. They thought I was saying that Microsoft had deliberately planted a backdoor in Windows metafiles. It was originally deliberate but it was never malicious. It was a reasonable thing to do back when we could trust every image our machines might try to render. But let's just say that it didn't age well. And neither has Microsoft's original NT LAN Manager and their SMB protocol. They were also designed back before we really understood security.

Closing The Loop

Ayiko Fred from Uganda:

> *Hey Steve and Leo,*
>
> *This is Ayiko Fred from Uganda. I've been listening to Security Now since 2021, starting around the 800s. I occasionally miss a few episodes when things get busy (sometimes up to a month), but I'm thoroughly enjoying the show!*
>
> *I don't have a formal background in CS, but I developed an interest in programming in 2020 and learned some Erlang and Elixir (my first and only languages), which I'm now using at work. It made me realize I had only a blurry understanding of many key concepts. I'd never thought to go back to the earlier episodes from 2005, but a few episodes ago, a listener recommended going back to the earlier episodes. So, I decided to give it a try—and wow! The way you explain topics like how the internet works, cryptography, and VPNs really clicked for me. I was blown away by how much easier it was to understand these concepts through your explanations. Now I feel like I've been "programming by superstition" all along. Each episode has left me wanting more, and I've even re-listened to some episodes 3-4 times, especially those on cryptography and internet fundamentals. I'm now on episode 58 and I'd encourage anyone with a shaky grasp of these topics to check out the earlier episodes—they won't regret it.*
>
> *One episode that made me think, "This is exactly what I needed!" was Episode 41, "TrueCrypt." Unfortunately, I learned that TrueCrypt's development was discontinued in 2014. Do you have any recommendations for alternative tools with similar features to TrueCrypt that are compatible with Linux? I'd love something with the same level of privacy and security.*
>
> *Thank you again for all your work—I really appreciate it! Looking forward to episode 1000.*
>
> *Best regards, Ayiko Fred*

This is the bit of feedback I mentioned last week. I wanted to share this because I know that this podcast has been discovered by many people years after we recorded those early fundamental technology podcasts. We've heard from others who, after discovering this podcast, had the idea of going back to the start to catch up. And those people have invariably found that it was worth their time. So part of me is tempted to just stop and recreate some of that work from the early days so that they're put back into everyone's feeds. But that doesn't make sense because they're already there – every podcast we've ever recorded remains available to everyone – and reproducing content we've already created would displace our new content for which we often barely have time as it is.

So, from time to time I'll take a moment, as I have here, to remind our listeners that back in the early days we laid down many of the fundamentals of the way everything we're talking about today, works. And it was done in a way that many people found to be extremely accessible.

Also, another thing we often hear is that while our listeners enjoy today's content they feel that there's much that they don't understand. It is true that I consciously build upon the foundation of what has come before. That's the only way for us to move forward. So to those who feel that they've been tossed into the deep end of the pool by showing up here late, let me note that all of that knowledge that's missing and assumed **was** once covered in detail back in the earlier days of this podcast. And it’s all still there, waiting and free for the asking for anyone who wants it.

At the end of his note, Ayiko noted that he had stumbled upon our discussion of TrueCrypt back in episode 41 and was disappointed to learn that it was no more. So I wanted to point him to TrueCrypt’s successor, VeraCrypt. VeraCrypt remains very much alive and well to this day. I have a link to it in today’s show notes: https://www.veracrypt.fr/en/Home.html

Scott Gottfried wrote to share his powerful solution for accessing his network from home. Scott leaves to the end that everything he describes is all a free service provided by Cloudflare.
I wanted to mention it up front so that while I’m sharing what Scott wrote everyone who might have a similar need will be taking it seriously. Scott wrote:

> *Hi Steve, Congrats on #1,000! I’ve listened for all 20 years — every episode. Thank you (and Leo)!*
>
> *I’ve heard several questions from your listeners about how to access their home network while traveling. VPN? Overlay network? I had this same question. My primary requirement for accessing my home network was that I didn’t want to open any ports on my router. I researched solutions for several months until I happened upon a blog post at Cloudflare. The solution for me is the Cloudflare Tunnel: https://www.cloudflare.com/products/tunnel/*
>
> *I run an old Intel NUC from inside my network that creates an outgoing tunnel to Cloudflare. The Cloudflare dashboard lets me add my own domains, has a firewall, provides authentication and allows me to configure routing for my 4 internal (home) subnets. It’s awesome. I run 2 separate photo sharing apps for the family. The apps run in Docker containers on the NUC which has Linux and CasaOS. But the tunnel could run on a NAS or Zimaboard.*
>
> *When traveling, I use the Cloudflare Warp app on my laptop and connect to my home network. I can then RDP to my Windows NUC. I can access my Ubiquity CAMs. And I can access my TrueNAS. Nothing on the home network is exposed to the internet. It all happens through the tunnel.*
>
> *The family accesses my shared photo apps (Jellyfin & Piwigo) using a web browser pointed to my custom domain. I add authorized family member email addresses to the Cloudflare dashboard. When a family member tries to log into one of the apps, they just enter their email address. They are sent a PIN for access. All of that is handled by Cloudflare.*
>
> *It’s a little bit of propeller beanie kind of stuff but one could just start with the tunnel to access the home network without sharing apps and dealing with authentication. Oh, I forgot to mention, all of the stuff I use at Cloudflare is FREE! I hope this might help anyone searching for this type of solution. Best… Scott*

This was news to me, so I went over to take a look. Cloudflare’s Tunnel page says:

> *Protect your web servers from direct attack. From the moment an application is deployed, developers and IT spend time locking it down — configuring ACLs, rotating IP addresses, and using clunky solutions like GRE tunnels. There’s a simpler and more secure way to protect your applications and web servers from direct attacks: Cloudflare Tunnel. Ensure your server is*

> *safe, no matter where it’s running: public cloud, private cloud, Kubernetes cluster, or even a Mac mini under your TV.*

From Scott’s description, it sounds like an extremely powerful and capable solution. For simple safe remote connections to an internal network it may be more than many of our listeners need. But I wanted to put it on everyone’s radar in case it might be a match.

Jeff Price also happened to write:

> *Thanks for the emails. They are very helpful. I have a medium sized network at home with Synology NAS, dozens of IOT devices etc. I have been using Tailscale for all remote connections. This means no open ports or port forwarding. I also set up a system inside my home as an exit node which means even when I am traveling I can encrypt all my traffic back to my home and then exit from there. Tailscale has worked great and it is much faster than OpenVPN.*

I wanted to share Jeff’s note since my sense is that whereas Scott’s solution using Cloudflare’s Tunnel service is extremely powerful and may be the best fit for higher-end needs, a simple overlay network such as TailScale, ZeroTier, Nebula or NetMaker may be all that most users need.

Adam B has an intriguing problem...

> *Hi Steve, I'm a long-time listener to the show - I'm not sure how long, but I definitely remember when you used to alternate episodes between topics and news. I'm a proud Spinrite owner and, thanks to you and Leo getting me interested in Hackerone, a few hundred dollars better off having found a couple of Local Privilege Escalation vulnerabilities during some poking around on my weekends.*
>
> *I have a question that I haven't been able to find an answer to online and I thought might interest you and my fellow listeners. I'm a hobbyist malware analyst and as part of that I often run the samples in a network that's isolated from the Internet - just to see what happens. Sometimes, the samples will try and communicate with a 'Command and Control' (C2) server. Often, the hard-coded C2 server is a Fully-Qualified Domain Name (FQDN), but sometimes it's a public IP address.*
>
> *It can often be useful to pretend to be the C2 server, just to see what the sample sends. When the C2 server is a FQDN it's easy enough to use my own DNS server in the isolated network to answer the DNS request with an A record IP address of my choosing.*
>
> *However, when the C2 server is a public IP address, this becomes more troublesome. I think I have two choices:*
>
> *1. Patch the sample to change the IP address to one on the LAN.*
> *2. Somehow get my LAN to answer the ARP request with a MAC address of my choosing.*
>
> *The problem with choice #1 is that this isn't practical at scale. As in, sometimes I like to run 10, 20 or 50 versions of the same malware family - I don't want to have to manually patch 50*

different samples. It also seems like the less satisfactory choice.

The problem with choice #2 is that I simply can't figure out how to do it! How can I configure my network so that if a sample makes a request for a public IP address (i.e., one that isn't in the /24 of my LAN) the request is handled by my C2 server? The best answer I could find online was concerned with "ARP poisoning", but this seemed very unreliable and likely to cause an unstable network. It feels like the answer will be something to do with the default gateway, but I can't figure it out.

I hope that makes sense! I would really appreciate your thoughts on the subject. A big thank you to you, Leo and the whole team. Kind regards, Adam

What Adam wants to do can definitely be done in a highly robust fashion. It would be possible to manually add static routes to the routing table of the machine that's hosting the malware. This would cause traffic bound for that target IP to override the normal non-local default route which would send the traffic out to the network's gateway interface and instead to another local network interface. But doing that is tricky and messy.

The more straightforward solution would be to obtain a router that has some extra hardware interfaces. That little NetGate SG-1100 I'm using here has an "AUX" network connection. It's not a simple switch using the same network as the LAN interface. It's a separate network interface that can be given its own LAN. Or one of those Protectli Vault devices that I'm using at my other location would be perfect. The idea is to have an extra physical network interface. You would use the router's software such as pfSense or OPNsense to define another small LAN network for that extra interface. And instead of using one of the normal private networks like 192.168.x.x or 10.x.x.x, you would create a network that includes the target IP of the command and control server. You then attach a machine to that interface and manually assign it the IP of the command and control server.

Now, whenever the malware in the host machine addresses Internet traffic to that remote public IP, your local router's routing table will see that the IP matches within that extra network and it will send the traffic to it rather than out onto the public Internet. So you wind up with a straightforward, robust and easily adjusted and maintained solution.

Dale Myers has a problem no one should ever face

Hi Steve, I never thought, when I started listening at #0001 that there would ever be a thousand (and still counting) Security Now podcasts. I started at the beginning, right after Fred Langa suggested that your podcast might be worthwhile. He was right.

At that time I was a volunteer in the IT department of a parochial school. The things I learned from Security Now led to important improvements in our system over the years. In those days there were not so many listeners and you took time to answer two of my questions submitted in the "Feedback" dialog box at the bottom of the SN page.

Now I have a new question that relates to using a password manager. I have been doing a bit of traveling by air lately and the last time I was in my travel agent's office I decided to use some of the accumulated points. She said she couldn't access my account without my

> *password. There was a place for it on her screen, but I couldn't figure out how to get the password there from my password manager. Thoughts? Dale Myers*

My first thought was "Huh! ... that's a really good question. How would you do that securely?" and then I thought: "I wonder why this isn't a problem that I've heard of before?" and then that question answered itself, since no one should EVER have this problem! No one should ever be asked to give their password to someone else – like a travel agent – so that she could access their account. So it's not a bigger problem because it should never be required of anyone, ever.

The whole thing seems like a fundamentally bad idea. But that doesn't help Dale who apparently does have this problem, even if everyone agrees he should never have this problem in the first place. Given that Dale has been listening since episode 1, we know that his travel account is currently protected by a ridiculously gnarly, long, random and impossible to manually enter or even communicate, password.
So my advice would be not to try. Briefly change your password to something ridiculously simple to type which meets the travel system's password policies—but otherwise minimal in every way. It's only going to be that way for a few minutes so its security doesn't matter. Once the travel points have been transferred, the account's password can either be restored to what it was before, or set to something new. A workable alternative would be to just send the account's initial gnarly password via email or text to the travel agent, let her login and do whatever she needs, then change the account's password to something secure once the points have been moved.

Chris C asked:

> *A while back, you said something about a large company that was fined for not keeping Teams or Slack chats per Federal Law. Do you remember who this was and what the law was?*

I replied to Chris: *"I vaguely recall that passing by, but I have no specific recollection. GRC's onsite search (in the upper right of every page) can be used to search (only) the podcast transcripts which are fully indexed. So you might be able to track down the reference that way."*
I wanted to share this because I use GRC's search from time to time in the same way when I'm looking for something from our own past. You've heard me casually mention that we talked about something or other back during podcast XYZ. Don't imagine for a second that I recalled the podcast. Like Chris here, I did recall that it was something that was mentioned, but not what or when exactly. Since I get these questions often, I wanted to pass on to everyone that both the show notes and Elaine's precise transcripts are fully indexed and that index can be easily searched using GRC's search box.

And Chris replied…

> *Thank you! I didn't know that was there. I found it in SN #959. Google didn't help me, but the search engine on your site ("Powered" by the same company) did.*

Artificial General Intelligence (AGI)

As planned, I want to start off our second one thousand episodes of Security Now! by setting the stage for what appears to be the biggest thing to happen to many aspects of our world. I'm talking, of course, about AI.

I should note that I already have everything I need with today's ChatGPT 4o — and it has changed my life for the better. I've been using it increasingly as a time saver in the form of a programming language super search engine and even a programming language syntax checker. I've used it as a crutch when I need to quickly write some throwaway code in a language, like PHP, where I do not have expertise but I want to get something done quickly.

In the past, I would take an hour or two of putting queries into Google and following links to Programmer's Corner, Stack Overflow or other similar online sites. And I would piece together the language construction I need from other similar bits of code that I would find online. Or if I was unable to find anything that way, I'd dig down deeper to read through the language's manuals to find the usage and syntax for the bits I needed and build up from that. But that's no longer what I do, because I now have access to a super programming language search engine.

Now I ask the experimental coding version of ChatGPT for what I need. I don't ask it to provide a complete program, since that's not what I want. I love coding in any language because I love puzzles, and puzzles are language-agnostic. But I do not equally **know the details** of every language. There's nothing ChatGPT can tell me about assembly language that I haven't known for decades. But if I want to write a quick throwaway utility program in Visual Basic dot Net, a language I've spent very little time with, and I need to quickly implement an associative array, rather than poking around the Internet or scanning through the Visual Basic syntax to find what I'm looking for, I'll now just pose the question to ChatGPT. I'll ask it very specifically and carefully for what I want and in about two seconds I'll get what I may have previously spent 30 to 60 minutes sussing out online. It's transformed my work path. It's useful whenever I need some details where I do not have expertise. I've seen plenty of criticism levied by other programmers of the code produced by today's AI. To me it seems misplaced and maybe just a bit nervous. I don't ask ChatGPT for a finished product because I know exactly what I want and I'm not even sure I could specify that in words. So I just ask for specific bits and pieces and the results have been fantastic.

But what I want to explore today is what lies beyond what we have today, what the challenges are and what predictions are being made about how and when we may get there. The "there" where we want to get is generically known as "Artificial General Intelligence", abbreviated AGI. Let's start by looking at how Wikipedia defines this goal. Wikipedia says:

> *Artificial general intelligence (AGI) is a type of artificial intelligence (AI) that matches or surpasses human cognitive capabilities across a wide range of cognitive tasks. This contrasts with narrow AI, which is limited to specific tasks. Artificial superintelligence (ASI), on the other hand, refers to AGI that greatly exceeds human cognitive capabilities. AGI is considered one of the definitions of strong AI.*

> *Creating AGI is a primary goal of AI research and of companies such as OpenAI and Meta. A 2020 survey identified 72 active AGI research and development projects across 37 countries.*
>
> *The timeline for achieving AGI remains a subject of ongoing debate among researchers and experts. As of 2023, some argue that it may be possible in years or decades; others maintain it might take a century or longer; and a minority believe it may never be achieved. Notable AI researcher Geoffrey Hinton has expressed concerns about the rapid progress towards AGI, suggesting it could be achieved sooner than many expect.*
>
> *There is debate on the exact definition of AGI, and regarding whether modern large language models (LLMs) such as GPT-4 are early forms of AGI. Contention exists over whether AGI represents an existential risk. Many experts on AI have stated that mitigating the risk of human extinction posed by AGI should be a global priority. Others find the development of AGI to be too remote to present such a risk.*
>
> *AGI is also known as strong AI, full AI, human-level AI, or general intelligent action. However, some academic sources reserve the term "strong AI" for computer programs that experience sentience or consciousness. In contrast, weak AI (or narrow AI) is able to solve one specific problem but lacks general cognitive abilities. Some academic sources use "weak AI" to refer more broadly to any programs that neither experience consciousness nor have a mind in the same sense as humans.*
>
> *Related concepts include artificial superintelligence and transformative AI. An artificial superintelligence (ASI) is a hypothetical type of AGI that is much more generally intelligent than humans, while the notion of transformative AI relates to AI having a large impact on society, for example, similar to the agricultural or industrial revolution.*
>
> *A framework for classifying AGI in levels was proposed in 2023 by Google DeepMind researchers. They define five levels of AGI: emerging, competent, expert, virtuoso, and superhuman. For example, a competent AGI is defined as an AI that outperforms 50% of skilled adults in a wide range of non-physical tasks, and a superhuman AGI (i.e. an artificial superintelligence) is similarly defined but with a threshold of 100%. They consider large language models like ChatGPT or LLaMA 2 to be instances of the first level, emerging AGI.*

Okay. So we're getting some useful language and terminology for talking about these things. The article that caught my eye last week as we were celebrating the 1000th episode of this podcast was posted on Perplexity.AI, titled *"Altman Predicts AGI by 2025."* The Perplexity piece turned out to now have meat, but it did offer the kernel of some interesting thoughts, and some additional terminology and talking points, so I still want to share it. Perplexity wrote:

> *OpenAI CEO Sam Altman has stirred the tech community with his prediction that Artificial General Intelligence (AGI) could be realized by 2025, a timeline that contrasts sharply with many experts who foresee AGI's arrival much later. Despite skepticism, Altman asserts that OpenAI is on track to achieve this ambitious goal, emphasizing ongoing advancements and substantial funding, while also suggesting that the initial societal impact of AGI might be minimal.*
>
> *In a Y Combinator interview, Altman expressed excitement about the potential developments in AGI for the coming year. However, he also made a surprising claim that the advent of AGI would have "surprisingly little" impact on society, at least initially. This statement has sparked*

> *debate among AI experts and enthusiasts, given the potentially transformative nature of AGI.*
>
> *And Altman's optimistic timeline stands in stark contrast to many other experts in the field, who typically project AGI development to occur much later, around 2050. Despite the skepticism, Altman maintains that OpenAI is actively pursuing this ambitious goal, even suggesting that it might be possible to achieve AGI with current hardware. This confidence, coupled with OpenAI's recent $6.6 billion funding round and its market valuation exceeding $157 billion, underscores the company's commitment to pushing the boundaries of AI technology.*
>
> *Achieving Artificial General Intelligence (AGI) faces several significant technical challenges that extend beyond current AI capabilities:*
>
> - ***Common-sense reasoning:*** *AGI systems must develop intuitive understanding of the world, including implicit knowledge and unspoken rules, to navigate complex social situations and make everyday judgments.*
>
> - ***Context awareness:*** *AGI needs to dynamically adjust behavior and interpretations based on situational factors, environment, and prior experiences.*
>
> - ***Handling uncertainty:*** *AGI must interpret incomplete or ambiguous data, draw inferences from limited information, and make sound decisions in the face of the unknown.*
>
> - ***Continual learning:*** *Developing AGI systems that can update their knowledge and capabilities over time without losing previously acquired skills remains a significant challenge.*

One thing that occurs to me as I read those four points – reasoning, contextual awareness, uncertainty and learning – is that none of the AIs I've ever interacted with has ever asked for any clarification about what I was asking; that's not something that appears to be wired into the current generation of AI. I'm sure it could be simulated if it would further raise the stock price of the company doing it. But it wouldn't really matter, right? Because it would be a faked question… like that very old Eliza pseudo-therapist program from the 70's. When you wrote "I'm feeling sort of cranky today." It would reply "Why do you think you're feeling sort of cranky today?"

The point I hope to make is that there's a hollowness to today's AI. It's truly an amazing search engine technology. But it doesn't seem to be much more than that. There's no presence or understanding behind its answers. The Perplexity article continues:

> *Overcoming these hurdles requires advancements in areas such as neural network architectures, reinforcement learning, and transfer learning. Additionally, AGI development demands substantial computational resources and interdisciplinary collaboration among experts in computer science, neuroscience, and cognitive psychology.*
>
> *While some AI leaders like Sam Altman predict AGI by 2025, many experts remain skeptical of such an accelerated timeline. A 2022 survey of 352 AI experts found that the median estimate for* ***AGI*** *development was around 2060 – also known as Security Now! Episode #2860. 90% of the 352 experts surveyed expect to see AGI within 100 years.*

> *This more conservative outlook stems from several key challenges:*
>
> - *The "missing ingredient" problem: Some researchers argue that current AI systems, while impressive, lack fundamental components necessary for general intelligence. Statistical learning alone may not be sufficient to achieve AGI.*
>
> - *Training limitations: Creating virtual environments complex enough to train an AGI system to navigate the real world, including human deception, presents significant hurdles.*
>
> - *Scaling challenges: Despite advancements in large language models, some reports suggest diminishing returns in improvement rates between generations.*
>
> *These factors contribute to a more cautious view among many AI researchers, who believe AGI development will likely take decades rather than years to achieve.*
>
> *OpenAI has recently achieved significant milestones in both technological advancement and financial growth. The company successfully closed a massive $6.6 billion funding round, valuing the AI startup at $157 billion.*
>
> *This round attracted investments from major players like Microsoft, Nvidia, and SoftBank, highlighting the tech industry's confidence in OpenAI's potential. The company's flagship product, ChatGPT, has seen exponential growth, now boasting over 250 million weekly active users. OpenAI has also made substantial inroads into the corporate sector, with 92% of Fortune 500 companies reportedly using its technologies. Despite these successes, OpenAI faces challenges, including high operational costs and the need for extensive computing power. The company is projected to incur losses of about $5 billion this year, primarily due to the expenses associated with training and operating its large language models.*

https://www.perplexity.ai/page/altman-predicts-agi-by-2025-tUwvEDkiQ9.auqNAMT0X5A

So this gives us a bit of calibration about where we are and the goals of AGI — artificial general intelligence. The follow-on piece I wanted to share appeared in Information Week. The piece's title was: *"Artificial General Intelligence in 2025: Good Luck With That"* with the teaser *"AI experts have said it would likely be 2050 before AGI hits the market. OpenAI CEO Sam Altman says 2025, but it's a very difficult problem to solve."*

> *A few years ago, AI experts were predicting that artificial general intelligence (AGI) would become a reality by 2050. OpenAI has been pushing the art of the possible, along with Big Tech, but despite Sam Altman's estimate of 2025, realizing AGI is unlikely soon.*
>
> *HP Newquist, author of The BrainMakers and executive director of The Relayer Group, a consulting firm that tracks the development of practical AI said: "We can't presume that we're close to AGI because we really don't understand current AI, which is a far cry from the dreamed-of AGI. We don't know how current AIs arrive at their conclusions, nor can current AIs even explain to us the processes by which that happens. That's a huge gap that needs to be closed before we can start creating an AI that can do what every human can do. And a hallmark of human thinking, which AGI will attempt to replicate, is being able to explain the rationale for coming up with a solution to a problem or an answer to a question. We're still trying to keep existing Large Language Models from hallucinating."*

I'll interrupt here just to say that I think this is **the** crucial point. Earlier, I described ChatGPT as being a really amazingly powerful Internet search engine. Partly that’s because that’s what I’ve been using it to replace. For my own needs it has been a miraculous replacement for a bunch of searching I would otherwise need to do myself. My point is, this entire current Large Language Model approach may never be more than that. This could be a dead end. If so, it’s a super useful dead end. But it might not be the road to AGI at all. It might never amount to being more than a super spiffy search engine. The InfoWeek article continues:

> *OpenAI is currently alpha testing advanced voice mode, which is designed to sound human (such as pausing occasionally when one speaks to draw a breath). It can also detect emotion and non-verbal clues. This advancement will help AI seem more human-like, which is important, but there’s more work to do.*
>
> *Edward Tian, CEO of ZeroGPT, which detects generative AI’s use in text, also believes the realization of AGI will take time. In an email interview with the article’s author, Edward said:*
>
> *“The idea behind artificial general intelligence is creating the most human-like AI possible -- a type of AI that can teach itself and essentially operate in an autonomous manner. So, one of the most obvious challenges is creating AI in a way that allows the developers to be able to take their hands off eventually, as the goal is for it to operate on its own. Technology, no matter how advanced, cannot be human, so the challenge is trying to develop it to be as human as possible. That also leads to ethical dilemmas regarding oversight. There are certainly a lot of people out there who are concerned about AI having too much autonomy and control, and those concerns are valid. How do developers make AGI while also being able to limit its abilities when necessary? Because of all these questions and our limited capabilities and regulations at the present, I do not believe that 2025 is realistic.”*
>
> *Current AI -- which is artificial narrow intelligence (ANI), performs a specific task well, but it cannot generalize that knowledge to suit a different use case.*
>
> *Max LI, the CEO of the decentralized AI data provider Oort and an adjunct associate professor in the department of electrical engineering at Columbia University said: “Given how long it took to build current AI models, which suffer from inconsistent outputs, flawed data sources, and unexplainable biases, it would likely make sense to perfect what already exists rather than start working on even more complex models. In academia, for many components of AGI, we do not even know why it works, nor why it does not work.”*
>
> *To achieve AGI, a system needs to do more than just produce outputs and engage in conversation, which means that LLMs alone won’t be enough. Alex Jaimes, chief AI officer at the AI company Dataminr, said in an email interview: “It should also be able to continuously learn, forget, make judgments that consider others, including the environment in which the judgments are made, and a lot more. From that perspective, we’re still very far. It’s hard to imagine AGI that doesn’t include social intelligence, and current AI systems don’t have any social capabilities, such as understanding how their behavior impacts others, cultural and social norms, etc.”*
>
> *Sergey Kastukevich (kas-tuke’-vich), the deputy CTO at the gambling software company SOFTSWISS said: “To get to AGI, we need advanced learning algorithms that can generalize and learn autonomously, integrated systems that combine various AI disciplines, massive computational power, diverse data and a lot of interdisciplinary collaboration. For example, current AI models like those used in autonomous vehicles require enormous datasets and computational power just to handle driving in specific conditions, let alone achieve general*

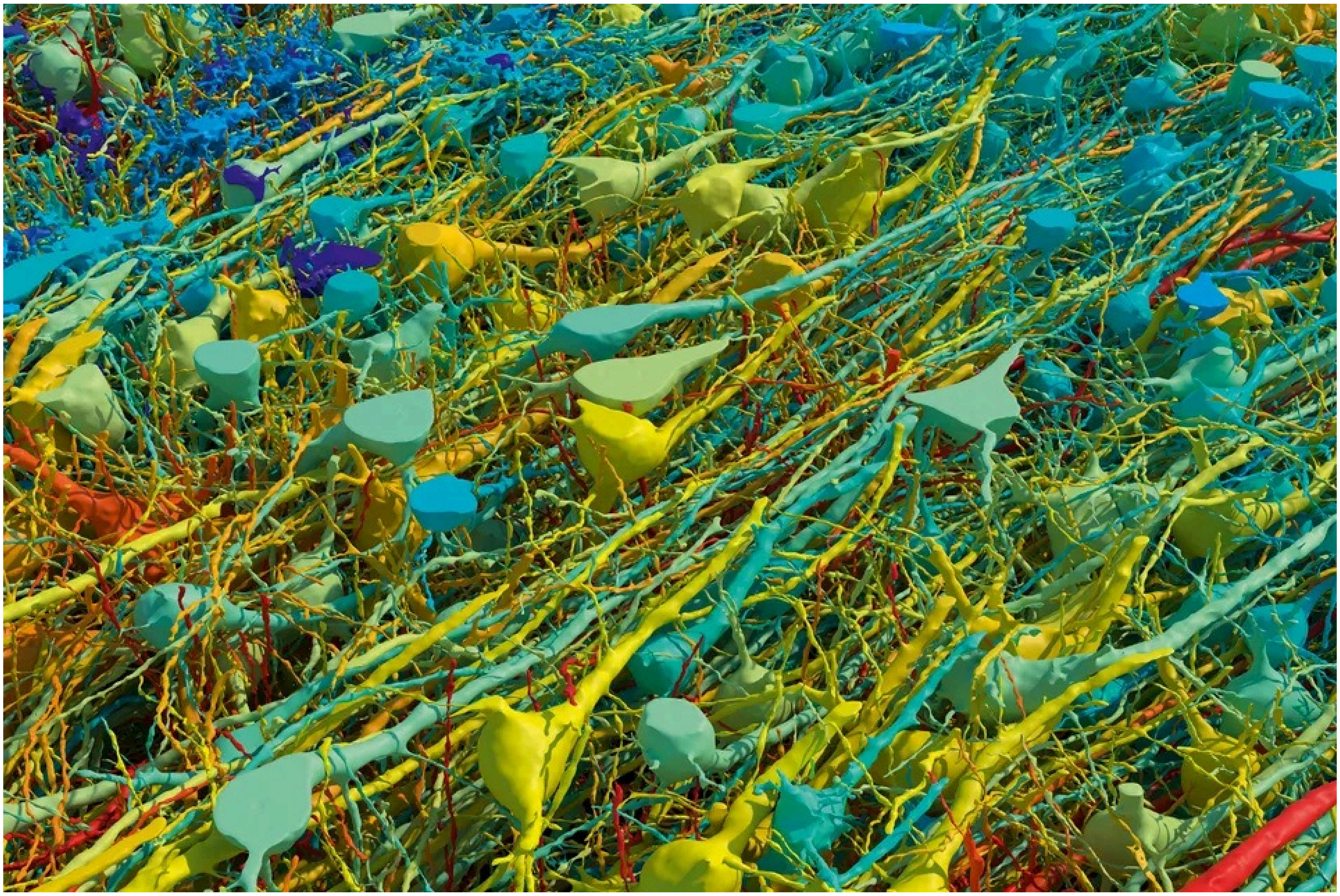
intelligence."

LLMs are based on complex transformer models. While they are incredibly powerful and even have some emergent intelligence, the transformer is pre-trained and does not learn in real-time. For AGI, there will need to be some breakthroughs with AI models. They will need to be able to generalize about situations without having to be trained on a particular scenario. A system will also need to do this in real-time, just like a human can when they intuitively understand something.

In addition, AGI capabilities may need a new hardware architecture, such as quantum computing, since GPUs will probably not be sufficient. Note that Sam Altman has specifically disputed this and said that current hardware will be sufficient. In addition, the hardware architecture will need to be much more energy efficient and not require massive data centers.

LLMs are beginning to do causal inference and will eventually be able to reason. They'll also have better problem-solving and cognitive capabilities based on the ability to ingest data from multiple sources.

It feels to me as though all of these people – except Sam Altman – know what they're talking about and they appear to be coming from a place of reason. Perhaps Sam has already seen things in the lab at OpenAI that no one else in the outside world has seen, because that's what it would take for Sam to not be guilty of overhyping and over-promoting his company's near term future.



I’m especially intrigued by the comments from the top academic AI researchers in the world who admit that, to this day, no one actually understands how Large Language Models produce what they do. Given that, I’m skeptical that just “more of the same” will result in the sort of qualitative advancement that AGI would require... which is clearly not just “more of the same.”

When I said in the past that I see no reason why a true artificial intellect could not **eventually** be created, I did not mean next year. I meant ... someday. I meant that I believe that a biological brain may only be one way to create intelligence. One thing I’ve acquired during my research into the biology of the human brain is a deep appreciation for the astonishing complexity of the biological computing engine that **is** us. The number of individual computing neurons is 10^11. That’s 100 billion individual neurons. A billion neurons 100 times over. Consider that. And not only are these individual neurons very richly interconnected, each individual neuron is, all by itself, astonishingly complex in its behavior and operation. They are far from being simple integrative binary triggers. And we have 100 billion of these little buggers in our heads.

Perhaps Sam is going to surprise the rest of the world next year. We’ll see. Color me skeptical but not disappointed. I’m quite happy to have discovered the wonderful, language accessible, Internet digest he’s created. It’s more than a simple parlor trick. It’s a big deal. And it’s kind of magic. But I suspect that all it is, is what it is. And that’s enough for now. I’d wager that we have a long ways to go.