Security Now! #816 - 04-27-21
The Mystery of AS8003

This week on Security Now!

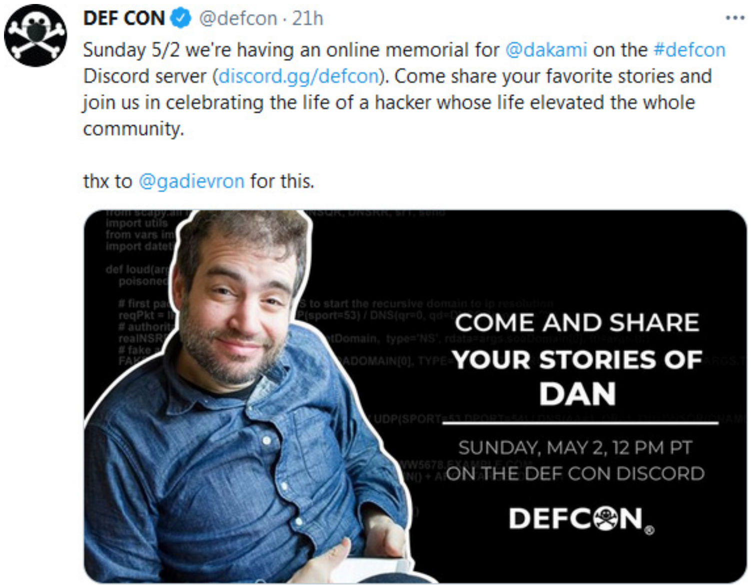
This week we begin by remembering Dan Kaminsky, who the world lost last Friday at the age of 42. We finally catch up with this month's Patch Tuesday, and look at a welcome maturation in Google's Project Zero vulnerability disclosure policy. We shine a light upon a new startup venture which, if successful, promises to dramatically improve the future of IoT security. We then look at some controversial security research, for which the researchers have apologized, and wonder whether any apology was due. We shine another light onto a new battle Cloudflare has chosen to wage against an abusive patent troll, to help Cloudflare with additional attention, and to let our listeners know that they can participate in a money-making hunt for prior art. And after a brief SpinRite progress report, we engage with the Internet mystery of the Autonomous System 8003.

13 years ago, Dan Kaminsky with his niece, Sarah, after BlackHat 2008

"Sarah on DNS" — a short, 1:45 video at DailyMotion:
https://www.dailymotion.com/video/x2lp3m4

Remembering Dan

(1979 - April 23, 2021 @ age 42)



DEF CON @defcon · 21h

Sunday 5/2 we're having an online memorial for @dakami on the #defcon Discord server (discord.gg/defcon). Come share your favorite stories and join us in celebrating the life of a hacker whose life elevated the whole community.

thx to @gadievron for this.

https://dankaminsky.com/

From Dan's blog, July 26th, 2017:

> *Cryptographically Secure Pseudorandom Number Generators are interesting. Given a relatively small amount of data (just 128 bits is fine) they generate an effectively unlimited stream of bits completely indistinguishable from the ephemeral quantum noise of the Universe. The output is as deterministic as the digits of Pi, but no degree of scientific analysis, no amount of sample data will ever allow a model to form for what bits will come next.*
>
> *In a way, CSPRNGs represent the most practical demonstration of Godel's First Incompleteness Theorem, which states that for a sufficiently complex system, there can be things that are true about it that can never be proven within the rules of that system. Science is literally the art of compressing vast amounts of experimentally derived output on the nature of things, to a beautiful series of rules that explains it. But as much as we can model things from their output with math, math can create things we can never model. There can be a thing that is true — there are hidden variables in every CSPRNG — but we would never know.*
>
> *And so an interesting question emerges. If a CSPRNG is indistinguishable from the quantum noise of the Universe, how would we know if the quantum noise of the universe was not itself a CSPRNG? There's an infinite number of ways to construct a Random Number Generator, what if Nature tried its luck and made one more? Would we know?*
>
> *Would it be any good?*

Dan was a prolific Tweeter, posting nearly 130 thousand tweets since he joined Twitter in 2007. Assuming a constant rate of tweets, that's 14 years, so 9,285.71 tweets per year, or an average of 25.42 tweets or commented retweets per day. So Dan's 94.3 thousand followers would have received a continual stream of what Dan was thinking and doing. He pinned a tweet of his from early 2018 to his feed:

Dan Kaminsky / @dakami / Jan 16, 2018
I'm increasingly thinking that every functioning system has two forms: The abstraction that outsiders are led to believe, and the reality that insiders actually and carefully operate. You don't incrementally learn a system. You eventually unlearn its necessary lies.

Security News

Week before last was Patch Tuesday

Patch Tuesday was the week before last, and I missed summarizing it last week. It would be nice if it was all old news by now. But believe it or not, the NSA contributed their knowledge of 4 additional remote code execution flaws in — you guessed it — Microsoft's Exchange Server. Now, I'm not a conspiracy guy. But ya gotta wonder whether these 4 very valuable RCE flaws in Exchange Server might have been the sort of thing that the NSA had been sitting on quietly as part of its own private stash of remote access tools. Remember that Exchange Server is used world wide and it has now become a well-proven means of gaining surreptitious entry into someone else's system. Exactly the sort of thing that the NSA or CIA might find quite handy. But now, with all of the heat and attention on Exchange Server, they might have correctly decided that it would be far better to help Microsoft more fully clean up this mess with Exchange Server, even at the cost of foreclosing on their future use of some valuable entry points.

Of course, that's all just wild speculation on my part. It's equally likely that they put some of their own hackers onto Exchange Server for the sake of the United States and discovered four previous unknown flaws.

In either case, as part of this month's 114 other security flaw repairs, Microsoft fixed these four RCE flaws (CVE-2021-28480 through CVE-2021-28483) which, naturally, affected all three recent versions of Exchange Server (2013, 2016, and 2019). Two of the four are juicy unauthenticated access flaws requiring no user interaction with CVSS scores of 9.8.

Overall, of the total 114 flaws, 19 are rated as Critical, 88 are rated Important, and one is rated Moderate in severity. The one of most concern was CVE-2021-28310, a privilege escalation vulnerability in Win32k that's under active exploitation, allowing attackers to elevate privileges by running malicious code on a target system. Kaspersky Labs found and reported this one to Microsoft. Kaspersky's Boris Larin said *"It is an escalation of privilege (EoP) exploit that is likely used together with other browser exploits to escape sandboxes or get system privileges for further access."*

Google's Project Zero responds to today's patch latency reality

Until now, Google's project zero set a fixed 90-day timer on the disclosure of vulnerabilities discovered and privately disclosed to vendors. Vendors had 90 days from the date of notification before Project Zero would lower the boom on them. And in the case of discovered-in-the-wild 0-days, that 90 days was reduced to just one week to reflect the extreme danger posed to users. The intention was to really light a fire in the vendor to get them moving on a patch with urgency.

But now in a welcome and, I think, sane announcement last week, Google announced the addition of a new 30-day patch-latency allowance: If the vendor fixes the problem and publishes a patch before the initially allotted deadline — 90 days or 7 days — then Google will add a 30-day grace period onto the end of the original deadline to reflect the reality that releasing a patch and actually having systems patched against the now-known vulnerability are two very different things.

Baking security into IoT

A newly financed startup named Thistle aims to take aim at IoT security by providing a secure turn-key security foundation.

For more than 20 years, the new firm's founder "Window Snyder" has been building security into the products of some of the largest companies in the world. The startup is creating tools that will help manufacturers build security into connected devices from the ground up. As we know, the manufacturers of Printers, ATMs, consumer electronics, automobiles, light switches and connected plugs — any and all IoT gizmos — typically don't have the security expertise that companies like Apple, Microsoft, and Google have developed over the past 20 years. And the result is all too well known to us: Billions of devices ship with vulnerabilities that are preyed upon by profit-driven criminals and nation-state hackers.

Window is a security veteran who previously served as chief security officer at Square, Mozilla, and Fastly and was chief software security officer at Intel. While a teenager, she was part of a Boston hacker collective before going on to be a consultant at @stake, a security company that employed many of the members of L0pht, another Boston hacker collective. She also spent time at Microsoft working on Windows XP SP2, the update that added a range of much-needed security improvements to Windows. Later, she worked on security at Apple. So she knows what she's talking about.

She was quoted: *"What it takes to build security into products… requires a lot of really specialized skills. You get folks, especially at the devices level, building the same security mechanisms over and over again, reinventing the wheel, and doing it to different levels of resilience."*

So Windows' firm, Thistle, will develop frameworks that allow device manufacturers to quickly build reliable and resilient security into their products more quickly than they could do on their own. And, believe it or not [be still my heart!] the company's initial work will focus on building a platform that delivers security updates to connected devices.

Patching devices typically requires reflashing firmware, a process that can be fraught with risk and as Window notes: *"It's one of the reasons that nobody delivers updates for devices, because the cost of failing an update is so high. If you've got 100 million devices out there and you've got a 1-percent failure rate—which is very, very low for updates—that's still a million devices that are bricked."*

This is wonderful news. I sure hope she succeeds. Farming this out to a security specialist is EXACTLY the right thing to do.

UNethical security research (???)

The University of Minnesota has just gotten itself in trouble with, and banned from all future contributions to, the Linux Kernel Project.

I'm inclined to label this controversial, though many don't believe there's any doubt about this being unequivocally wrong. The industry's reporting on this stated that three security researchers — an associate professor and two of his grad students — deliberately introduced live use-after-free vulnerabilities into the production Linux kernel in the cause of security research aiming to highlight how potentially malicious code could be deliberately introduced into an open source project and sneak past the patch approval process. Their goal was to demonstrate the problem and suggest ways to improve the security of the code approval and patching process. So they did this in the real world by attempting to sneak their own malicious code patches into the actual production Linux kernel.

This research was conducted earlier this year, and when what they had done came to light recently — as a result of their own admission — they apologized, saying: "While our goal was to improve the security of Linux, we now understand that it was hurtful to the community to make it a subject of our research, and to waste its effort reviewing these patches without its knowledge or permission. We did that because we knew we could not ask the maintainers of Linux for permission, or they would be on the lookout for the [so-called] hypocrite patches." (I'll explain that in a minute.)

The researchers claimed "We did not introduce or intend to introduce any bug or vulnerability in the OS," but evidence emerged to the contrary, suggesting that the research was conducted without adequate oversight and risked the kernel's security. This resulted in a unilateral ban of all future code submissions from anyone using a "umn.edu" email address, and also invalidated all past code submitted by the university's previous researchers.

The Linux Kernel Project was quite upset. They said: "Our community does not appreciate being experimented on, and being 'tested' by submitting patches that either deliberately do nothing or deliberately introduce bugs." Responding to the Linux Project's response, another observer tweeted "This is worse than just being experimented upon; this is like saying you're a 'safety researcher' by going to a grocery store and cutting the brake lines on all the cars to see how many people crash when they leave. Enormously unethical."

Following the incident, the university's Department of Computer Science and Engineering said it was investigating the incident, adding that it was looking into the "research method and the process by which this research method was approved, determine appropriate remedial action, and safeguard against future issues."

https://github.com/QiushiWu/QiushiWu.github.io/blob/main/papers/OpenSourceInsecurity.pdf

So what about this research? Their paper, which has been accepted for publication and presentation by the IEEE Symposium on Security and Privacy 2021, is titled: "On the Feasibility of Stealthily Introducing Vulnerabilities in Open-Source Software via Hypocrite Commits"

It explains itself in is abstract:

> Open source software (OSS) has thrived since the forming of Open Source Initiative in 1998. A prominent example is the Linux kernel, which has been used by numerous major software vendors and empowering billions of devices. The higher availability and lower costs of OSS boost its adoption, while its openness and flexibility enable quicker innovation. More importantly, the OSS development approach is believed to produce more reliable and higher-quality software since it typically has thousands of independent programmers testing and fixing bugs of the software collaboratively.
>
> In this paper, we instead investigate the insecurity of OSS from a critical perspective—the feasibility of stealthily introducing vulnerabilities in OSS via hypocrite commits (i.e., seemingly beneficial commits that in fact introduce other critical issues). The introduced vulnerabilities are critical because they may be stealthily exploited to impact massive devices. We first identify three fundamental reasons that allow hypocrite commits. (1) OSS is open by nature, so anyone from anywhere, including malicious ones, can submit patches. (2) Due to the overwhelming patches and performance issues, it is impractical for maintainers to accept preventive patches for “immature vulnerabilities”. (3) OSS like the Linux kernel is extremely complex, so the patch review process often misses introduced vulnerabilities that involve complicated semantics and contexts. We then systematically study hypocrite commits, including identifying immature vulnerabilities and potential vulnerability-introducing minor patches. We also identify multiple factors that can increase the stealthiness of hypocrite commits and render the patch-review process less effective. As proof of concept, we take the Linux kernel as target OSS and safely demonstrate that it is practical for a malicious committer to introduce use-after-free bugs. Furthermore, we systematically measure and characterize the capabilities and opportunities of a malicious committer. At last, to improve the security of OSS, we propose mitigations against hypocrite commits, such as updating the code of conduct for OSS and developing tools for patch testing and verification.

I would argue that this is a very valid, very important, and very much needed avenue of research. It should be clear that surreptitious commits would inherently be a huge problem for any large open source project that’s open to many contributors. The only solution I can see is extremely careful multi-party scrutiny of any changes made to the Kernel. But that's rather thankless and boring work.

This amounts to debugging code that's assumed to be correct and is not known to have anything wrong with it. So it's very similar to the trouble we've often spoken of, of debugging one's own code which is, similarly, believed to be correct. As I've observed, it often takes single-stepping through such code — which you “know” is correct even though it's misbehaving, to have the code debugger rub your face in the mistake — before it is seen and inevitably invokes the “Ah Hah!!” reaction.

So, auditing every line of code, that may have been very very cleverly designed to misbehave only under a very subtle edge case condition, is probably impossible. It represents an ongoing Achilles heel for large open projects.

In a follow-up FAQ they attempted to clarify what they had done. The FAQ is titled:
“Clarifications on the “hypocrite commit” work (FAQ)”

https://www-users.cs.umn.edu/~kjlu/papers/clarifications-hc.pdf

We recently finished a work that studies the patching process of OSS. Its goal is to improve the security of the patching process. The corresponding paper has been accepted by IEEE S&P 2021. I shared the abstract of the paper on Twitter, which then resulted in heated discussion and pushback. I apologize for the misleading abstract which did not show the details and caused many confusions and misunderstandings. Therefore, we would like to make a few clarifications.

We would like to first mention that we are a young research group with improving the kernel security as the first priority. In the past several years, we devote most of our time to improving the Linux kernel, and we have found and fixed more than one thousand kernel bugs; the extensive bug finding and fixing experience also allowed us to observe issues with the patching process and motivated us to improve it. Thus, we consider ourselves security researchers as well as OSS contributors. We respect OSS volunteers and honor their efforts. We have never intended to hurt any OSS or OSS users. We did not introduce or intend to introduce any bug or vulnerability in OSS. The following are the clarifications to the common concerns we received.

* The purpose and research value of the work

The project aims to improve the security of the patching process in OSS. As part of the project, we study potential issues with the patching process of OSS, including causes of the issues and suggestions for addressing them. This study indeed reveals some issues, but its goal is to call for efforts to improve the patching process---to motivate more work that develops techniques to test and verify patches, and finally to make OSS safer.

In this work, we collect 138 previous bug-introducing patches (not introduced by us). Based on these patches, we summarize their patterns, study specific reasons why bug-introducing patches are hard to catch (with both a qualitative and a quantitative analysis), and more importantly, provide suggestions to addressing the problem. In this work, we introduce the concept of “immature vulnerability” where a vulnerability condition of it is missing, but it can be turned into a real one when the condition is implicitly introduced by a patch for another bug. We also develop tools that help us find code places that may suffer from bug-introducing patches, and suggest what may make these bug-introducing patches hard to catch.

* Did the authors introduce or intend to introduce a bug or vulnerability?

No. As a part of the work, we had an experiment to demonstrate the practicality of bug-introducing patches. This is actually the major source of the raised concerns. In fact, this experiment was done safely. We did not introduce or intend to introduce any bug or vulnerability in the Linux kernel. All the bug-introducing patches stayed only in the email exchanges, without being adopted or merged into any Linux branch, which was explicitly confirmed by maintainers. Therefore, the bug-introducing patches in the email did not even become a Git commit in any Linux branch. None of the Linux users would be affected. The following shows the specific procedure of the experiment.

... And it continues. I have the links to both PDFs in the show notes. I really don't find any fault with what these guys did. I would argue that it's vital research. What they did was to exercise the Linux Project's patch management infrastructure — without the Project's knowledge or permission. So the Project's managers are upset over being used in this way. I may not have all of the facts. And I haven't taken the time to study this deeply, nor to look at the maintainer's side of the argument. But this seems like a critically important piece of work, and I get it that it was necessary to use the patch management process without its knowledge or permission.

Their original research paper concluded:

> This paper presented hypocrite commits, which can be abused to stealthily introduce vulnerabilities in OSS. Three fundamental reasons enable hypocrite commits: the openness of OSS, which allows anyone including malicious committers to submit patches; the limited resources of OSS maintaining; and the complexity of OSS programs, which results in the manual review and existing tools failing to effectively identify introduced vulnerabilities. We then systematically characterized immature vulnerabilities and studied how a malicious committer can turn immature vulnerabilities into real ones. We also identified multiple factors that increase the stealthiness of the introduced vulnerabilities, including concurrency, error paths, aliases, indirect calls, etc. Furthermore, we provided a proof-of-concept to safely demonstrate the practicality of hypocrite commits, and measured and quantified the risks. We finally provided our suggestions on mitigating the risks of hypocrite commits and hope that our findings could motivate future research on improving the patching process of OSS.

Miscellany

CloudFlare refuses to knuckle under to Patent Trolls.

https://blog.cloudflare.com/project-jengo-redux-cloudflares-prior-art-search-bounty-returns/

[Leo: Judge Albright from the Western District of Texas strikes again!]

On March 15, Cloudflare was sued by a patent troll called Sable Networks — a company that doesn't appear to have operated a real business in nearly ten years — relying on patents that don't come close to the nature of our business or the services we provide. This is the second time we've faced a patent troll lawsuit.

As readers of the blog (or followers of tech press such as ZDNet and TechCrunch) will remember, back in 2017 Cloudflare responded aggressively to our first encounter with a patent troll, Blackbird Technologies, making clear we wouldn't simply go along and agree to a nuisance settlement as part of what we considered an unfair, unjust, and inefficient system that throttled innovation and threatened emerging companies. If you don't want to read all of our previous blog posts on the issue, you can watch the scathing criticisms of patent trolling provided by John Oliver or the writers of Silicon Valley.

We committed to fighting back against patent trolls in a way that would turn the normal *incentive structure on its head. In addition to defending the case aggressively in the courts, we also founded Project Jengo — a crowdsourced effort to find evidence of prior art to invalidate all of Blackbird's patents, not only the one asserted against Cloudflare. It was a great success — we won the lawsuit, invalidated one of the patent troll's other patents, and published prior art on 31 of Blackbird's patents that anyone could use to challenge those patents or to make it easier to defend against overbroad assertion of those patents. And most importantly, Blackbird Technologies went from being one of the most prolific patent trolls in the United States to shrinking its staff and filing many fewer cases.*

We're going to do it again. And we need your help.

Turning the Tables — A $100,000 Bounty for Prior Art

Sable Networks and its lawsuit fit neatly within the same troubling trends we were trying to address the first time we launched Project Jengo. Sable is taking ancient, 20-year-old patents and trying to stretch those patents lightyears beyond what they were meant to cover. It has already sued over a dozen technology companies targeting a wide range of different products and services, and by extending its claims to a company like Cloudflare suggests it may next try to stretch its claims to people that merely use routers … namely, anyone that uses the Internet.

We think Sable's choice to bring these lawsuits on such a tenuous basis should come with some risk related to the underlying merits of its patents and its arguments, so we are sponsoring another prior-art contest seeking submissions to identify prior art for all of Sable's active patents. We are seeking the help of the Cloudflare community to identify prior art — i.e., evidence that the patented technology was already in use or known before the patent application was filed — that can be used to invalidate Sable's patents. And we will make it worth your while, by offering $100,000 to be shared by the winners who are successful in finding such prior art.

Again this time, we are committing $100,000 to be split among entrants who provide what we determine to be the most useful prior-art references that can be used in challenging the validity of Sable's patents. You can submit prior-art references as long as Sable's case is pending against us (Sable Networks, Inc. v. Cloudflare, Inc., No. 6:21-cv-00261-ADA (W.D. Tex.)), which means until Sable drops the case fully (and with prejudice — meaning Sable can't re-file later), there's a settlement, or the case has been resolved by the court and all appeal rights are exhausted.

ADA and W.D.Tex. stands for "Alan D. Albright" aka "Judge Albright" who is exceedingly patentee friendly and his jurisdiction in the Western District of Texas.

Every three months for two years or until the case ends, whichever comes first, we will select winners from the submissions to date, and give out a portion of the $100,000 as awards. Once the case ends, we will select final winners from all submissions and award the remaining funds. We will also make all relevant submissions available to the public.

CloudFlare's blog post goes on at some length, and it's all REALLY interesting to anyone who has a passion, as I do, for issues surrounding intellectual property rights. They explain how Sable sues companies — like Cisco and Juniper Networks — then settles out of court just before the deadline to actually make their case. It's PURE patent trolling harassment and, as Leo and I have commented before, it really makes our blood boil. I'll share a bit more:

In addition to helping with our case, we hope that making prior art public on all of Sable's patents will provide a head start and decrease the costs for others who want to fight back against Sable's patent trolling. The significant decrease we saw in Blackbird's staff and filings after publishing prior art on Blackbird's patents suggests this approach can be an effective way to undermine the threat posed by those patents.

To understand why we are asking again for your help fighting back, it's worth taking a closer look at this case and the fundamental problems it represents.

The patents at issue in this case started with Caspian Networks, a company that tried to commercialize what it called a "flow-based router" in the early 2000s. Caspian was originally founded as Packetcom in 1998, and revealed to the public its flow-based router named Apeiro in 2003. A press story from that time explained that Apeiro routers worked like traditional routers already in existence, but with additional memory and logic for handling packets from the same "flow." A 2003 slide deck from Caspian distinguished its "flow-based" router from the already existing conventional routers in the following way:

Despite its attempts to tout the benefits of its router, Caspian went out of business in 2006.

That's when Sable Networks, the company that is suing Cloudflare, enters the picture. Though it doesn't appear to be much of an entrance. As best we can tell, Sable briefly picked up where Caspian left off and tried to commercialize Caspian's flow-based routing technology. Sable was equally unsuccessful in doing so, and the last activity of any sort that we could find was from 2011. After a long period of apparent inactivity, Sable's focus shifted last year to trying to extract money by filing lawsuits through broad application of patents filed on Caspian's flow-based router technology twenty years ago. In other words, Sable became a patent troll.

In the first round of litigation, Sable filed, and later promptly settled, eight lawsuits asserting infringement of Sable's router patents. The defendants in those cases (including Cisco and Juniper Networks) provide a range of Internet services, but they all at least manufacture and sell network equipment.

Interestingly, all of those cases were settled just before Sable would have had to do two things that would have actually put its legal claims to the test: (1) respond to an administrative proceeding before the US Patent & Trademark Office ("USPTO") challenging the validity of its patents; and (2) attend a hearing before the district court where the judge would have determined the proper interpretation and scope of the patent claims. So Sable filed cookie-cutter cases against eight defendants, waited for the defendants to respond, then settled the cases before meaningfully litigating its claims or facing a binding court or administrative ruling, which may have addressed, or likely undermined, Sable's overly-broad assertion of those patents.

Shortly after settling the original eight cases earlier this year, Sable turned around and filed six new lawsuits against a new batch of technology companies, this time including Cloudflare. Unlike the earlier named defendants, Cloudflare is not in the business of making or selling routers or switches. Sable's infringement claim therefore is not a close one, and now it's picked a defendant that is eager to fight back.

This case is a good illustration of how patent trolls operate. All four patents asserted by Sable *against Cloudflare were filed between 2000 and 2004, when dial-up Internet access was still common, and are based on Caspian's "flow-based routing" technology, which is nothing like the technology Cloudflare's products and services employ. To take one example, one of the patents Sable asserts is U.S. Patent No. 6,954,431, entitled "Micro-Flow Management." This patent is from April 19, 2000 — almost exactly 21 years ago. Just like Caspian's Apeiro routers from 2003, the '431 patent discloses a router that puts a label on the packets for a given flow, and forwards all the packets in that same flow based on the label:*

It claims to teach a "[n]ew switching technology [that] relies upon state information for providing a previously unavailable degree of quality of service" — presumably, Caspian's flow-based routing technology featured in its 2003 Apeiro, which the market rejected over a decade ago.

Sable is now trying to stretch the patents way beyond what they were ever meant to cover. Many of Sable's infringement claims appear to extend to basic routing and switching functionality known long before any alleged invention date. For the '431 patent, Sable's interpretation of the patent appears to stretch the scope so broadly as to cover any kind of packet processing, possibly even the "conventional routers" from the 2000s that routed each packet independently.

Having made this leap, Sable has demonstrated an intent to apply its patents far afield. It's now a small step for Sable to assert claims against any person or business using a firewall or even a router. On the basis of its logic, any person using a WiFi router in their home may be in Sable's cross-hairs. And Sable has already claimed that its patents cover firewalls — including firewall software and firewall devices. Again, its wildly broad interpretation threatens not just large companies; anyone trying to protect their networks from outside threats is potentially at risk.

Since Patents are all in the public domain, searching for Prior Art is quite engaging. If I weren't committed to SpinRite I might make some time. But I'm not going to. Instead, I wanted to let our listeners know about the CloudFlare prior art contest. It's this week's shortcut of the week to make it easy to find:

https://blog.cloudflare.com/project-jengo-redux-cloudflares-prior-art-search-bounty-returns/
https://grc.sc/816

Closing The Loop

Krv / @Krv (via DM) : You asked for someone's FLoC id, here is mine: You are FLoCed!
Your FLoC ID is 5393

SpinRite

As I hoped, the 3rd work-in-process testing release of SpinRite v6.1 went public last Thursday the 22nd. And it fared very well considering that it incorporated two months of work that hadn't been tortured at all. The only problem that the testing gang found was with some older machines with diskette drives. Since SpinRite can still boot from a floppy and can log its results to a floppy, that needs to work correctly. And now it does.

The Mystery of AS8003

Since inauguration day, January 20th of this year, those who run the Internet have been puzzled by a deep mystery for which no answers were available. It all began on that Wednesday in January when a surprising BGP message arrived from a previously unknown entity advertising that they would henceforth be receiving all traffic for all 16,777,216 IPv4 addresses beginning with 11:

```
TIME:        01/20/21 16:57:35
TYPE:        BGP4MP/MESSAGE/Update
FROM:        62.115.128.183 AS1299
TO:          128.223.51.15 AS6447
ORIGIN:      IGP
ASPATH:      1299 6939 6939 8003
NEXT_HOP: 62.115.128.183
ANNOUNCE
  11.0.0.0/8
```

Today, more than three months later there's still much that we don't know. But this weekend we learned a bit more. Still, I'm getting ahead of my story. Here's how the Washington Post began their coverage of this mystery:

> *While the world was distracted with President Donald Trump leaving office on Jan. 20, an obscure Florida company discreetly announced to the world's computer networks a startling development: It now was managing a huge unused swath of the Internet that, for several decades, had been owned by the U.S. military.*
>
> *What happened next was stranger still.*
>
> *The company, Global Resource Systems LLC, kept adding to its zone of control. Soon it had claimed 56 million IP addresses owned by the Pentagon. Three months later, the total was nearly 175 million. That's almost 6 percent of a coveted traditional section of Internet real estate — called IPv4 — where such large chunks are worth billions of dollars on the open market.*
>
> *The entities controlling the largest swaths of the Internet generally are telecommunications giants whose names are familiar: AT&T, China Telecom, Verizon. But now at the top of the list was Global Resource Systems — a company founded only in September that has no publicly reported federal contracts and no obvious public-facing website.*
>
> *As listed in records, the company's address in Plantation, Fla., outside Fort Lauderdale, is a shared workspace in an office building that doesn't show Global Resource Systems on its lobby directory. A receptionist at the shared workspace said Friday that she could provide no information about the company and asked a reporter to leave. The company did not respond to requests for comment.*

The only announcement of Global Resource Systems' management of Pentagon addresses happened in the obscure world of Border Gateway Protocol (BGP) — the messaging system that tells Internet companies how to route traffic across the world. There, messages began to arrive telling network administrators that IP addresses assigned to the Pentagon but long dormant could now accept traffic — but it should be routed to Global Resource Systems.

Okay. So the stage is set for this mystery. Let's step bank and examine this from the perspective of someone who runs the Internet at the BGP level. His name is Doug Madory and he's the director of Internet analysis for Kentik. His recent blog posting is titled: "The Mystery of AS8003"

Before I share Doug's description of what happened and what he saw and thinks, I'll remind our listeners about the odd BGP nomenclature. Within the weird world of Inter-Autonomous-System routing, an autonomous system is said to "advertise" or "announce" that it is the destination for all Internet traffic within one or more ranges of IP space. The claimed owner of the address space uses their own router to communicate to all of the routers it's connected to using the BGP — Border Gateway Protocol. They update their own routing tables to incorporate that new block of IP routing and their router, in turn, forward any changes that resulted to the routers they are connected to. The upshot of this is that an advertisement quickly propagates throughout the Internet, adjusting all other routers, where needed, so that any packet that's dropped onto the Internet anywhere will be routed from one router to the next until it reaches its new destination.

And, as we have discussed several times before, a simple slip of the finger when updatithose crucial tables — or deliberate shenanigans — can raise quite a ruckus across the Internet as large blocks of traffic are rerouted from their intended destination. So, with that in mind, here's what Doug Madory experienced:

On January 20, 2021, a great mystery appeared in the internet's global routing table. An entity that hadn't been heard from in over a decade began announcing large swaths of formerly unused IPv4 address space belonging to the U.S. Department of Defense. Registered as GRS-DoD, AS8003 began announcing 11.0.0.0/8 among other large DoD IPv4 ranges.

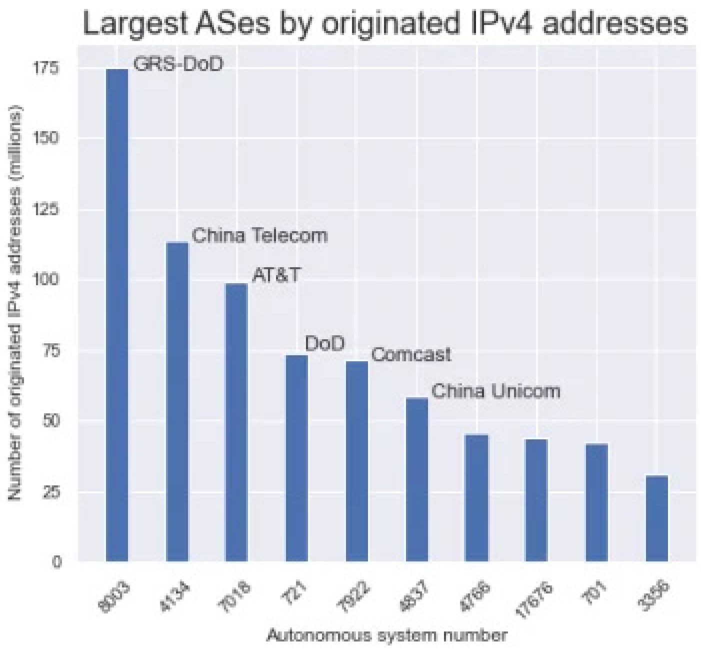
The message bore a timestamp of 16:57 UTC (11:57am ET) on January 20, 2021, moments after the swearing in of Joe Biden as the President of the United States and minutes before the statutory end of the administration of Donald Trump at noon Eastern time.

The questions that started to surface included: Who is AS8003? Why are they announcing huge amounts of IPv4 space belonging to the U.S. Department of Defense? And perhaps most interestingly, why did it come alive within the final three minutes of the Trump administration?

By late January, AS8003 was announcing about 56 million IPv4 addresses, making it the sixth largest AS in the IPv4 global routing table. By mid-April, AS8003 dramatically increased the amount of formerly unused DoD address space that it announced to 175 million unique addresses. [175 million IPs is 1/25th of the Internet's entire 4.3 billion IPv4 space.]

Following the increase, AS8003 became, far and away, the largest AS in the history of the internet. By comparison, AS8003 now announces 61 million more IP addresses than the now-second biggest AS in the world, China Telecom, and over 100 million more addresses than

Comcast, the largest residential internet provider in the U.S.



In fact, as of April 20, 2021, AS8003 is announcing so much IPv4 space that 5.7% of the entire IPv4 global routing table is presently originated by AS8003. In other words, more than one out of every 20 IPv4 addresses is presently originated by an entity that didn't even appear in the routing table at the beginning of the year.

A valuable asset: *Decades ago, the U.S. Department of Defense was allocated numerous massive ranges of IPv4 address space - after all, the internet was conceived as a Defense Dept project. Over the years, only a portion of that address space was ever utilized (i.e. announced by the DoD on the internet). As the internet grew, the pool of available IPv4 dwindled until a private market emerged to facilitate the sale of what was no longer just a simple router setting, but an increasingly precious commodity.*

Even as other nations began purchasing IPv4 as a strategic investment, the DoD sat on much of their unused supply of address space. In 2019, Members of Congress attempted to force the sale of all of the DoD's IPv4 address space by proposing the following provision be added to the National Defense Authorization Act for 2020:

Sale of Internet Protocol Addresses. Section 1088 would require the Secretary of Defense to sell at fair market value all of the department's Internet Protocol version 4 (IPv4) addresses over the next 10 years. The proceeds from those sales, after paying for sales transaction costs, would be deposited in the General Fund of the Treasury.

The authors of the proposed legislation used a Congressional Budget Office estimate that a /8 (16.7 million addresses) would fetch $100 million after transaction fees. In the end, it didn't matter because this provision was stripped from the final bill that was signed into law - the Department of Defense would be funded in 2020 without having to sell this precious internet resource.

What is AS8003 doing? *Last month, astute contributors to the NANOG (**N**orth **A**merican **N**etwork **O**perators' **G**roup) listserv highlighted the oddity of massive amounts of DoD address space being announced by what appeared to be a shell company. While a BGP hijack was ruled out, the exact purpose was still unclear. Until yesterday when the Department of Defense provided an explanation to reporters from the Washington Post about this unusual internet development. Their statement said:*

> Defense Digital Service (DDS) authorized a pilot effort advertising DoD Internet Protocol (IP) space using Border Gateway Protocol (BGP). This pilot will assess, evaluate and prevent unauthorized use of DoD IP address space. Additionally, this pilot may identify potential vulnerabilities. This is one of DoD's many efforts focused on continually improving our cyber posture and defense in response to advanced persistent threats. We are partnering throughout DoD to ensure potential vulnerabilities are mitigated.

I interpret this to mean that the objectives of this effort are twofold. First, to announce this address space to scare off any would-be squatters, and secondly, to collect a massive amount of background internet traffic for threat intelligence.

On the first point, there is a vast world of fraudulent BGP routing out there. As I've documented over the years, various types of bad actors use unrouted address space to bypass blocklists in order to send spam and other types of malicious traffic.

[Remember "Hamachi"? — that favorite and very clever ad hoc peer-to-peer networking system that we loved back in the day? It used the 5.0.0.0/8 network which was similarly unrouted back then. It was able to do this since no other public traffic would ever be going to any IPv4 address beginning with '5'. So any IP packets destined to an IP starting with '5' would be routed through the Hamachi peer-to-peer network.]

On the second, there is a lot of background noise that can be scooped up when announcing large ranges of IPv4 address space. A recent example is Cloudflare's announcement of 1.1.1.0/24 and 1.0.0.0/24 in 2018.

For decades, internet routing operated with a widespread assumption that ASes didn't route these prefixes on the internet (perhaps because they were canonical examples from networking textbooks). According to their blog post soon after the launch, Cloudflare received "~10Gbps of unsolicited background traffic" on their interfaces.

And that was just for 512 IPv4 addresses! Of course, those addresses were very special, but it stands to reason that 175 million IPv4 addresses will attract orders of magnitude more traffic. More misconfigured devices and networks that mistakenly assumed that all of this DoD address space would never see the light of day.

Conclusion: *While yesterday's statement from the DoD answers some questions, much remains a mystery. Why did the DoD not just announce this address space themselves instead of directing an outside entity to use the AS of a long dormant email marketing firm? Why did it come to life in the final moments of the previous administration?*

We likely won't get all of the answers anytime soon, but we can certainly hope that the DoD uses the threat intel gleaned from the large amounts of background traffic for the benefit of everyone. Maybe they could come to a NANOG conference and present about the troves of erroneous traffic being sent their way.

I want to add some additional information from the Associated Press' independent reporting. Toward the end of their longer article, the AP wrote:

What a Pentagon spokesman could not explain Saturday is why the Defense Department chose Global Resource Systems LLC, a company with no record of government contracts, to manage the address space.

The company did not return phone calls or emails from The Associated Press. It has no web presence, though it has the domain grscorp.com. Its name doesn't appear on the directory of its Plantation, Florida, domicile, and a receptionist drew a blank when an AP reporter asked for a company representative at the office earlier this month. She found its name on a tenant list and suggested trying email. Records show the company has not obtained a business license in Plantation.

Incorporated in Delaware and registered by a Beverly Hills lawyer, Global Resource Systems LLC now manages more internet space than China Telecom, AT&T or Comcast.

The only name associated with it on the Florida business registry coincides with that of a man listed as recently as 2018 in Nevada corporate records as a managing member of a cybersecurity/internet surveillance equipment company called Packet Forensics. The company had nearly $40 million in publicly disclosed federal contracts over the past decade, with the FBI and the Pentagon's Defense Advanced Research Projects Agency among its customers.

That man, Raymond Saulino, is also listed as a principal in a company called Tidewater Laskin Associates, which was incorporated in 2018 and obtained an FCC license in April 2020. It shares the same Virginia Beach, Virginia, address — a UPS store — in corporate records as Packet Forensics. The two have different mailbox numbers. Calls to the number listed on the Tidewater Laskin FCC filing are answered by an automated service that offers four different options but doesn't connect callers with a single one, recycling all calls to the initial voice recording.

Saulino did not return phone calls seeking comment, and a longtime colleague at Packet Forensics, Rodney Joffe, said he believed Saulino was retired. Joffe, CTO of Neustar Inc., which provides internet intelligence and services for major industries, including telecommunications and defense, declined further comment.

In 2011, Packet Forensics and Saulino, its spokesman, were featured in a Wired story because the company was selling an appliance to government agencies and law enforcement that let them spy on people's web browsing using forged security certificates.

[In the name "Packet Forensics" rings any bells for our listeners, that's why. We've spent a lot of time covering the covert use of TLS intersection "middleboxes", and Packet Forensics was among the purveyors of those technologies.]

The company continues to sell "lawful intercept" equipment, according to its website. One of its current contracts with the Defense Advanced Research Projects Agency is for "harnessing autonomy for countering cyber-adversary systems." A contract description says it is investigating "technologies for conducting safe, nondisruptive, and effective active defense operations in cyberspace." Contract language from 2019 says the program would "investigate the feasibility of creating safe and reliable autonomous software agencies that can effectively counter malicious botnet implants and similar large-scale malware."

Deepening the mystery is Global Resource Systems' name. It is identical to that of a firm that independent internet fraud researcher Ron Guilmette says was sending out email spam using the very same internet routing identifier. It shut down more than a decade ago. All that differs is the type of company. This one's a limited liability corporation. The other was a corporation. Both used the same street address in Plantation, a suburb of Fort Lauderdale.

So now everyone listening to this podcast knows as much as anyon else, aside from those who **do** know what's going on... and they are clearly choosing not to go on anyone's record.

It's interesting that the DoD had been squatting on that much super-valuable IPv4 address space. It might be that they are planning to finally put it to some use — though I agree with Doug that the timing of its sudden routing is quite suspicious. Under the principle of "use it or lose it" it might also be that they suffered a very close call with losing their precious allocation under the Trump administration and that they were worried that they might have less control during a Biden administration. I'm sure that not having that address space in active use would make it far more vulnerable to commandeering. So, if they are now able to say that it **is** in active use in support of our nation's cyber defense, because they have created the mother of all honeypots, I'm sure that would be enough to protect it from Congress.