The Mystery of AS8003

Description: 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.

High quality (64 kbps) mp3 audio file URL: http://media.GRC.com/sn/SN-816.mp3
Quarter size (16 kbps) mp3 audio file URL: http://media.GRC.com/sn/sn-816-lq.mp3

SHOW TEASE: It's time for Security Now!. Steve Gibson is here. Lots to talk about. We remember security researcher Dan Kaminsky. Steve has some personal memories to share and a little video. We'll talk about the University of Minnesota researchers who are in big trouble with the Linux kernel project. And then it's a look at the mysterious case of the AS8003, the Pentagon's massive IP address space. It's all coming up next on Security Now!.

Leo Laporte: This is Security Now! with Steve Gibson, Episode 816, recorded Tuesday, April 27th, 2021: The Mystery of AS8003.

It's time for Security Now!, the show where we protect your security and privacy online with our majordomo, Mr. Steve Gibson. Hello, Steverino of GRC.com. Good to see you.

Steve Gibson: Mr. Laporte.

Leo: Yes.

Steve: Good to be with you once again for Episode 816, the last one of April. What happened to April?

Leo: Boy.

Steve: It just kind of shot right by.

Leo: I'm kind of glad, though, because now I am fully immune. So I can go out and party with the rest of them, with the kids.

Steve: Nice.

Leo: Yeah, I'm excited.

Steve: This is going to be kind of a fun episode. I have two interesting big topics that will be fun to talk about. I titled it "The Mystery of AS8003," AS as in Autonomous System, and 8003 as a so-called Autonomous System number, which is something that all of the major players on the Internet are assigned. And I sort of wish that I had - I could have gotten one back in the early days. Mark Thompson was saying, oh, you've got to get yourself an AS number. Then ARIN will allocate a little Class C network, and those will be your IPs, and you'll own them, and you can transport them wherever you go. And if I were moving around much, that would have been useful. I think I've moved, what, like maybe twice in the entire life of GRC. So not so important. I don't really need 256 IPs.

Leo: Is that over? Can you do that still?

Steve: You probably can still do it. But I wouldn't be able to get - I don't know if you can be allocated a block smaller than a Class C.

Leo: Right, right.

Steve: And a Class C at 256 IPs, that's more than I need. Even as it is, I had - with Verio I had I think 64, and I'm at 16 now. And there is like a form, an IP justification form.

Leo: Yeah, I'm looking at it on the RIPE site. It says you have to have a contractual agreement with the RIPE NCC, which means either become a member or find a sponsoring LIR who will submit the request on your behalf.

Steve: Yeah. So it's involved.

Leo: Yeah. RIPE is the network coordination center, so you need their support.

Steve: Right. And so we've long talked about IPv4 address space depletion. So what happened, and what was really weird, was the timing. It was minutes before the end of Trump's presidency. After, well, okay, I'm getting ahead of myself. But anyway, this is really interesting, and we're going to have fun talking about that.

So but we've got something not fun to talk about. I want to start by, and we will in a minute, by remembering Dan Kaminsky, who the world lost last Friday at the tender age of 42.

Leo: So sad. So sad.

Steve: He had been struggling with diabetes his whole life, Type I diabetes, where the pancreas doesn't produce enough insulin. And a danger of unregulated blood sugar is that your body will go into diabetic ketoacidosis, which is an overproduction of ketones, which are acidic and deadly, and which your body produces when it starts to burn fat sort of out of control. Anyway, of course we know Dan so well on the podcast. We'll talk about that in a minute.

We're also going to finally catch up with this month's Patch Tuesday and look at a welcome maturation in Google's Project Zero vulnerability disclosure policy. I was really happy to see them announce this. We're also going to shine a light upon a new startup venture which, if successful, and boy, based on the creds of the gal who founded it, it sure ought to be, promises to dramatically improve the future of IoT security, which lord knows we need. We also then look at some controversial security research for which the researchers have apologized. And we'll wonder whether an apology was due. I'm not sure. I mean, I get it that the Linux maintainers are all upset. But maybe it was necessary.

We're also going to shine another light onto a new battle which Cloudflare has chosen to wage against, Leo, an abusive patent troll. Actually the court we know really well. We've talked about Judge Albright in the Western District of Texas and the abuse that is...

Leo: Yee haw.

Steve: ...occurring down there.

Leo: I'm sure it's not the case, but I just imagine him in a 10-gallon hat with six-guns going, "Let's get them patent - let's get them big shots [indiscernible]. Whooee." Geez, what a jerk.

Steve: Yeah, yeah. The background is really interesting because this guy apparently has just gone so far overboard now that - the problem, of course, is that his judgments, which are strongly pro-patentee, are not being challenged because it costs more money to challenge than it does just to say, okay, fine, we'll settle.

Leo: Yeah, yeah.

Steve: And of course that's the problem. Cloudflare says, "Eff no. We are not going to support this. We're going to fight it."

Leo: Good.

Steve: And so what they're doing, well, again, I'm getting ahead of myself. We'll talk about that. And our listeners get to play a part in this and make some money if they're interested in helping Cloudflare. And then, after a brief SpinRite progress report, we're going to engage with the Internet Mystery of the Autonomous System 8003.

Leo: I can't wait. I was hoping you'd talk about that.

Steve: A great podcast for our listeners.

Leo: Yeah, yeah, yeah.

Steve: So Dan Kaminsky cut a wide swath through the computer industry. He was a prolific tweeter and a real character and personality. He and I were last together, we followed each other onstage during DigiCert's first security conference. And Dan peppered me with some questions about SQRL back then, and I was able to satisfy his many salient questions. He was probably first on the map for this podcast when he realized in doing just some research that he was always up to that the transactions which all of the DNS servers throughout the industry were using had way too little entropy. Their port numbers for the queries they were generating were often sequential, so they were marching through the port space.

And often the transaction IDs, which is a 16-bit number that is used to associate queries with the replies when they come back, those were also sometimes a fixed, well, they weren't a fixed number. Ports were sometimes fixed. But the transaction IDs might just be an incrementing counter. And what Dan realized was that the lack of query entropy being emitted by DNS servers allowed replies to be spoofed. You could ask a DNS server yourself something and see where its counters were, and then induce somebody else to ask it a question and provide a spoofed reply before the real reply would get back. And because you knew where the counters were, you were able to, with high accuracy, get a spoof to be accepted as legitimate. And that's, you know, because the DNS runs over UDP, there is no TCP handshake to validate IPs, so you're able to completely spoof the replies.

So what this meant was, if the world were to realize that, as Dan had privately, it would be a catastrophe. So Dan privately got in touch with all the purveyors of the various DNS servers. They all recognized what he had. And privately, all of the servers were updated, and an industry-wide reveal was coordinated in order to maximize the probability of getting all this fixed before the bad guys had a chance to abuse it.

And of course because I recognized this was a problem, and we covered it on the podcast, we owe Dan the existence of my DNS Spoofability Service, which I created in honor of his discovery, which allowed individuals to go to GRC's DNS Spoofability. I arranged to cause, by setting up my own DNS, like pseudo DNS servers, I could cause a visitor to my site's DNS to use me as its resolver, and then I collected all the queries that I was inducing through that web page and analyze the nature of the queries coming from the DNS servers that the user is using.

Anyway, Dan has a large following. I think, what is it, I had it here in my notes somewhere, 94.3K followers on Twitter. As I said, he's a prolific tweeter. He joined Twitter in 2007. Since then he has posted 130,000 tweets. Now, if we assume an average tweet rate over 14 years, that's 9285.71 tweets per year.

Leo: Wow.

Steve: Or an average of 25.42 tweets or commented retweets per day.

Leo: That's amazing.

Steve: So if you were following Dan, you knew what he was thinking and doing. And he was also quite literate. He pinned a tweet of his from January 16th, 2018 to the top of his feed. He wrote this: "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."

Leo: That's really good. And I think it's absolutely right. Absolutely.

Steve: Just really, really good stuff. He had a site, DanKaminsky.com, which was his personal blog, and he hasn't blogged in about four years. But his last blog, I'll just share a couple paragraphs from it. He wrote: "Cryptographically Secure Pseudorandom Number Generators" - right, we've talked about them a lot, CSPRNGs - he says, "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?"

Leo: Uh-oh. Uh-oh. Uh-oh. Uh-oh.

Steve: "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?" So anyway, we have lost...

Leo: Wow. That's a beautiful, beautiful thing.

Steve: ...a critical thinker among us who made all manner of contributions to security and the Internet. He was working on some weird JavaScript stuff that I never really

tracked. But we have, and I wanted to play it into the podcast so that it is captured, a minute and 45-second video which he and his young niece produced 13 years ago, niece Sarah, following Black Hat 2008, which was where the DNS problem was first revealed. So here's - and this is fun because his niece is precocious and following a script that he produced. But they made a really fun minute and 45 seconds.

[BEGIN CLIP]

DAN KAMINSKY: I'm security researcher Dan Kaminsky, and I'm here today with my niece Sarah.

SARAH: Hi, everybody.

DAN: Hey, Sarah. So Sarah here has an important message for all of you.

SARAH: Fixing DNS is so important and so cool.

DAN: Well, what's DNS, Sarah?

SARAH: Well, Uncle Daniel, I think you should know.

DAN: Be that as it may, why don't you tell the people a little bit about it.

SARAH: Well, DNS is a Domain Name System. It tells my computer where on the Internet all my favorite websites are.

DAN: Was there something wrong with DNS?

SARAH: It'll be okay. Everyone got together a while back to make sure everything would work out.

DAN: Oh, everybody? Even ISC, the makers of BIND, and Microsoft, and Cisco, and Nomi, Nomi...

SARAH: You mean Nominum?

DAN: Totally Nominum. But that's really cool. So what should everyone do, Sarah?

SARAH: Well, this is really geeky stuff. But most people should get automatic updates and be okay.

DAN: Well, who might not?

SARAH: Well, there might be some servers that don't get automatic updates because they're really important, and people want to keep an eye on them.

DAN: Oh, so we should ask those people to take a look?

SARAH: Totally.

DAN: Oh, well, when should they look?

SARAH: Right now. Duh.

DAN: Well, when do they have to fix it by?

SARAH: Well, the attack is pretty weird, but people will probably figure it out after a month. So I'll give you an exact date: August 6, 2008.

DAN: August 6?

SARAH: August 6.

DAN: All right, then. Now, is there any way for the non-geeks to make sure they're safe?

SARAH: Only if you build them a website.

DAN: Hmm, I'll get right on that.

SARAH: You better.

DAN: Well, thanks, Sarah. And there you have it. Kids, talk to your parents about their DNS. They'll be glad you did. All right. That's a wrap.

CAMERA: All right, that's a wrap. Okay. Thanks, Cousin Dan. And thank you, Mattie.

[END CLIP]

Leo: That was so sweet. Oh, my god. Oh, I'm sure they miss him terribly. And of course his friend Steve wrote something in assembly to do that, so it was okay. He didn't have to do that. Wow, wow, wow.

Steve: Yeah, very cool.

Leo: Oh, he'll be missed.

Steve: Also DEF CON has announced on their Twitter feed that they're having an online memorial for Dan on Sunday, May 2nd, on the DEF CON Discord channel, Discord.gg/defcon.

Leo: Nice. Nice.

Steve: So they tweeted: "Come share your favorite stories and join us in celebrating the life of a hacker whose life elevated the whole community."

Leo: Wow. Good. Thank you for doing that, Steve. Yeah. Of course his name comes up all the time on our shows.

Steve: Yeah, yeah.

Leo: He'll be missed.

Steve: Yeah. Good guy. And for what it's worth, DanKaminsky.com. I only shared the top of that really cool posting. He gets really into it. So if you're wondering maybe if in fact we are in the Matrix, Dan may give you some pause. Of course I don't know how long the site will be up. Hopefully it will stay.

So Patch Tuesday was the week before last, and I missed summarizing it last week just because there was so much else to talk about. And it would be nice if it was all old news

by now. But believe it or not, the NSA contributed their knowledge of four additional remote code execution flaws in, you guessed it, Microsoft Exchange Server. Okay, now, I'm not a conspiracy guy. But you've got to wonder whether these four 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 worldwide, not just in the U.S., and it's now become a well-proven means of gaining surreptitious entry into someone else's system. That's exactly the sort of thing that the NSA or the CIA might find quite handy. And assuming that we're to believe the various leaks from the past, we know this is exactly the kind of thing that they do in fact hold onto and use, explicitly without telling the world, because they want their access ability to be retained.

But I was thinking that, with all the heat and attention on Exchange Server, they may 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, presuming that, if everyone was starting to look at it so closely now, they would get found anyway, or the bad guys could be using them against us. So, yeah, let's close the other backdoors that exist.

So, yes, two weeks ago, well after the early March updates, we got another set of these four RCEs closed. And again, that's all just wild speculation. It's equally likely that they may have put some of their own hackers onto Exchange Server for the sake of the U.S. and discovered four previously unknown flaws. That could be true, too. And in either case, as part of this month's 114 other security flaw repairs, Microsoft fixed these four, which of course affect all versions of Exchange Server - 2013, 2016, and 2019, presumably 2010, too, since all of these flaws seem to have been around forever. Two of the four of those four that the NSA found are fully juicy, unauthenticated access flaws requiring no user interaction, and thus they rank a CVSS of 9.8, right up there at the top.

Overall, Patch Tuesday fixed a total, as I mentioned, of 114 flaws: 19 were rated critical, 88 important, and one moderate. One of the most concerning was CVE-2021-28310, which is a privilege escalation vulnerability in Win32k. We've seen a lot of those over the course of the last year. This one is a worry because it is under active exploitation, which allows attackers to elevate privileges by running malicious code on the target system.

And at first you say, well, it doesn't get you in, so what? But Kaspersky Labs found and reported this one to Microsoft. Their Boris Larin said it's a privilege of elevation exploit that is likely used together with other browser exploits to escape sandboxes to get system privileges for further access. And just a couple weeks ago we were talking about the Pwn2Own, the most recent 2021 Pwn2Own competition, where there were exactly this sort of chain of exploits used to break out of browsers and get to people's computers. And we've also talked about how the browser is the main entry point now - okay, Microsoft Exchange Server aside - for intrusions into people's systems. It is the target of opportunity. So anyway, good to get those patched. And it was, yes, another big Patch Tuesday.

Google's Project Zero responded to, in my opinion, today's patch latency reality. And I was really glad to see that. As we know, until now, Project Zero set a fixed 90-day timer on the disclosure of vulnerabilities discovered and privately disclosed to vendors. And we've shown the Project Zero postings where they announce that there's a vulnerability, they say nothing about it, but they say if it's not fixed within in 90 days, we're going to tell the world because, A, the world maybe needs to know so they can fix it themselves. That's of course dicey because bad guys now jump on that and abuse it before it can be fixed, mostly obviously to put pressure on vendors so they don't just sit around twiddling their thumbs and don't deal with problems that have been found and could presumably easily be fixed.

Oh, and also remember that when a vulnerability is discovered in the wild, that is, when it is a true zero-day, and by that we mean it's in use, it's completely unknown at the time that it is seen being abused. As we know, the term "zero-day" has been watered down recently. Everyone now just uses it because it seems more exciting. But in real zero-day instances, Project Zero sets the timeout, the disclosure time to seven days, one week. So that's meant to reflect the extreme danger posed to users of the fact that this thing is happening now, and to really light a fire under the vendor to get them moving on fixing it with some urgency.

What's changed in what I think is a welcome and sane announcement last week is that they're going to add a new 30-day patch latency allowance, which makes so much sense. If the vendor fixes the problem and publishes a patch before the initially allotted deadline expires - whether it's 90 days for a problem found but not known to be under abuse, or seven days for a true zero-day which is discovered - if it's fixed and patched within that deadline, then Google will add a 30-day grace period onto the end of the original deadline to reflect the reality that we're all observing now that releasing a patch and actually having systems patched against the now-known vulnerability are two very different things. So this is great. And clearly some rethinking has been done.

There is some discussion about maybe tightening up on that 90 days, that initial 90-day deadline, saying that's three months. Come on, folks. Do you need three months to fix a problem? Even Microsoft with a 30-day iteration, give them 60. In case they miss the first one, they ought to be able to catch the second one. So anyway, we'll see how that goes. But it really makes sense. What we keep seeing now is that the moment the details of a flaw emerge publicly, bad guys jump on it and immediately start abusing it because they know not everyone is going to get patched quickly. So 30 days from the time that the patch is available, no matter when that occurs, that's going to allow a 30-day patching cycle, like many people in the industry now have, to hit its patch event and allow automatic updates to happen, so even users who aren't doing anything. And this allows the vendors not to have to force an emergency update cycle, which many systems really don't support now. So bravo to Google for fixing that.

The other thing I wanted to mention is I'm excited about this. A newly financed startup named Thistle, T-H-I-S-T-L-E, aims to take aim at IoT security by providing a secure turnkey security foundation to future devices. For more than 20 years, this new firm's founder, by the name of Window Snyder, S-N-Y-D-E-R, she's been building security into the products of some of the largest companies in the world. Her startup is creating tools that will help manufacturers build security into connected devices from day one.

As we know, the manufacturers of, well, everything - printers, ATM machines, 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 last several decades. And the result is all too well known to us. We see billions of devices shipping with vulnerabilities that are preyed upon by profit-driven bad guys and nation-state hackers. I would argue that the IoTpocalypse hasn't yet been seen.

But when you imagine, and I've talked about this, all the connections which are reaching out from all of our devices in all of our homes over to services in China which anchor these things, wow. And then you look at things like all of the IP stacks which these things use which are then known to have, be riddled with vulnerabilities, yet they're never able to be updated because it's a wall switch or a plug. We're just, you know, like I said, you want this on its own network.

So Window, her first name, is a security veteran. She has previously served as Chief Security Officer at Square, Mozilla, and Fastly, and was Chief Software Security Officer at Intel. While she was a teenager she was part of a Boston hacker collective before going

on to be a consultant at @stake, which is a security company which employed many of the members of L0pht. Remember we used to talk about L0pht 15 years ago, L-0-P-H-T, that was another Boston hacker collective. She also spent time at Microsoft working on Windows XP Service Pack 2, which was the update which added a range of really much-needed security improvements to Windows. She worked on security at Apple. So she knows what she's talking about. About the new company she was quoted saying: "What it takes to build security into products requires a lot of really specialized skills. You get folks, especially at the device level, building the same security mechanisms over and over again, reinventing the wheel, and doing it to different levels of resilience."

So her 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.

Leo: Yeah, good.

Steve: I mean, has she been listening to this podcast? Patching devices typically requires reflashing firmware, a process that can be fraught with risk. And as Window notes, she said: "It's one of the reasons 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% failure rate, which is very low for updates, that's still a million devices that are bricked." So this is wonderful news. I hope she succeeds. I wanted to put her on the map through this podcast. If any of our listeners are in any position to influence the development of any IoT work, check out Thistle because farming this stuff out to a security specialist who's got this kind of CV behind herself sure seems like exactly the right thing to do, in my opinion. So bravo. And in advance, thank you, Window.

Okay. So the University of Minnesota, I'm inclined to label this "controversial," though many don't believe there's any doubt about this being unequivocally wrong. Okay. So 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 sneaked 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. So says the press. A closer reading will show that that's not actually what they did. But, okay. So we'll take this one step at a time. The research was conducted earlier this year, or at least it was published, well, it had to have been last year because it was published late last 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 [what they called] 'hypocrite patches.'" And I'll explain that terminology in a second with their paper.

So the researchers claimed: "We did not introduce or intend to introduce any bug or vulnerability in the OS." Okay. But it appeared that evidence emerged to the contrary, suggesting that the research was conducted without adequate oversight and risked the

Linux kernel's security. And this resulted in a unilateral ban of all future code submissions from anyone using the "umn.edu" email address, and also invalidated all past code submitted by the university's previous researchers.

Leo: Ouch. Wow. I don't blame them.

Steve: Oh, Leo.

Leo: I'd be pissed, too.

Steve: So, yeah, the Linux Kernel Project did not take this lightly.

Leo: No, they're a little irritated, I'm thinking. I'm guessing.

Steve: They were, yes, they were, it would be fair to say, pissed. 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." So responding to the Linux Project's response, another observer thought it was even worse, tweeting: "This is worse than just being experimented on. 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."

Leo: Yeah, yeah.

Steve: They said: "Enormously unethical."

Leo: Yes.

Steve: Okay. And 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 process by which this research method was approved, determine appropriate remedial action, and safeguard against future issues."

Leo: As in, why didn't somebody stop these knuckleheads?

Steve: Okay. So what about this research? Their paper, which has been accepted for publication and presentation at the IEEE Symposium on Security and Privacy 2021, is titled "On the Feasibility of Stealthily Introducing Vulnerabilities in Open Source Software via Hypocrite Commits." And the paper explains itself in its abstract, which reads: "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 open source software boost its adoption, while its openness and flexibility enable quicker innovation. More importantly, the open source software 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 investigate the insecurity of open source software from a critical perspective: the feasibility of stealthily introducing vulnerabilities in open source software 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 numbers of devices.

"We first identify three fundamental reasons that allow hypocrite commits. One, open source software is open by nature, so anyone from anywhere" - except now the University of Minnesota - "including malicious ones, can submit patches. Second, due to the overwhelming patches and performance issues, it is impractical for maintainers to accept preventive patches for 'immature vulnerabilities.'" And they talk about that a little bit later. "And, three, open source software 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" - and actually they do it in vivo, right, which was the problem - "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 open source software 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 open source software and developing tools for patch testing and verification."

Okay. Now, I would argue that this is a very valid and very important and very much needed avenue of research. It should be clear to everyone, and apparently a bit of a sore spot with the Linux kernel maintainers, that surreptitious commits would inherently be a huge problem for any large open source project which is open to many contributors. The only solution I can see is being extremely, you know, is to have extremely careful multiparty scrutiny of any changes which are made to the kernel. But that's rather thankless and boring work. It's much more fun to create new patches and apply them. But careful scrutiny of changes made to the kernel that are predominantly going to be fine sort of 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 I've often spoken of, of debugging one's own code, which similarly is believed to be correct. As I've often observed, it often takes single-stepping through such code which you "know," in quotes, is correct, even though it's misbehaving. You finally have the debugger just rub your face in the mistake before you see it. And it inevitably evokes the "aha" reaction that anybody who's written code and has been forced to have a debugger show them where the problem is has experienced. 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. So it represents an ongoing Achilles heel for any large open source projects. So I would argue that this was important and useful research.

In a follow-up FAQ, these guys attempted, as a consequence of the backlash that their in vivo experimentation with the actual Linux project incurred, they attempted to clarify what they had done. Their FAQ is titled: "Clarifications on the hypocrite commit work." And I've got a link to that also in the show notes. I'll just share their first introduction of it, which really does put this into context.

They said: "We recently finished a work that studies the patching process of open source software. Its goal is to improve the security of the patching process. The corresponding paper has been accepted by IEEE S&P 2021." He says: "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." You know, in other words, yeah. If you only read the abstract, you don't realize what it is they did. And so the people who got all upset, and the press, who apparently also just followed the upset, didn't pursue the details. Which, as you know on this podcast, we find is often valuable.

So he says: "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 our first priority. In the past several years, we've devoted most of our time to improving the Linux kernel, and we have found and fixed more than 1,000 kernel bugs." So, yeah, these guys know what they're doing.

They said: "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 open source software contributors. We respect open source software volunteers and honor their efforts. We have never intended to hurt any open source software or users. We did not introduce or intend to introduce any bug or vulnerability in open source software. The following are clarifications to the common concerns we received."

So first, the purpose and research value of the work. They said: "The project aims to improve the security of the patching process in open source software. As part of the project, we study potential issues with the patching process, including causes of the issues and suggestions for addressing them. This study indeed reveals 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 open source software 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 quantitative analysis; and, more importantly, provide suggestions for 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."

So then they ask themselves the question so they can answer it: "Did the authors introduce or intend to introduce a bug or vulnerability?" Answer: "No. As 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 for the experiment."

And it continues. Again, link to the PDF for anyone who's interested. I don't find any fault with what these guys did. And 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 the facts. I haven't taken the time to study this more deeply, nor look at the maintainers' 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.

Maybe they could have said, in fact somewhere I read that they were worried that if they asked for permission, they would be told no and foreclose what is arguably a valuable piece of research. So if they fixed a thousand bugs in the Linux kernel, and the Linux guys are so upset that they're saying they're going to, what, revert a thousand Linux improvements because they all came from these guys when they were doing this research? Well, I mean, before this research. And it was the act of fixing the bugs that led them to realize, hey, there's some problems here in this process that we've been participating in that we need to understand better. And I think it's wrong to slap the university like this. That was, you know, hopefully this is an overreaction that can be backed out of.

The paper finishes, their conclusion, you know, as all good research papers have, they concluded saying: "This paper presented hypocrite commits, which can be abused to stealthily introduce vulnerabilities in open source software. Three fundamental reasons enable hypocrite commits: the openness of open source software, which allows anyone including malicious committers to submit patches; the limited resources of open source software maintaining; and the complexity of open source software programs, which results in the manual review and existing tools failing to effectively identify newly introduced vulnerabilities.

"We then systematically characterized immature vulnerabilities and studied how a malicious committer can turn immature vulnerabilities into real ones." And now we understand what that means is an inadvertently introduced vulnerability as part of an intended fix which is sort of latent. And then another deliberate "improvement" can mature that immature vulnerability into one that can actually be leveraged. So, I mean, these guys know their stuff.

And they said: "We also identified multiple factors that increase the stealthiness of the introduced vulnerabilities, including concurrency, error paths, aliases, indirect calls, and so on. 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 and patching the process of open source software." So to these guys I say thank you. Yeah, I think on balance the reaction has been wrong from the maintainers. And again, as I said, I hope this gets some potentially well-deserved attention.

Leo: I'm looking at the banning notice. And it says commits from these addresses have been found and submitted in bad faith. So they say it's just an email, but it sounds like they submitted commits. Having commit privileges, especially to something as critical as the Linux kernel is, that's a real privilege. Very few people have it. Very few people have it for this very reason. If these guys have commit privileges, and attempted to commit a flawed update, I'd block them, too. They'd never get to commit again. That's a privilege. There's a reason they call it a privilege. The maintainers say these guys were trying to commit.

Now, it says "Commits from a @unm.edu address have been found to be submitted in bad faith." So there's a dispute over what the facts of the matter are. I mean, writing an email, no big deal. Big deal, you know. What about this? What do you think? Big deal. But if you have commit privileges, especially the Linux kernel, that's

a big deal. Very, very, very few people have that. And you in bad faith submit something that's to test the system, I'd revoke your privileges. I think that's completely appropriate. You can't be trusted. I think your ethics are definitely in question. So I'm looking at the email. I think that maybe there's other things. I don't know what's going on because it doesn't match what they said.

Steve: So what we have is clearly a collision of facts where these guys are saying, as I read, we absolutely never endangered the kernel. The kernel guys are saying, well, you could have.

Leo: Not true. They submitted it, yeah.

Steve: And we no longer trust you.

Leo: So they're reverting all submissions from that group from the kernel tree, as they should. The patch set has easy reverts, but there are 68 remaining ones that need to be manually reviewed. And they must all be reviewed at this point. Some of them are not able to be reverted as they'd already been reverted or fixed up with follow-on patches as they were determined to be invalid. Proof that these submissions were almost universally wrong. So there's a lot of work that has to go on at this point.

Steve: Yeah.

Leo: I'm not - I don't - I wouldn't say nothing from the University of Minnesota will ever be accepted again. I don't think that's what's happened here. I think that these people apparently had commit privileges which have been revoked, as they should have been. And they should have known better. That's inappropriate. And by the way, everybody knows these flaws exist, that this is a problem. That's why it's so hard to get commit privileges. And there are so many stages to get approval to get to the kernel. The Linux kernel is not the problem. It's all those other little things that we've talked about before that have one maintainer out there who's overworked and underappreciated. Those are the problems. It's OpenSSH. It's not the Linux kernel. That's pretty well protected. Maybe that's why they attacked it. But I don't care what your motive for attacking the Linux kernel is. That's not okay. I don't think that's okay. That's my opinion. So just thought I'd throw that in there.

Steve: I'm glad for it, Leo. We have multiple views.

Leo: Yeah. And I think we need to know more about what really happened. I think that's part of it. I just don't think that it was that useful, what they've so-called proven. We know that.

Steve: Cloudflare. Our favorite company.

Leo: Oh, this is an interesting story, yeah.

Steve: Yeah, an interesting favorite company of ours. On March 15th - I'm reading from Cloudflare's announcement of what went on to set the stage. "On March 15th," Cloudflare wrote, "Cloudflare was sued by a patent troll called Sable Networks a company that doesn't appear to have operated a real business in nearly 10 years relying on patents that don't come close to the nature of our business or the services we provide."

They said: "This is the second time we've faced a patent troll lawsuit. As readers of the blog, or followers of the tech press, ZDNet and TechCrunch, will remember" - and I'm sure we talked about it at the time. They said: "Back in 2017 Cloudflare responded aggressively to our first encounter with a patent troll, Blackbird Technologies, making clear we would not 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.'"

They said: "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 (J-E-N-G-O), a crowd-sourced 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 then 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."

They said: "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 light years beyond what they were meant to cover. It has already sued over a dozen technology companies" - and I don't think I have it here, but like Cisco and Juniper Networks won settlements. That's what these people do; right? It's more expensive to fight the lawsuit than it is just to pay these people off. So that's their profit model.

And Cloudflare says "eff no." They said: "It's 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 patent 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," they wrote, "by offering $100,000 to be shared among the winners who are successful in finding such prior art." They said: "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." And then they cite Sable Networks, Inc. v. Cloudflare, Inc. And they have the case number. And then I noted ADA, and then it says in parens (W.D. Tex.). Well, ADA is Alan D. Albright, a.k.a. the infamous Judge Albright, who is exceedingly patentee friendly. And his jurisdiction is W.D. Texas, the Western District of Texas.

They said: "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."

So anyway, their post goes on at some length. And it's all really interesting to anyone who has a passion, as I do - and I know, Leo, you do, we've talked about this a number of times on the podcast - for issues surrounding intellectual property rights and the abuse thereof, unfortunately by the U.S. patent system. And in the posting, I have a link in the show notes, they explain how Sable sues companies, then settles out of court just before the deadline to actually make their case. It's pure patent trolling harassment. And it makes our blood boil. And I have more here. I don't think there's anything else relevant that I haven't talked about.

Leo: Yeah, we went through this almost exact situation with the podcast patent troll.

Steve: Oh, the podcast troll; right.

Leo: Yeah. And, you know, Cloudflare's offering a reward of $100,000 for prior art, which is of course what you want to find. There are a couple of ways you can fight this stuff. And it's interesting, you know, we actually hired a law firm to prepare us for this because we got a demand letter from these guys for I can't remember, million dollars or something. The EFF chose one path, which was you can do an inter partes challenge of the patent itself with the U.S. Patent & Trademark Office. So you go to them, and you say, here's prior art. This is evidence that this patent was...

Steve: Was not original.

Leo: Was not original. Others had done this before. And you hope that the PTO will overturn the patent, and then the whole thing goes up in smoke. There's a risk, our attorneys told us, if you lose, that's really going to prejudice the case against you because that loss with the Patent & Trademark Office will be brought up at court.

Steve: Oh, further strengthens the patent, yes.

Leo: So it's a risky process, which we had decided not to pursue. Our attorney said just ignore it until they sue you. They did sue a number of people, including Adam Carolla, who raised money and fought it. He did what Cloudflare's going to do. He took it to court, and they lost in court. But actually maybe it was the other way around. No, I'm sorry, the EFF did the inter partes and won, the Patent & Trademark Office, as this parallel Adam Carolla court case was going on.

Steve: Ah.

Leo: And the PTO overturned the patent, and then the whole thing was over. So but Adam Carolla spent a lot of money, I think. I mean, he raised the money from contributors. But he was going to fight this. And that's what you have to do because

what happens is the little guys give them money, and then they go to the next big guys, they get more money, and they're building up a war chest to eventually go after Apple and Google and whoever and get the big bucks. And so at some point somebody has to say, no, we're going to court. And it's funny because Lisa, I love Lisa for this, she never - she says, "You don't ever settle because once you settle..."

Steve: [Crosstalk] flames.

Leo: We'll go down in flames. She said, "Once you settle, the word goes out. Oh, yeah, they'll settle." So we never settle. I should just - everybody should know that. We go to court because that's the only way you can stop this. You just have to fight it. And the reason it works is because they usually ask - this patent troll was dumb with us. They usually ask for an amount that's less than the cost of fighting it.

Steve: Right, right.

Leo: Our attorney said, "It's going to cost you about a quarter of a million to fight it." He asked for a million dollars. Stupid. He should have asked for $240,000, and then the reasonable intelligent business thing to do, we still wouldn't have done it, would have been to say okay. But we've been sued a couple times since then for amounts just low enough so that you go, oh, here. And in every case we fight it, even though it costs us more, because it's just not - and we've always won, by the way.

Steve: It's wrong.

Leo: It's wrong.

Steve: Yeah, yeah. And so it is, you know, a lot of people assume that a patent means something.

Leo: It means nothing. It means the right to defend it in court.

Steve: Yes. A patent is literally a license to be sued.

Leo: Yeah.

Steve: First of all, anybody can patent anything that they want to. Doesn't even have to - it doesn't have to make sense. It doesn't have to be reasonable. And our Patent Office is, I mean, it's a hard job. And so I have some sympathy for them. But the presumption is that they are really doing due diligence and making sure that something that someone submits as claiming that, hey, I invented something, more often than not it's just engineering. I look at these patents, and anybody who came out the other end of a university with a degree in the subject would go, well, that's the way you solve this problem.

Leo: Right. It's obvious. It's an obvious solution, yeah.

Steve: Yes. I mean, Microsoft was, as a technologist, as a coder, I looked at many of Microsoft's patents early on. And the only thing that happened was that they faced a problem before somebody else. And anyone trained in the art, and that is actually the language of the patent law, it is supposed to be non-obvious to anyone trained in the art. Meaning that some other programmer is given this problem, they go, oh, here's how you solve that. Well, Microsoft was leading the development of software, so their programmers encountered problems that other programmers hadn't encountered. Well, they patented everything. I mean, patent sneezing to the right. And, oh, because most people don't do - it's like it was insane.

Leo: And the presumption is, they even say this, the PTO says this, we expect if there's an issue that it'll go to court, and it'll be solved there. And they just, I think, probably don't have the examiners and the time to do it right. So they do the best they can, but ultimately all they're saying.

Steve: And what mature companies end up doing, and this is what they will say when you challenge them is, well, we're building a patent portfolio so that we can cross-license other large companies' patent portfolios. And so it's sort of an insider large corporate thing that goes on. And unfortunately the little guy is the one who ends up being in trouble.

So anyway, bravo to Cloudflare. I wanted to put this on our listeners' radar. I've got a link in the show notes. In fact, even in my show notes I go on at great length about Caspian and this other company, Sable. For anyone who's interested can just read more than I'm going to put into the podcast because there's no need to go into additional detail. But Cloudflare is worried that, if this company is allowed to keep going, they're going to start suing people who process packets because there's this notion of tagging packets to be part of a flow, which was a technology that the firm that went out of business, Caspian, tried to market this thing, and they probably went belly up. Sable probably bought their intellectual property portfolio out of bankruptcy with the intention of pursuing this.

Anyway, so what Cloudflare is worried about is that these guys really do need to be stopped. Secondarily to the fact that it's wrong, and that also that Cloudflare's been sued. And of course it's a good idea, it's prudent from Cloudflare's standpoint to send the message out that you sue us, we're going to invalidate your entire patent portfolio, you troll, and put you out of business. So anyway, cool stuff.

Leo: Yeah. "Non-practicing entity" is the polite name.

Steve: Yes, NPE, non-practicing entities. We even have a phrase and an acronym for these slime balls.

I wanted to note, we got a bit of feedback from a listener, somebody who had been FLoCed. I put the call out now two weeks in a row. I went back a little bit into my Twitter feed. And Krv, wow, I guess he's been in Twitter for a while, literally his Twitter handle is @Krv.

Leo: Three letters. That's good. That's old school.

Steve: That's a goodie. Anyway, he said: "You asked for someone's FLoC ID. Here is mine." And he said: "You are FLoCed! Your FLoC ID is 5393." So that's cool. That says, I mean, we don't know what the maximum ID is. But the fact that he got 5393 suggests large pools with a few number of IDs. But again, Leo, as you've said, all subject to change, and this tells us nothing at this early stage. So definitely something to keep an eye on. And did you know, I was following some trails, Microsoft has their own proposal. And when I saw that it was called Parakeet, and it was based on Turtledove, I said, okay, no, everybody.

Leo: The birds. The birds.

Steve: Enough with the birds. Really. This is for the birds.

In a brief note, as I hoped, the third work-in-progress testing release of SpinRite 6.1 was taken public last Thursday. I found that bug that I mentioned that I was going to pursue after last week's podcast. And it fared very well, considering that it incorporated two months' worth of work that hadn't been tortured at that point at all. The only problem that the testing gang found was with some older machines with floppy disk drives. Since SpinRite can still boot from a floppy and can log its results to a floppy, that all needs to work correctly. Floppies are weird, and they've always been weird because remember, Leo, you will, the very first PCs did not have hard drives. And they often did not have two floppy drives.

Leo: Oh, yeah.

Steve: You could get single-drive PCs.

Leo: Yeah.

Steve: But how could you copy a floppy if it only had one floppy drive?

Leo: Swapping. A lot of swapping.

Steve: Yes. The kludge was that the BIOS, actually in the BIOS, it supported a second virtual floppy drive. And so it always showed two drives. So DOS always had A and B. And if you did like a dir of B, which DOS said, well, yeah, okay, he's got two drives...

Leo: Give us B. Give us B. Where's the disk?

Steve: Yeah. So what would happen is the BIOS would prompt you on the screen saying please insert the floppy for Drive B. And then it would keep track of which floppy was in the one floppy drive you actually had. Well, believe it or not, I'm still fighting this today. I mean, because none of that has changed.

Leo: What? Why?

Steve: Well, because unless I intercept the BIOS's writing to the screen, please insert the floppy drive for B, if SpinRite touches B, that comes out.

Leo: Wow. Oh, wow.

Steve: So I had to intercept what's known as the multiplex interrupt, which is INT 2f. And if AH equals, I think it's 4,000, I have to return with the CX register set to all F's in order to suppress that message being written to the console. And I needed to touch A and B because I want to build a list of the valid drives to which SpinRite can log its results. So anyway, turns out there was something I got wrong. A couple people actually had machines with floppy drives, which is why I so much believe in testing this stuff, and it's all fixed and working now. So after today's podcast I'll be moving on to the next stage of work.

Leo: AS8003. How big - so you said Class C is the smallest. Is this a Class A? I mean, this has got to be a lot of addresses.

Steve: Bigger than a Class A, actually.

Leo: Bigger than that. Wow.

Steve: Yeah, now more than. So 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 - remember Border Gateway Protocol, I'll refresh our listeners in a second - about that arrived from a previously unknown entity advertising that they would henceforth be receiving traffic - they, the entity - for all 16,777,216 IPv4 addresses beginning with 11. So today...

Leo: The number 11?

Steve: The number 11. If it's 11-dot...

Leo: You never see those. You never see those 11-dots, do you.

Steve: There has never been anywhere for them to go. 11-dot has never been used.

Leo: Interesting. Don't forget that the Internet was invented by ARPA. I mean, it was done for the Pentagon.

Steve: Right, right.

Leo: So of course they're going to get whatever they want. A giant block.

Steve: Right, right. And once upon a time remember there was no competition.

Leo: No one else wanted them, yeah.

Steve: HP had 14-dot and 15-dot.

Leo: Amazing. Amazing.

Steve: Yeah. And in fact also remember that 5-dot had never been allocated, and that's why Hamachi was able to use 5-dot-anything as virtual IPs for all of the people within its peer-to-peer networking system.

Leo: It could be routed, but it wouldn't conflict with anything.

Steve: Right, exactly.

Leo: Wow. Wow.

Steve: So today, more than three months later, there's still much we don't know. But this weekend, as you referred to, Leo, we learned a bit more. So I'm getting a little bit ahead of myself. Here's how the Washington Post began their coverage of this mystery. They said: "While the world was distracted with President Donald Trump leaving office on January 20th, an obscure Florida company discreetly announced to the world's computer networks a startling development. It was now managing a huge unused swath of the Internet that for several decades had been owned by the U.S. military. What happened next," they wrote, "was still stranger.

"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% 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, Florida, outside of 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, the messaging system," they wrote, "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 should now accept traffic, but that it should be routed to Global Resource Systems."

Okay. So the stage is set for the mystery. Let's step back 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," thus the name of this podcast. And 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 - that's the AS - 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 the routers it's connected to using the BGP, Border Gateway Protocol. They update their own routing tables to incorporate this new information into their routing table, and then they in turn forward any changes that they made which resulted, as a consequence of incorporating this information into their tables, to the routers they are connected to.

The upshot of this is that a so-called "advertisement" or "announcement" quickly propagates throughout the Internet, adjusting all other routers as needed, so that any packet that's dropped onto the Internet anywhere will end up being routed to and eventually arrive at the router that maintains, that is responsible for that block of IP addresses which it is now announcing.

So as we've discussed several times before, a simple slip of the finger when updating those crucial tables, or deliberate shenanigans, can raise quite a ruckus across the Internet as large blocks of traffic are rerouted from their intended destination, if in fact those IPs were active already. And we've also talked about how, because the Internet was sort of assumed to be run by people who were responsible and careful and knew better, and after all it was all just a big experiment anyway that really wasn't known to succeed or not in the beginning, it was fine. The point is that we still have a lot of that same architecture, and nothing has changed since then. BGP is notoriously vulnerable and lacking in security.

Okay. So with all that in mind, here's what Doug Madory experienced and shared this weekend. He wrote: "On January 20th, 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 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" - which is 11:57 a.m. Eastern - "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.

"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 Autonomous System (AS) number in the IPv4 global routing table. By mid-April" - meaning this month - "AS8003 dramatically increased the amount of formerly unused DoD address space that it announced to 175 million unique IPs." Okay. So 175 million IPs is 1/25th of the Internet's entire 4.3 billion IPv4 space.

Then Doug continues: "Following the increase, AS8003 became far and away the largest Autonomous System in the history of the Internet. By comparison, AS8003 now announces 61 million more IP addresses than the now second largest Autonomous System in the world, China Telecom, and over 100 million more addresses than Comcast, the largest residential Internet provider in the U.S." And I've got a graph of largest ASes

by IPv4. GRS-DoD is in first place, then China Telecom, AT&T, the Department of Defense's active network, then Comcast, China Unicom, and so on down. So big, big, big.

He says: "In fact, as of April 20th" - so a week ago today - "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 this year." So he says: "As a valuable asset," he said, "decades ago, the U.S. Department of Defense was allocated numerous massive ranges of IPv4 space. After all, the Internet was conceived as a Defense Department project. Over the years, only a portion of that address space was ever utilized," he says, "in other words, 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 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" - that is to say, that's 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."

So he poses the question, what is AS8003 doing? "Last month, astute observers to the NANOG (N-A-N-O-G)" - that's the North American Network Operators' Group listserv, and that's basically sort of the inner sanctum where those who run the Internet talk to each other. He said: "Astute observers on the 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."

The DoD's statement said: "DDS (Defense Digital Services) authorized a pilot effort advertising DoD Internet Protocol space using Border Gateway Protocol. This pilot will assess, evaluate, and prevent unauthorized use of DoD IP 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."

So Doug said: "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 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 back in 2018."

He said: "For decades, Internet routing operated with a widespread assumption that ASes did not route these prefixes" - that is, the 1-dot prefixes - "on the Internet, perhaps because they were canonical examples from networking textbooks. According to their

blog post soon after the launch, Cloudflare received around 10Gb of unsolicited background traffic on those interfaces that were announcing those two Class C networks beginning with 1.1.1.* and 1.0.0.*. And that was just," he writes, "512 IPv4 addresses." He says: "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."

So he says, in 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 Autonomous System 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 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 enormous traffic being sent their way."

And I also have some reporting from the AP. It's sort of more gossipy in nature. Digging down, you can of course uncover all kinds of weird things. I'll share some of what the AP wrote. They said: "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 AP wrote: "The company did not return phone calls or emails from The Associated Press. It has no web presence, though it has a 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 attorney, 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 partner 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 (DARPA) 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 of 2020. It shares the same Virginia Beach, Virginia address a UPS store in corporate records as Packet Forensics. The two have different mailbox numbers at the same location. 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, who is now 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." And if the name Packet Forensics rings any bells for our listeners, that's why. We spent a lot of time covering the covert use of these so-called TLS interception middleboxes, and Packet Forensics was among the purveyors of those technologies that we talked about before.

Anyway, the AP coverage that I was quoting here finishes up, saying: "The company continues to sell 'lawful intercept' equipment, according to its website. One of its current contracts with DARPA is for 'harnessing autonomy for countering cyber-adversary systems.' A contract description says it's 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.'"

Anyway, deepening the mystery, they conclude, is Global Resource System's 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. That's where that reference to AS8003 came down. 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, for what it's worth, listening to this podcast knows as much as anyone else, aside from those who do know what's going on, and those who do are choosing not to say anything on the record.

Leo: It's nothing.

Steve: It is interesting that the - what, Leo?

Leo: It's silly. Okay, put yourself in - okay. You're in charge of AS8003. Some low-level functionary in the IT department, or I don't know who is responsible for those addresses. But somebody at some point says, you know, we really ought to make sure, you know, remember that 10 years ago somebody misused that 11-dot? We really want to make sure nobody's got a botnet on there posing as that or using it for DDoS. How do we do that? Well, we can do this. We can do a little quick BGP route and see what traffic we get, maybe run through it and see if we find anything. Okay. How would you do that? Well, I don't know. Look, we've got this shell company we always use for these kinds of things. We've got that address down at the UPS Store in Fort Lauderdale. We don't want to make a big deal about it.

Steve: Actually it's even closer to the NSA. It's the UPS Store in Virginia.

Leo: I guess it could be the NSA. But don't you think the DoD has people that do exactly this kind of thing? It's the Department of Defense. This is their block. I don't think it's anything nefarious. I think they just didn't want to make a big deal about it. They didn't want anybody to know what they were doing, especially the bad guys. So they just ran a little operation. What do you think?

Steve: My take is different.

Leo: Okay.

Steve: I think - I don't think it's nefarious. But I think it was in danger, they were in danger of losing it.

Leo: Oh. Because Congress might have made them sell it.

Steve: Yes. They might have thought, whew. We had a Congress that let us...

Leo: Keep it, yeah, yeah.

Steve: ...get that out of the Appropriations Bill that year. But a Biden Congress might not be so amenable.

Leo: Oh, maybe that's why, right, right.

Steve: And so if we're not using it, how do we defend our need for it? But, boy, do we want it.

Leo: Well, how does this defend their need for it?

Steve: They can just say, I mean, somebody said, "Look at all of that IP space that the DoD has, and obviously has no need for."

Leo: Right. But this doesn't show a need for it. This is just a security assay, basically.

Steve: Well, exactly. It is absolutely. I don't disagree with the brainstorming that has followed its allocation because you are indeed, you know, we've often talked about honeypots. You can make the honeypot of all time.

Leo: Oh, yeah. Oh, yeah.

Steve: On the other hand, the bad guys know which IP spaces are now being forwarded to AS8003. And Doug made a good point, too. You just have to black hole all of that bandwidth because when you announce that much unused space, you're probably going to just be buried with packets, which you really don't want because you actually don't have any use for them right now.

Leo: You know, Jonathan Bennett in Discord says the sensible thing. This was signed off by the White House, probably two years before they started this. And they didn't want to have to go through a whole process again with a new White House. So they just said, look, if we're going to do it, we've got to do it before there's a new Secretary of Defense and all this.

Steve: Makes sense.

Leo: So let's just get this over with. It's already been approved. The actual Secretary of Defense, Mark Esper, left a week or two before this. He was gone. So I bet you that it's something that simple. And the whole thing about the reporter going down to the shared workspace and saying, hey, is this company here, and the woman throwing her out, that's just silly filler from the Washington Post. It was just a security operation, and somebody got wind of it and made a big story out of it. I don't - do you see any - what's nefarious about this? This is what they should be doing. They probably should be doing it more often.

Steve: I think it's just fun.

Leo: Yeah. Oh, no, I agree. It's a great story.

Steve: Yeah. I think it's just fun. And what we don't know is which came first. Did the concern about losing it since they weren't using it, did that precede, hey, you know, we could be using this.

Leo: But they're not using it now, are they? I mean...

Steve: No.

Leo: No, they just ran a scan, basically, and downloaded terabytes of garbage data, and now they're done.

Steve: So what they did was they had unused space.

Leo: Right.

Steve: They have always had it Blocks of the Internet, the original IP space, were just allocated to DARPA.

Leo: Right. Never used it.

Steve: And someone said, hey, DARPA, you want 2, 3, 5 - or no, not 5 because we know that was never used. But what blocks do you want? And so someone said, well, you know, we'll take 11.

Leo: We'll take 11.

Steve: Yeah.

Leo: Goes to 11.

Steve: And they got a few other ones.

Leo: Honestly, why are they even...

Steve: And they just never got around to using - they just never got around to it.

Leo: Who cares? They're never going to use it anyway, now that v6 is here. They can have a trillion addresses if they want.

Steve: Right. And that's too low a price, by the way.

Leo: Hundred million? Yeah.

Steve: Oh, yeah. You could get a lot more money than that.

Leo: A hundred million to the Pentagon, I might add, is cigarette money. It's not...

Steve: Yes, yes.

Leo: Even Congress turns its nose up at 100 million.

Steve: That is not going to help the budget.

Leo: If you have a $92 billion budget...

Steve: That's not even a rounding error on the general fund.

Leo: Yeah, yeah. That pays for the hot dog stand in the Pentagon's [crosstalk] circle.

Steve: Anyway, it was a fun story that lets us talk about the way the world works.

Leo: I love the story. Well, I'm not knocking you at all, and I love it, yeah. I mean, it's fascinating. But I don't think there's - do you think there's anything nefarious at all about this? It doesn't sound like it.

Steve: No, no. And I agree with you that they had some random company somewhere, and they said, oh, just hook it up over there. We need to anchor these addresses somewhere. Let's send them over there.

Leo: Exactly. And the Pentagon said very clearly, no, no, we still own the addresses. That's not a real company. That's just a...

Steve: Exactly.

Leo: It's just a shell.

Steve: I just think that - I think they needed to light them up because they needed to be able to say, no, we're using those. Don't go selling out from under us.

Leo: It might be enough to convince Congress. You see? Something happened. I guess. Nobody's thinking about the 100 million, I think, at this point. It'd cost you more to administer an auction. And besides, who's buying IPv4 addresses these days? Anybody? Are they worth anything?

Steve: Oh, yeah, yeah, yeah, yeah.

Leo: They're still precious?

Steve: Yeah.

Leo: Comcast would want them. Steve, you're always fun. I love it. This show is a good conversation starter if you're really a geek, I guess. We do this show every Tuesday, 1:30 Pacific, thereabouts. That's 4:30 Eastern time, 20:30 UTC. If you want to watch live, we stream it live at TWiT.tv/live. There's live audio and video. You can chat with us live at irc.twit.tv. You can also get on-demand copies of every show at our website, TWiT.tv/sn. Steve's got his own copies. He's got the unique 16Kb version. If you don't have an IPv6 address, and you want to save some bandwidth, just get that 16Kb version. There's also handwritten transcripts, those are probably even smaller, by Elaine Farris. She does such a good job. You can get all that. Have you put up the current show yet? Because somebody's saying, oh, 2015's still up or something like that. I don't know.

Steve: Oh, did I forget?

Leo: They said: "He got busy with his new bride and probably forgot about us."

Steve: Whoops. Yeah, actually I probably was busy with SpinRite.

Leo: I know, that's exciting. Getting close.

Steve: I'm back in the groove.

Leo: We're getting close, yeah. GRC.com's the place to go. Actually, this is a good time to get SpinRite. 6.0 is the current version, but you'll get a free upgrade to 6.1. That's what's coming out now. And you could be an early beta tester. You could participate in the development of it. There's a great forum going on at his website, GRC.com. Lots of free stuff there to highly recommend it. SpinRite, the world's best hard drive, storage...

Steve: Mass storage.

Leo: Mass storage. I can't say hard drive anymore. The world's best mass storage maintenance and recovery utility, works on all kinds of mass storage.