A Crazy News Week!

Description: So much happened in the security and privacy worlds this past week that it will be everything Father Robert and I can do just to cover and discuss it all during a single podcast. So this is one of our pure news coverage and catch-up episodes. I'm sure it's going to be a blast!

High quality (64 kbps) mp3 audio file URL: http://media.GRC.com/sn/SN-515.mp3
Quarter size (16 kbps) mp3 audio file URL: http://media.GRC.com/sn/sn-515-lq.mp3

SHOW TEASE: It's Security Now!. IPv4 is done - again. We got WikiLeaked, and Steve actually feels sorry for the NSA. Microsoft gets friendly with your WiFi password, and hackers get hacked. It's all coming up next on Security Now!.

FATHER ROBERT BALLECER: This is Security Now! with Steve Gibson, Episode 515, recorded July 7th, 2015: A Crazy News Week.

It's time for Security Now!, the show that covers your privacy and security concerns online. I'm here with the one, the only, the man whose packets never arrive out of order, Steve Gibson from GRC.com. Steve, such a pleasure.

Steve Gibson: Hey, Padre. This is our second of three podcasts we get to do together, and I'm delighted.

PADRE: Absolutely. Well, Leo is still in the Tardis, I believe, traveling through Europe or something. So he will be back, I think, in is it eight days? Nine days. Something like that. But while he's gone, Steve Gibson and I can play. And Steve, I've got to tell you, this week has been crazy. Originally this was supposed to be a question-and-answer episode. But I haven't had this much high-impact security news in months.

Steve: Well, yeah. As you know, we do allow ourselves the freedom to just do news when that's all we have. And you and I tend to go deeper into these things, I mean, if last week was any example. You know, we were - we broke a record last week in the length of the podcast. And so as I was putting this together, looking at all the news that we had to talk about, I thought, okay. There's just no way we're going to have any chance to - and in fact, even as it is, I sorted these in the order of some we can kind of dispense with quickly, a bunch of meaty stuff in the middle, and then other stuff, if we don't even get to it, it'll be fine. So, yeah, we're - and I didn't even try to get any mention of SpinRite in. I've got two great testimonials from users, but I thought, oh, I'll just push those. Everybody knows about SpinRite, so - or, you know. So we'll do that one

when we can.

PADRE: I don't know where you're going to cut out space because, I mean, I'm looking at all the stories, and they're all meaty. I mean, we could spend 30 minutes on each one of these, and it wouldn't be done.

Steve: As I promised, I've got news on the details of v39 of Firefox. Not a lot to talk about, but I will just cover it briefly. There's the question of ICANN reconsidering the WHOIS privacy policy, which has generated a lot of controversy because they're considering not allowing WHOIS privacy for some class of domains, and we'll talk about that. There's a new DDoS attack protocol being found in use in the wild. Amazon has stepped into the game with their own TLS protocol stack, to the surprise of many. ARIN, on I think it was Wednesday night, ran out of IPv4 space, had to deny a request for the first time ever, and switched policies. We've of course been tracking this for a long time, so we'll talk about that.

Then everyone wants to talk about Italy's Hacking Team which got hacked. We'll cover that. Lots of stuff about the NSA's XKEYSCORE program was published, I think by WikiLeaks. And also news of the NSA's international spying has come to light. And then Windows 10 is worrying people over something called WiFi Sense facility, which is sort of borrowing from Windows 8.1, but is apparently making it a little bit more pervasive. And even more. So, yes, tons to talk about this week.

PADRE: Oh, good. So there's not much. We're really kind of dry on topics this week.

Steve: Yeah, we'll have to stretch it out. We'll just hem and haw. Now, I just - your mentioning Carbonite put me in mind of a tweet that I received, and I had it in the show notes because I thought it was kind of clever. It's at the very end of the show notes, so we'll do it out of order just because it ties in with Carbonite. I got this tweet from Chris Wronski, whose Twitter handle I kind of got a kick of, it's @theemptyset. So anyway, he...

PADRE: Nice.

Steve: He did an @SGgrc and @leolaporte. And he wrote: "I just solved the settlement-free peering problem. Netflix merges with Carbonite. You're welcome." And of course very geeky. What he's talking about is that, you know, we were explaining last week how the problem with peering is that people who have settlement-free peering want to push as much bandwidth as they pull. They want to give as much as they get, the idea being that bandwidth that's coming into them is using their network, so they want bandwidth to also leave them in order to use their peering partner's network.

The problem with Netflix is that it's entirely one-sided. It's one direction. It's all of Netflix customers sucking bandwidth from Netflix. So anyone Netflix peers with sees all of that bandwidth. And as we know, during peak Netflix viewing usage at night, Netflix traffic is the majority of the bandwidth on the Internet, which is just mindboggling in itself. But anyway, the point is that Carbonite is a service that inherently runs in the other direction. It's all of the data on your hard drives going in the other direction toward Carbonite in order to have it - in order to keep it backed up. So I just thought that was cool and clever, a hyper geeky observation from Chris. And so thanks, Chris, for sharing that. And a chuckle for us all.

PADRE: Yeah, and there's another way to handle that, that peering problem, the asynchronous peering. And that's what Google did. So Google used to be like Netflix.

Google had their big datacenters, and they were a big strain on Tier 1 ISPs, and Tier 1 ISPs were complaining. So Google built out their physical plant. They've got - they've actually - depending on who you believe, because most of these companies are very, very secretive about how much fiber they actually have in the ground.

Steve: Yup.

PADRE: Google may actually be the number one owner of fiber in the United States. Of course they're not disclosing. That's why they can do things like Google Fiber, their own ISP. And that's why they have relatively inexpensive transfer between their datacenters. And what they now have on the ISPs is they could open up their networks and say, look, we can be a Tier 1, too. If you really want that, go ahead and charge us the higher price, and we'll open up our networks, and we'll become our own ISP. Netflix actually has the momentum right now where they could do the same thing. Yeah, they're delivering a lot of traffic. But if they start buying up unused fiber, dark fiber, they could have another ultra geeky solution - other than becoming Carbonite.

Steve: Well, yeah. And of course the solution is - and I guess, you know, again, as you said, a lot of this is sort of murky. But caching inside the ISP makes sense. I mean, how dumb is it for Netflix to have a copy of, like, you know, "House of Cards" series or "Sense8," for example, and be individually sending repetitive copies of the same stuff to all the customers, for example, at Level 3. Makes so much more sense for Netflix to arrange to have a cache inside of Level 3 so that it's being sourced locally. Anyway, we've talked about that stuff in the past, and it's, you know, ultimately it'll all get resolved one way or the other.

PADRE: It's a fun topic, but we're not going to solve it in two hours.

Steve: Yeah. So Firefox v39. Not much new. They did add their so-called "safe browsing malware detection" to both the Mac OS X and Linux. And I put in the show notes "Oh, joy" because this is based on Google's Safe Browsing info. And about two months ago or so the TrueCrypt files that I have been hosting ever since the day that TrueCrypt announced they were going to no longer support TrueCrypt, those files got blacklisted for no reason anyone knows by Google, and thus by Firefox. So the fact that they're now extending, I mean, it's good for Mac OS X and Linux users who are Firefox users. But it's like, yeah, these things do sometimes false positive, as we know. And they've added malware detection for downloads that cover the Mac file types, as well. They added URL-sharing on their Hello, which is the built-in real-time chat component of Firefox.

Um, one problem they were having was that plugins, Firefox plugins that initialized slowly could hang the startup process of Firefox. And so in a nice change they've made that asynchronous now, so that they launch the initialization, but it's not the initialization thread. They essentially spawn another process to initialize the plugins to allow Firefox to come up and for the plugins to initialize as they're able to. They formally removed support for SSLv3 from network communications. And I've noticed my own jargon usage. I'm now comfortable saying TLS rather than, like, TLS/SSL, trying to pay homage to the fact that it's still sort of around. I mean, it really is time now for us to move to TLS and say goodbye to SSL.

They also disabled use of the RC4 ciphers, the various cipher suites using RC4 encryption, except you are able to temporarily whitelist specific hosts that you can only reach that way if necessary. But, you know, they're just sort of doing that as a soft goodbye. I don't think anyone will ever encounter that. They fixed 13 security-related bugs and then continued offering support, they added support for rather obscure HTML5 features.

And I did get sort of a related tweet from an old friend of the podcast, Alex, I hope I'm pronouncing his name, Neihaus, maybe it's Neihaus, has been a friend of the podcast forever. He was the VP Marketing at Astaro Corp., and Astaro was this podcast's first sponsor, a great bunch of UNIX guys who produced the Astaro Security Gateway that we've talked about and many of our users are using. Or our listeners are using. Anyway, Alex tweeted: "After years of avoiding it, switching to Firefox plus NoScript from Chrome. It hurts so good. Not recommended for the faint of heart." So Alex, great to have you over here.

You know, of course, we were talking last week about the pair of Firefox and NoScript and how, with Firefox being more pro privacy, and Chrome and Google seem to be going in a different direction, going in the Google direction, I think we're seeing a little more of a separation of intent between those browsers. And I'm really happy with Firefox.

PADRE: It causes a little bit of startup annoyance.

Steve: Oh, yeah.

PADRE: The first time you do it, yeah, you're going to have to put in your rules. You've going to have to approve every site that you actually want to allow through. But the end result is so much better. I'm with you. After last week's episode I actually started using Firefox probably 50-50 with Chrome. And I'm starting to be okay with switching over. I really got addicted to Chrome. I got addicted to all the integration. But you're right, the more I think about it, the more Chrome has now become the IE. It's bloated. It's not super secure, not the way that I think it is. And I will put up with a little bit of pain if it gives me a more secure surfing environment.

Oh, Steve, I do want to ask you about this. I understand why they went with the asynchronous plugin initialization. That always makes sense because you don't want people to feel like the browser is slow. And if it takes the browser 45 seconds to load up, they're just not going to use it. But at the same time, that now means that users can start using the browser. They can start making connections before all the plugins are initialized. And if some of your plugins are security plugins, that's - I'm not sure if I'm okay with that. I would like at least an indication of whether or not my plugins are actually active at the moment.

Steve: Yeah. And I didn't look at it enough to see whether, for example, they're waiting in order to - like before they make page queries, whether they're waiting for the plugins to stabilize. It's not, you know, I would be surprised if they did something that was insecure. But you're right, that is an issue when we have something like, for example, NoScript, that wants to be very proactive about protecting us. Or like uBlock, for example. I have another mention of that later in the show notes because someone came up with a power tip for that, that I liked a lot.

I did want to take one moment to mention something that came up in the context of my tracking down Alex because I was trying to - I knew him, of course, but I was trying to remember whether it was the fact that he was VP Marketing at Astaro. So I used a tool that I've recommended in the past, that I think I only mentioned it once. And it's called MailStore Home 8, which is free for noncommercial use. It is an amazing email archiving tool. And so what happened was, when I - I think it was that my Eudora, you know, all my past mail just got to be too much to keep. And so I fired up MailStore Home 8 and let it have it all. It's all indexed. It's instantly searchable. I had a lot of positive feedback from people who listened to my previous recommendation of it.

And so I just wanted to remind people it's there, and it is still cranking away for me. And all the people that switched to it have found it to be just indispensible. Basically, it allows you to get all the email out of your client into this indexed, like, very fast, perfect solution for basically archiving all of the email that you have locally, for people who just don't want to leave it in, for example, Google's clutches forever.

PADRE: Or you can just do it my way. I'm still running Office 2003, I think. Actually, I was on Office 97 for the longest time. And all my email are just in PSTs in my NAS store. Don't do it that way. But, I mean, I've gone too far, Steve. I can't turn back.

Steve: Well, I'm still using Eudora, if we want to out old-stuff each other. And, you know, Eudora runs on 98. And it's sort of limping along. It's like, you know, I just - it works for me. It's like, why change? Ultimately, I probably will at some point. I mean, probably it's 16-bit code, and it won't run over on - actually, I think there is a question about whether it runs under Windows 7. I know that some people who are Eudora fanatics, as I am, have managed to get it running under Windows 7. But for what it's worth, this MailStore Home 8, you might want to take a look at it because it can suck in your PSTs and give you a single indexed archive where you just type in a few letters and, bang, here's all of the email that contains that string. It's really a nice piece of work.

PADRE: Nice. Actually, I think I have my Eudora, my last Eudora installation, I had Eudora Pro, is on a ZIP disk. I just need to find a drive, and I'll be able to pull it back off. Maybe, if I don't get the click of death.

Steve: Well, and if you do, we have a solution for that, too.

PADRE: Yes, we do.

Steve: Okay. So ICANN. When I saw all this controversy about ICANN, I thought, okay, what's going on? Even Google's Adam Langley did one of his infrequent blog postings to talk about this. And I'll wrap this segment or this discussion of ICANN up with what Adam found. But I thought, okay. Let's find out what's going on.

So the first thing I hit was a 98-page PDF of bureaucratic doublespeak. And I just, as I'm looking at this, I'm thinking, I salute the people who somehow have the fortitude to, like, deal with this because I recognize in something as big as the Internet you're going to have to have committees and working groups, and it's going to be political, and everybody's going to have, like, what they want. And so I'm just dispositionally unable to participate in that kind of process. It would just drive me crazy. But, you know, so I salute ICANN people who somehow manage to survive this, maybe even thrive in this environment.

So what this is, is this 98-page bureaucratic doublespeak is the initial report on what they call the "Privacy & Proxy Services Accreditation Issues Policy Development Process." This is the "Initial Report on the Privacy & Proxy Services Accreditation Issues Policy Development Process." Now, okay. So WHOIS, as those old-timers among us know, there is this database, the WHOIS database, which it's possible to query to find out who registered a domain name. That's what this is all about is the - so the idea is, there has to be a registrant for every domain name. But you can have a proxy service that provides privacy for people who don't want, for whatever reason, for privacy reasons, to have a public registration. They don't want, you know, their name and address and phone number and email to show.

GRC doesn't take advantage of that. If you look up the WHOIS registration for GRC, you

will find our business address and contact information and so forth. But I understand that just sort of because people have a right to privacy they may not want that. So what's controversial is, and what stirred up everybody, is that ICANN has been considering limiting the privacy available for something called "commercial organizations." And then, like, okay, wait a minute. Commercial organizations? That sounds like, you know, GRC, for example, is a commercial organization. I don't - I've never needed the privacy. And actually I'm annoyed at the idea that I have to pay for it every - I'm still using Network Solutions, and they want $10 a year in order to mask my ID. And the idea of that, I mean, I would do it if it was like a one-time fee. But the idea of paying them $10 a year just annoys me so much, it's like, no, I'm not paying you $10 a year.

Anyway, so here's the story. Digging into this, because I wanted to try to figure out what it was, Section 1.3 is the Working Group's Preliminary Recommendations. And so under 1.3.1 is the summary of the working group's agreed preliminary conclusions. And under Section 2 of that Section 1.3.1, in all caps, it says, "NO DISTINCTION IN TREATMENT; WHOIS LABELING REQUIREMENTS; VALIDATION & VERIFICATION OF CUSTOMER DATA." And so under points 2 and 3, which are the only salient ones, it says, first of all, it defines privacy and proxy services as P/P services are to be treated the same way for the purpose of the accreditation process.

And then it said: "The status of a registrant as a commercial organization, noncommercial organization, or individual should not be the driving factor in whether [this] P/P" - that's the privacy and proxy - "services are available to the registrant. Fundamentally, P/P services should remain available to registrants, irrespective of their status as commercial or noncommercial organizations or as individuals. Further, P/P registration should not be limited to private individuals who use their domains for noncommercial purposes." Then there's a reference to Note 10.

So Note 10 says: "Note that while the working group agreed that there is no reason to distinguish between commercial and noncommercial registrants simply because of their organization's entity status, it has not reached consensus as to whether the use of proxy and privacy services for certain types of commercial activity associated with a domain name should be barred." So basically they have a preliminary - they've agreed to a preliminary conclusion that this should not be changed, that there should be no distinction being made in whether someone can have a private registration or have their registration be private one way or the other. But even though they have agreement, they do not have consensus within the working group.

PADRE: This sounds like this is chisel instead of hammer. This is lawyer speak. I've seen this before. So essentially what they're saying is the first plan, which was to ban anonymity because of some bad actors out there, isn't a good idea. And that's what that whole Section 1.133 and 2...

Steve: Slash slash 2.

PADRE: Slash slash 5.

Steve: Right, right.

PADRE: Alpha acorn says. Which is, look, we understand there are bad actors out there, but we can't remove anonymity for everybody just because there are some bad actors. But then that Note 10 is essentially saying, but we still say in special cases we should be able to take action because there's a bad actor. Which is a very interesting way to go about it.

Steve: Well, and so what they ended up doing, because they could not reach a consensus within the working group, is to open it for a 60-day public comment period, which ended, or ends, today. So for the last two months they've been open to receiving comments. And because he was curious, Google's Adam Langley wrote some scripts to automate the process of going through the 10,000 email submissions which have been made. And he basically came up with sort of a soft determination that about 90 percent of those public comments supported making no change, that is, argued against the idea that there should be a change made such that some entities would not be able to be anonymous, which was what this - which is essentially concurring with the nonconsensual preliminary agreement that was reached.

So looks like everything stays the way it was. And we should note, as you have, that essentially it's largely sites that are perpetrating fraud, that have copyright violations, where it's more work to find out who's behind the domain which is perpetrating some sort of conduct which people like law enforcement believes need to receive a letter to tell them to cease and desist. And of course they've got a known IP that their domain name resolves to. So if you can't find them, then you then send a legal action to the provider of that IP space and then say, look, we need to find out who's behind this IP because they're bad people, and so turn over the information and so forth. So it's not like you really get bulletproof security from this.

And, I mean, over the years I've often made use of WHOIS when I've needed, for whatever reason, to get a hold of somebody. It's convenient. If you want to say hey, well, for example, you've got a domain name. I'd like to buy it from you. I did buy one once. I bought - it was for the encrypted VPN tool, CryptoLink. I bought CryptoLink.com. He had it, and he wasn't using it for anything. I said, hey, interested in selling? And I paid him $5,000 for it. And so that was nice that I was able to easily get a hold of him because he also did not have a private WHOIS registration.

PADRE: I need to put you in touch with Karl Auerbach. He's a friend of mine. He's on This Week in Enterprise Tech quite a bit, actually. He is a former ICANN board member. He's the one who actually sued ICANN because he wanted their finances to be public, because he wanted everyone to understand where the money was coming from and where it was going to. So he, yeah, he is the last at-large board member. After him they decided it's too much of a pain in the butt. But he can tell you exactly what kind of lawyer speak they have at these meetings because he rails against it.

Steve: Oh. Oh. And I just, again, it's just pure bureaucratic, like just shoot me now. I still, oh my lord, I can't imagine surviving that. But I recognize somebody has to do it.

PADRE: Right.

Steve: So I'm glad there are people who manage to work through it. Wow.

PADRE: It's not going to be me. Unh-unh. Yeah.

Steve: So - go ahead.

PADRE: No, I'm sorry, I had to have a cleansing breath to get ICANN out of my head.

Steve: Yes, exactly. Well, now we'll switch to technology, which is safe and fun. So we know that historically there have been different ways of perpetrating denial of service attacks. And one of the more in-fad approaches has been to use the UDP protocol. UDP differs from TCP in that it is generally non-authenticated. That is, inherently, you make a

request by sending typically one UDP packet carrying the request to a UDP server, which is listening for them. And then it says, oh, yeah, I have what you need. And it sends back whatever and however many packets are necessary in order to answer your query, to respond to that request. And it sends it back to the IP address that was the source IP in the packet it received.

Now, that differs from TCP. TCP, oh, and I should mention that UDP is consequently considered a connectionless protocol, whereas TCP is a connection-oriented protocol. In TCP, as we've discussed, there's a so-called SYN packet which first goes to the server. And SYN is short for synchronize. That provides the TCP protocol at the server end with some important numbering information for the bytes that will then be sent. The server sends back its own SYN, along with an ACK in the same packet - thus it's called a SYN ACK packet - back to the initiating client. And the client then sends an acknowledgment of the receipt of the SYN portion of the server's packet back to the server.

This has several effects. It allows the ends to synchronize with each other with this synchronization information. But notice that it also verifies the possibility of roundtrips because there had to be a roundtrip from the client to the server, and also from the server to the client and back to the server. So this sort of establishes both endpoints and allows them to set up their communications. What that means is, though, you cannot spoof TCP/IP, which is something that some people don't understand. You could do a SYN flood, where you spoof the source IP of just SYN packets, spraying those, you know, at someone from many different source IPs, or spoofed source IPs. But you can't actually initiate a connection because that requires successful roundtrips confirming the receipt of information. UDP, though, can be spoofed.

So there have been, and we've covered them, and I'm sure you have, Padre, on your podcasts, DNS has been an often-spoofed protocol that is used in denial of service attacks. And it's an amplification attack, which is what people want. They want to be able to send one packet to a server with a spoofed source IP and have the reply that the server generates be much larger than the packet that asked the question, the reason being that - that's called a bandwidth amplification attack. And then that server, if you spoof the source IP, that is, if you change the IP of where you say the client is, then the server responds to the source IP, which is the target that you are trying to flood with excess traffic.

So DNS has been a target of those attacks in the past. But what happens is, since DNS servers tend to be managed by watchful IT personnel, after the DNS servers realize that they need to lock themselves down, so for example they will only respond to DNS queries from their own clients, not broadly to the whole Internet, then they're able to filter their traffic. And so DNS stops being useful, as it initially was, as a bandwidth to use for attack. The same thing is true of the more recent abuse of network time protocol. NTP servers have been recently used for exactly the same sort of bandwidth amplification reflection attacks using spoofed source IPs.

Anyway, Akamai recently reported that they've been seeing attacks from a heretofore not used UDP protocol, and that's RIP. The Routing Information Protocol v1 is ancient. It's, I mean, it dates back to 1988. It was published in RFC 1058, 27 years ago. I mean, it's virtually useless because, for example, among other things, it is a classful routing protocol. That is, it can only obey the original class A, B, and C designations, where you either have the high bit of the 32-bit, the 4 bytes of IP space specifies the network, and then the other 24 bits are the hosts on the network. That's a Class A network. A Class B divides the 32-bit IP space in half so you have a 16-bit network number and 16 bits' worth of hosts. Or of course a Class C is what users have in their homes who have, for example, 192.168.0.x in a Class C.

But of course it turns out that, as the Internet grew, those original class designations became way too restrictive. And so what was evolved was something called CIDR, C-I-D-R, Classless Inter-Domain Routing, where we could flexibly set the division of, like, the dividing point within the 32-bit space anywhere we wanted to. Well, RIPv1 can't do that. So it's essentially not used anymore, and it was replaced by RIPv2 back in 1998. So it was worked on in the mid-'90s, standardized in 1998, so still a long time ago.

Nevertheless, it turns out that a large number - and here's the problem - of SOHO, you know, Small Office/Home Office routers, the typical little plastic blue boxes, it turns out that a lot of them are supporting RIPv1. It's a UDP protocol that listens on port 520. And so some nefarious individuals have scanned the Internet, sending out probes to port 520 for RIPv1 and recording all the IP addresses that responded. And it turns out there's lots of them. So Akamai has reported denial of service floods of, like, 12.8 gbps from attackers that have been using 500 of these SOHO routers that are still configured with RIPv1. So what the bad guys are doing is, just as with the other attacks, they are spoofing the source IP to that of the target. And they're sending UDP packets.

And unfortunately you can send a tiny query to a router that supports RIPv1, and basically what it's saying is "send me your routing table." And so that can be a, well, is always going to be a multi-entry larger response, which then is sent to the victim, I mean, and it's enough bandwidth that in aggregate it's greater than 12 gigabits of total bandwidth. And the problem is these are routers unlike the DNS routers and NTP that normally have IT personnel that are keeping an eye on these, these are never going to get fixed. There's no way that most of these are going to get fixed. So here we have a new network amplification attack that is using routers that are not going to see the RIP protocol removed in the foreseeable future.

PADRE: You know, that's what makes this attack so scary for me because DNS and NTP amplification attacks, they made a lot of splash because they were big. I believe the last big NTP amplification attack, the attacker used a 1 megabit line and was able to generate 200 gigabits of traffic. It's that asynchronous. But as you said, DNS servers get patched, and everyone's moving over to DNSSEC, which is immune to amplification attack - oh, right now is immune to amplification attacks. NTP servers, I heard of two ways that they were dealing with it. One was they were patching it, and then they were also upstream blocking the NTP servers that have been abandoned, basically they're just sitting out there on the Internet. And that has basically gotten rid of that.

There is no way to patch or block all the SOHO routers on the Internet that have RIPv1 enabled. They're just - there's too many of them, and you can just keep - it's very easy to scan for them. In fact, I read this article, and I started using one of my tools. I found three dozen unpatched routers in my network segment that were running v1.

Steve: Right.

PADRE: And I'm thinking, that's too easy. And I have no idea who these routers belong to. I can't contact them and tell them could you patch your router, or please turn off that port. That's a problem that's just going to sit out there until this hardware dies and is replaced.

Steve: Yeah, I guess the only thing that might happen if this really became a problem, I mean, my sense is it's also, like, maybe it doesn't reach the threshold of being a big enough problem. But, for example, I've switched - I was talking to you before the podcast. The podcast listeners don't know that after 19 years of having two T1s connecting me to the Internet here in my home office, that I've spoken of often with Leo, they went dark on July 1st after 90 days - I had not received a notice, just because the

contact information that my bandwidth provider had had expired a decade or more before. I'm now using standard cable modem bandwidth with my provider Cox, and it's a filtered connection.

So, for example, I can't - I don't need to run a port 25 SMTP server here, but I can't. And I don't need to expose and don't want to expose Windows networking, but Cox is blocking 137, 138, 139, and famously 445, which are the Windows printer and filesharing ports. So what Cox could do is proactively block port 520, which would protect the routers of their subscribers, in the same way that they're blocking Windows networking to protect - I don't know who has those ports open to the Internet anymore, but the ISP has been proactive. And so it could happen. But my guess is it would be hard, in this day and age, for that to sort of rise to the level of it actually occurring.

PADRE: Yeah, they could do that for the consumer service. So if you're buying consumer service, those ports should be blocked because in the terms of service you shouldn't be running a server anyway. It's not on your local connection. But especially in my area, Comcast is really ramping up business services. In fact, they're ramping up business services to people who would traditionally be considered consumers because they say, look, if you get the business services, we'll give you an IP, a dedicated IP. We'll take off all the caps. You pay a little bit more, you could also get phone service with it. Well, in business service, none of that is blocked. That's the idea of a business service.

Steve: Right, right.

PADRE: So I guess it's our job to go around and just start breaking these things.

Steve: Yeah. It's cool that you did a little scan to see what was going on in your own neighborhood. That is, as you may know, my own history of ShieldsUP! was when I first - when GRC was first getting on the network, and we had a Novell 10Base - or, no.

PADRE: 10Base2?

Steve: 10Base2, yes. Was it called 10Base2? I guess it was. Anyway, it was coax connection...

PADRE: Oh, yes.

Steve: ...running, you know, terminated on each end running in a big loop through the office. And we had, you know, coax T connectors hooking to our thousand-dollar LAN adapters, you know, back in the day. I did the same thing. I realized, wait a minute, I'm on a network now, I'm on the Internet, and unless we do something, our ports are going to be exposed. And so I did a little scan of my own local Internet neighborhood and found people's C drives. I mean, there it just was, C:. It was like, yikes. And so of course ShieldsUP! was my response. I realized that I could help people realize that they had this exposure that they weren't aware of otherwise. So that's how it happened.

PADRE: I think it was 2001 when I discovered Nmap, and I started mapping out wherever - because I moved a lot. That's a part of my job. And wherever I was, I would map out my local segment. And like you, I was just amazed at how much stuff was wide open. I found a couple of web cameras. I found many, many shared hard drives. I found printers. And I always wanted to send out a notice: By the way, this is open to the Internet.

Steve: Oh, by the way.

PADRE: You've got to close this.

Steve: So Amazon surprised everybody, really nice surprise. I'm just going to - I've paraphrased from their announcement blog posting because it contains some surprises which our listeners will appreciate. And I love the name of this, which I'll explain in a second. It's called s2n. And so what they said was, they said: "At Amazon Web Services, strong encryption is one of our standard features, and an integral aspect of that is the TLS" - and they said "previously called SSL," but none of us are going to do that anymore because it actually has died - "encryption protocol. TLS is used with every AWS API and is also available directly to customers of many AWS services.

"Part of the challenge is that the TLS protocol, including all of its operational extensions" - I'm sorry - "all of its optional extensions, has become very complex. OpenSSL, the de facto reference implementation, contains more than 500,000 lines of code, with at least 70,000 of those involved in processing TLS. Naturally, with each line of code there's a risk of error. But this large size also presents challenges for code audits, security reviews, performance, and efficiency.

"In order to simplify our TLS implementation, and as part of our support for strong encryption for everyone, we are pleased to announce availability of a new Open Source implementation of the TLS protocol: s2n." Okay. I'm still not going to tell anyone what that means yet because it's just too wonderful. s2n is a library that has been designed to be small, fast, with simplicity as a priority. s2n avoids implementing rarely used options and extensions" - which I think is brilliant, by the way - "and today is just more than 6,000 lines of code." Okay? Down from more than half a million in total in OpenSSL, and 70,000 involved in OpenSSL's handling of TLS, Amazon's s2n implementation is just a little over 6,000 lines of code, and it's on GitHub.

Continuing, Amazon says: "As a result of this, we've found that it is easier to review s2n. We've already completed three external security evaluations and penetration tests on s2n, a practice we will be continuing. Over the coming months we will begin integrating s2n into several AWS services. TLS is a standardized protocol, and s2n already implements all the functionality we use." I'm going to say that again. "s2n already implements all the functionality Amazon Web Services uses. So this won't require any changes in your own applications," Amazon writes, "and everything will remain interoperable. If you are interested in using or contributing to s2n, the source code, documentation, commits and enhancements are all publically available under the terms of the Apache Software License 2.0 from the s2n GitHub repository." Okay, s2n?

PADRE: I like that, the idea of simplifying and taking off all the options that are going to be security holes. I like that. But Steve, I'm wondering, because this is open source, and because you're free to contribute to it, how long before we start seeing feature creep? Because there are going to be those who say, well, this is nice, but I wish it had X; I wish it had Y. And then you go from that 6,000 lines of code closer to the half million that you have with TLS.

Steve: Well, okay. So first of all, signal to noise is what this stands for, and I just love that. The idea, you know, essentially they've reimplemented TLS with an implementation that has extremely high signal-to-noise ratio because what they've done is just what's necessary. Now, OpenSSL is the - it's the armature of TLS. Any new feature, I mean, everything that is experimented with for the TLS protocol is implemented first in OpenSSL. There are other TLS stacks of various heritage. I just love the idea of starting over. I mean, it's always a good thing to do. For example, it's what I did a week ago with NoScript in Firefox. I zeroed my whitelist, and I started over.

And of course that's why setting up a new machine is a good thing every decade or so. And I dread doing it. But eventually I will because, you know, systems just acquire barnacles over time. And OpenSSL is no different. There's no question that, even if you were to reimplement everything, you could do a far better job than OpenSSL is today. And in fact we know that. There have been efforts to just say, okay, we're taking OpenSSL, and we're just going to go through it and hack out the debris. Because, I mean, there's just, you know, there's abandoned things. There's, like, stuff no one needs anymore, that no one's using. But because it is, you know, it's the grandfather of SSL, it's all there. And it's just by virtue of the fact that it's there, it stays there. So starting again makes sense.

And the other nice thing about s2n is that you know you're getting a stack that works if it is the stack Amazon is using. Although they don't use it on their Amazon.com nearly as much as we wish, it is backing all of, or will be, backing all of their various services as they continue to roll this out. So I understand it's inherently the kind of thing that gets crusty over time, exactly as you say, Padre. But the idea of starting over, I just - I salute them. It's, boy, a lot of work, but yay.

PADRE: Great effort. I'm wondering, though, if anyone has put something up to tell us what were all the things they took out. I'd love to see a comparison, just to take a look, as you said, at some of those features that have long been dead, that no one has ever used, but just hung around because it's been in the old implementation. Now, this is going to be easy for AWS because, as they said, the only features they kept in are the features that are supported by AWS.

Steve: That they need. Right.

PADRE: But, I mean, assuming that you're going to have some customers who are pushing out services to, say, Azure or to Google services, I think that's when you're going to start to see it grow from its base 6,000 lines. But as you said, great experiment, great effort. I'd like to see them do that with other services and protocols that we have available to us right now.

Steve: Yeah, I mean, we talk about this all the time on this show. It is so difficult to leave behind something that works. And in fact, that is a perfect lead-in to our next story here, which is ARIN's first time ever initiation of what they call their "Unmet Requests" policy. This occurred on July 1st, at the same time my T1s were being disconnected. ARIN ran out - and it's not technically, but we'll cover that in a second. But the story was "ran out of IPv4 space and had to decline a request." So ARIN activated what they call their IPv4 Unmet Requests policy with the approval of an address request that was larger than the available inventory in the regional IPv4 free pool.

So essentially - so I was sort of curious about, okay, what exactly does this mean? So they have - what they have is a waiting list. And so when they say that they approved the request, what they did was they approved it for the waiting list. So somebody who now requests a block of IP, a contiguous block of IPs larger than they have available, is handled as follows. So ARIN says: "When ARIN receives a justified request" - now, that's the other thing, too. I've talked about in the past how when I set up my bandwidth at Level 3, they said, you know, how many IPs do you want?

And I said - and I understood even then they were like jealous of them. And I had 64 on my two T1s here, which I realize - which was, you know, waste because I was only using a few. But, you know, at GRC I've got a lot going on. So multiple true services. I've got all kinds of crazy other things happening. So I said, uh, can I have 16? And they said

sure. You need to justify what you're going to be doing with each of those. So they gave me an IP justification form to fill out to explain why I needed that IP space. And there was some room to grow.

So basically I said, okay, and came up with a lot of need all of a sudden. But I'm glad I have those because every so often I have to get a little clever now with how to fold things in. But so justifying IPv4 requests is a thing now. You really need to explain why you need this. So ARIN says they activated - oh. They said: "When ARIN receives a justified request for IPv4 address space that cannot be filled by a single block from ARIN's available IPv4 free pool, the requestor will have three options: Option No. 1, accept the largest available block in the ARIN IPv4 free pool that is equal to or less than your approved size. ARIN will fill the request per ARIN policy, and the requestor will then be ineligible to receive IPv4 addresses from the ARIN IPv4 free pool for the next three months."

So you get a block, and you've got to wait 90 days in order to ask for another one. And so if you ask for more than they've got, they'll say, well, you can have one of these. It's not as much as you wanted, but it's the biggest we've got. And so you can either decide to or not. If you elect not to accept an available block from the ARIN IPv4 free pool, and request to be put on the waiting list" - you can request to be put on the waiting list for unmet requests. "ARIN will ask you to specify the smallest block size you are willing to accept and will place your request on the waiting list for a range that includes your approved size through the minimum size you designated. This procedure is in accordance with" - and then they've got a Policy 4.1.8. Or choice number three: Elect not to accept an available block from the free pool and close out your request. Now...

PADRE: That's the "take my ball and go home" policy.

Steve: Yeah, it's like, okay. Now, what's interesting is, so I dug a little deeper. Last night, when I was putting this portion of the notes together, they had - this is the total amount of available space remaining. They had 46 remaining /23s. Now, okay, that's - remember that this is the so-called CIDR, the classless inter-domain router. So a /24 means 24 bits of network number, leaving 8 bits for host, meaning 256. But a /23 gives you 9 bits of host numbering. So that is to say, they had 46 available network blocks of 512 IPs per block, and 431 remaining /24s. Which, I mean, that's like none.

PADRE: It is nothing.

Steve: Okay. I happened to check again this morning. Okay. So just over the course of maybe 10 hours, they went from 46 remaining /23s to 39. And from 431 /24s to 426. So folks, it is draining quickly. Wow.

PADRE: Steve, the funny thing about this is - so we've been hearing about IPv4 address exhaustion for the last 15, I want to say 15 years. And the first round of doom and gloom scenarios was solved by NAT. Once we started NAT'ing things, we got a lot of that back. But if you look at what's actually being used out there, it's not like - it's not saying that all the IP addresses on IPv4 are being used. It's just saying that they're allocated. In fact, I got this graphic here, this is from xkcd. They did this a while back. They mapped out the Internet and who owns what. And there are huge...

Steve: Oh, cool.

PADRE: ...huge swathes of this that they're holding onto it because they're holding onto it, but there's nobody using it. Like the one I like to use is this. This is Interop, so this is

the group that I worked with a lot. It's a networking conference. It used to be huge. It's in Las Vegas, New York. They had one in Moscow, Berlin, Tokyo. And we had, we were one of the very first to request IP address space. So we had a Class A, 16 Class B's, and, like, 150 Class C's. We had a very nice chunk.

Steve: Nice.

PADRE: And when we started to hit address exhaustion, what the lead engineer, a man by the name of Glenn Evans, what he did was he actually returned all the space that we weren't using to ARIN. Which is what you're supposed to do. There's not supposed to be a black market because the idea of having a black market means that all of these players are now incentivized to hold onto IP address space that they're not using. And this is what we've been seeing, which is as it gets more scarce, these entities are no longer willing to return unused address space, even if they have a huge chunk of it, because they realize the going rate for an IP address now is something like $6,000?

Steve: Yup.

PADRE: It's - no. No.

Steve: It's funny, too, because when I lost my T1s, I lost the network which GRC's servers knew I was using. So that caused me to - I had to run around and make a bunch of changes. It happened that I was poking around within the various sets of known IPs, and ShieldsUP! has a blocked nets list, that is, over the years there have - I've heard from various organizations who have said, you know, who the hell are you to be sending probes into our network? And I've said, whoa, whoa, whoa, okay, calm down. I can black out your network if you don't want GRC's benign probes to enter your network. And so as it happened, I was looking at that list again, and I saw a /8 on there. And I thought, huh? U.S. Postal Service is Network 56. So they have 56.0.0.0/8. And somewhere in the past I was instructed, do not probe the U.S. Postal Service with ShieldsUP!. And I said okay. And let's see, the USGAO, they didn't want me probing them either. They've got a /16.

PADRE: Yeah, these are all USA government addresses.

Steve: Right, right. So really interesting. So would you tweet that xkcd to both you and me, to @SGgrc? I'd love our listeners to be able to see that. That's a cool chart.

PADRE: And that was 2006, by the way. But a lot of that still holds. That green area is supposed to be unallocated. And of course that's all gone. That's all been allocated. You know, Steve, I probably shouldn't say this...

Steve: Well, well, and remember when we were first messing around with, what was that crazy - Hamachi. The Hamachi...

PADRE: Oh, right, right.

Steve: ...network. That used five-dot because five had never been allocated. It was completely unused. The brilliance of that was that many people who wanted to use Hamachi were in 192.168, or maybe they were in 10-dot. And so that meant that they couldn't safely use either of those spaces because it might collide with your own local network. So by using something that was completely never used, Hamachi essentially was able to set up a virtual network of unallocated IP space, which of course has since been allocated.

PADRE: That's actually exactly what I wanted to talk about because I deal with some people who they do routing for a living. These are the BGP table guys, and they're very good at what they do.

Steve: Cool.

PADRE: But they know, and they all have their pet range, that some of these addresses are never used. They're held on by large corporations or large entities, and they've been sitting dormant for years, decades for some of them.

Steve: There's just no traffic on those IPs.

PADRE: No traffic whatsoever. So they know, if they have a temporary event that lasts, like, three days, they can advertise those routes and just steal them for three days and then put them back, and nobody will be the wiser.

Steve: Very cool. What a hack.

PADRE: So we've come down to that.

Steve: What a hack. So it's funny because ARIN instituted what they called "four phases of exhaustion." And they said Phase 1 began in February of 2011. So, what, more than four years ago, when ARIN received its last /8 from the IANA. That is, IANA - and so, for example, that five-dot, that's a /8. And so there were a few of those that had just, you know, the IANA had never had to give out. And so it was February 2011 when the IANA said, here you go, ARIN. We got no more /8s. So that was Phase 1.

Phase 2 began about a year, a little more than a year and a half after that, in September of 2012, when ARIN received the three remaining - and they said "/8 equivalents." So that means three smaller networks that together were the same size as a /8, but not a single network in terms of contiguous IP space. Phase 3 began about nine months later than that, in August of 2013, when they received two remaining /8 equivalents. So two networks that were a total of that many IPs. And then 4, that is, Phase 4, began in April of last year, 2014, when they received one remaining allocation equivalent to a /8. That is, you know, a bunch of networks that together were a /8, as there was just less more to go around.

So here we are with /23s and /24s. So that's all ARIN has left to allocate are little, I mean, you know, that's what we have behind a NAT router. We don't all have 253 IPs in use behind our NAT router, but that's how many they're essentially now giving out, and they're down in the hundreds, or in the tens in the case of one bit larger networks, the /23s.

Oh, and I got a big kick out of this. If you can click the link there at the end of that ARIN note, it's called IPv6 Depletion. Someone tweeted this, and I thanked them. This is just a - it's a kick. I had to enable scripting. But you'll see that it's counting down. And so this is a little bit of a joke on the IPv6 exhaustion problem and how many remaining IPv6 addresses there are and when we can expect to run out of those at the current rate of consumption.

PADRE: Wasn't the story I could give an IPv6 address to every molecule in the planet Earth, and I still wouldn't use them all up?

Steve: It might even have been bigger than that. I mean, this is - 128 bits is a huge, huge allocation.

PADRE: You know, Steve, this brings up a question, and this is, all the gear that I've been playing with the last 10 years is dual stack. It can do IPv4; it could do IPv6. And actually most of my tools now dual stack. And so the idea was, as we got closer to IPv4 exhaustion, enterprises and businesses would lead the way, and they would just naturally go with IPv6 because it makes sense for them to prepare for the future. We've kind of seen it, but not really.

Steve: I know.

PADRE: I mean, we've seen companies make a big brouhaha to get press. But they still dual stack all their gear. And there're very few pure IPv6 operations on this planet. For me, it's just a lot easier to remember IPv4 addresses. I still don't have the knack of doing IPv6. Everything I've done with IPv6 has been with some weird translation tool that I'm using in order to make it work properly. I mean, is that - do you get the same thing? I'm never going to memorize an IPv6 address.

Steve: It is exactly how I feel. I'm having people ask when ShieldsUP! will support IPv6. And my answer is, boy, I would love to have nothing else to do, rather than rewrite the whole NanoProbe system for IPv6. And I'm not being facetious. I mean, I would enjoy that a lot But unfortunately, it's just me, and we all know that, as soon as I get done with SQRL, I'm immediately back to SpinRite 6 and 6.1 and 6.2 and 6.3 in order to get that code updated. And then who knows what. So, yeah, I would love to do it. But right now I'm 100% IPv4. And unfortunately I have a huge code investment in IPv4 that would require a great deal of rewriting.

PADRE: Yeah. I'm the same way. I would love to say that I'm IPv6.

Steve: It's just inertia.

PADRE: Yeah.

Steve: It's just inertia.

PADRE: And the fact that I know all of my important IPs by heart in IPv4, that keeps me going back to them. If I need to test my network, I'm going to be pinging an IPv4 address.

Steve: Yup.

PADRE: I will never ping an IPv6 just out of sheer laziness.

Steve: Well, and we're talking inertia. Also the problem is IPv4 is never going to go away. I'm using my 16 IPs. Level 3's not going to take them away from me. I'm using them. I mean, so it's like anybody who now absolutely has to have more space is going to be forced to either buy them, buy slack - which we know exists. We've talked about the gray market in IPs, I mean, there's even a dealer that is, like, serves as an intermediary, you know, negotiating between people who have IPv4 space and who want it. And people would rather buy it than deal with switching to IPv6. So at some point they're going to have to.

PADRE: IPv6 is something I want everyone else to do. You should all go to IPv6.

Steve: Right. Exactly.

PADRE: I'm going to stay on 4. You should go to 6.

Steve: Exactly. Oh.

PADRE: Now, see, this next story, I don't know if I should shake my head or laugh with glee.

Steve: I know. So the Hacking Team. There's a notorious hacking team based in Italy that was hacked. The Hacking Team was hacked. And, I mean, major hacked, to the tune of 400GB of private data from their servers was dumped in a torrent and made available publicly. And the next story is an amazing tidbit that was found in there that we'll cover. But first let's talk about these guys a bit. They had their Twitter account taken over. They still apparently don't have control of their email servers. They didn't take it very well. For hackers, you'd think they'd have a little bit more of a sporting attitude. They threatened people. They've accused them. They've denied clear facts. Again, as I said, they've not been very sporting about their own attack.

But one thing that happened that came from this was that, I mean, it's been a treasure trove of information about their clientele. And, for example, one of the things that we learned is that the U.S. FBI has spent - it's hard to even imagine this - nearly a quarter of a million - wait. Three quarters of a million dollars, $775,000 since 2011, buying hacking penetration spyware from these guys. Because what they - they're a global marketer and seller of spyware. They've got something called - it's sometimes known as RCS, what they call their Remote Control Service, also known as Galileo. And I've also seen one of their tools referred to as Da Vinci, which is their premier spying product.

And we've talked about remote access trojans. It's a standard RAT in that it is able to siphon off data and intercept communications of that local machine where it's installed prior to its encryption. So they deal with the encryption problem by getting in and tapping before it's encrypted. It can record Skype calls, emails, instant messages; log keystrokes typed into web browsers, obviously before they're encrypted; can of course switch on the victim's webcam and microphone and spy on them that way.

And what's really interesting is that this Italian Hacking Team maintains an office in the U.S. And there is a government contractor known as Cicom or Cicom, C-I-C-O-M, USA, that logs from our intelligence agencies, the FBI and the DEA, show have purchased surveillance technology from a government contractor under this name of Cicom. And their address and phone number is identical to the Hacking Team's U.S. office address and phone number.

So this is just a very thinly veiled cover for a government contractor that's actually the Hacking Team, selling technology globally. And one of the things that was revealed was that they are saying no to nobody. They're selling this stuff to very repressive regimes in the world that are, like, that are formally banned from receiving this technology from a supposed forthright Italian hacking company.

PADRE: You know, Steve, the thing that really worries me about this is Hacking Team, they want to portray themselves as they're a security firm, just like any other security firm.

Steve: Legitimate. Legitimate, yes.

PADRE: A legitimate security firm. And there are companies in the United States that the FBI has spent far more money with. I'm thinking of, like, Gigamon. Gigamon sells these very high-quality, very high-capacity TAPs that I think the lowest you can buy a box for is, like, 200 grand. And they buy a lot of those. Or Juniper, Cisco, HP, they all sell security appliances that are pretty cool.

Steve: We know they're big on tapping.

PADRE: Exactly. But what kills me about this, about Hacking Team, is this is not a tool that you could misuse. It's black hat all the way. There is no legitimate use of a RAT. There is no legitimate use of a zero-day, except to do something that you're not supposed to do, especially if you're a government.

Steve: Yup. Yup. So what was really interesting, and this just happened, is that among what was found in this 400-gig download was evidence of, and all the details, on a heretofore unknown Flash zero-day vulnerability. These guys had it. These guys knew about it. We don't know who they have sold it to. But it immediately became public knowledge when this 400GB of data that was exfiltrated from them, due to them being hacked - oh, and by the way, also in there are some password files, the own hackers' passwords, which are extremely unimpressive. I've taken a look at them, and they're, you know, they're literally, they're like variations on the word "password," believe it or not. I mean, it's just like, whoa.

PADRE: 1234?

Steve: Okay, you know, like numeric zero in P-A-S-S-W-0-R-D. It's like, whoa, really, guys? Anyway, CERT now has the vulnerability. It is a - and of course we dove into rather deeply last week's emergency out-of-cycle patch for Flash that Adobe released. Well, they may be doing another one soon because the CERT posting links to a tweet which contains the file. I downloaded it and checked it out. If you click that RAR file, you get an archive with a very nice Read Me that explains the entire exploit and contains the ActionScript 3 code to pull this off. And if you do it with a sample web page, it launches calc.exe on your Windows machine in Chrome.

So Chrome is not patched against this. It's a zero-day. And they've had it, and it wasn't public. We don't know who they sold it to. Maybe they were using it themselves. But we also know that they do make these things available to various agencies who want - who purchase zero-day exploits in order to install their own software on other people's machines. So this is just so amazing that, you know, in this windfall of data, we found a zero-day that, I mean, an exploit with complete how-to-use-it code that who knows how many tens of thousands of dollars they were getting to everyone they sold this to. And everyone's vulnerable to it right now.

PADRE: Steve, were you at Black Hat last year?

Steve: No.

PADRE: They had a speaker by the name of Daniel Geer. And one of the things that he said he'd love to see the U.S. government implement is this idea of it, as an entity, because it's the only one that has the resources to do it, buying up zero-day and then immediately making them available to security researchers in the United States. And it sounds like they're doing half of that. They're buying up zero-days. They're just not making it available to security researchers in the United States.

Steve: Well, and we've also known - I don't remember now what the source of knowledge is. But, for example, Microsoft provides the U.S. government with advance knowledge of problems with Windows before they tell the rest of the world. And we've always been wondering, okay, why? You know, why do they get that, and before the public does? You know, it would be handy.

PADRE: It would be very handy. And, you know, not just handy, but this is kind of table stakes now of, I mean, you can't - I'm sure there are people who look at a tool like this, and they say, oh, this is a great way for us to gather intelligence. But at the same time, I mean, if you are working in the U.S. government, you also have to understand that this is something being used against the very corporations and the individuals who make up our constituency and the people we're trying to protect.

Steve: Right.

PADRE: It doesn't make sense to sit on something like this. You know, a zero-day for a plugin that's on every browser being run, almost every browser that's being run in the United States, that sounds like something that you would probably want to get patched immediately.

Steve: Well, yes. And it also means that while you have that and nobody else does, you can sell that for some serious money to, no doubt, all the clients that you've got on the global stage that want to be able to penetrate other people's machines.

PADRE: The funny part about the story is Hacking Team, they say that this isn't going to do them in. They're not going to shrivel up. They're not going to go away. But their source code's on the Internet.

Steve: Yeah.

PADRE: So unless they create something entirely new, their exploits won't work anymore.

Steve: Yeah.

PADRE: Well, of course, assuming everyone patches their machines, which we probably won't.

Steve: It's certainly the case that no one's going to pay for it because now they've got source for what they were trying to sell before. So...

PADRE: I wonder if you could - if you're an oppressive government, can you get a refund? If you just bought a package like a week ago? I mean, it seems like you should be able to stop payment on that.

Steve: Yeah. I would bet refunds are probably a nonstarter. So in the other big news, the Intercept on Sunday released 45 classified documents which laid out in, I mean, in such detail, I feel sorry for the NSA, finally. I mean, this is not just, like, claims, like sort of the obscure slides that we saw of funky network drawings that, you know, don't really make too much sense. Scroll to the bottom of that article that you just pulled up, and you'll see, these are the links to all the documents down there at the bottom that are - each one of those is a multipage slide of frightening detail. There's like a user's manual sort of in the middle somewhere. It's the unofficial, I think it's called the XKEYSCORE or something, yup, there it is, XKEYSCORE User Guide, showing people how to submit

queries to this KEYSCORE system.

So first of all, remember, what the NSA's XKEYSCORE system is, is it's their global taps on the Internet infrastructure, on the main high-bandwidth backbone, the undersea fiber cables that connect. As they emerge from the ocean, there's an NSA hub that they go through that taps them so that they're able to be monitored. What we now know, we now know virtually everything from these documents.

PADRE: Wow.

Steve: It's amazingly comprehensive. I mean, as I said, I feel sorry for the NSA.

PADRE: Email, phone number, MAC address, domain, everything.

Steve: Yes. So, for example, the system, the XKEYSCORE system consists of Linux OS running - or Linux software, I should say, running on Red Hat Linux Servers with Apache Web Server and MySQL database. They are clustering file systems using the NFS file system with the "autofs" service. CRON runs the system's scheduled tasks. Admins connect using secure shell (SSH), and use tools like "rsync" and "vim" to manage the software. They connect to the XKEYSCORE over HTTPS using a standard web browser - well, I should say Firefox, but IE is not supported. They log into the system using a userID and password, or they can use public key authentication.

The system consists of, now, some of these slides are dated, like six years old. So as of 2009, that are the dates on some of these, there were a hundred - we now know 150 global sites. So 150 sites scattered around the world where as many as, now, in aggregate, 700 servers were located where lower bandwidth TAPs had fewer servers, and higher bandwidth TAPs had more servers. In general, they are a full-feed TAP. That is, they are intercepting and storing the full feed that goes by. And since that is a torrent of information, they are only able to store three to four days' worth. And so that's their target.

And one of the things that we've learned is that an NSA operative with no legal foundation, that is, no explicit permission to enter a query because essentially there's no time, because this data spools off of the end of the storage in three to four days, you don't have time to go through the paperwork required to get approval for every query that you make. It's just - it's impossible. It's impractical. So anyone with access to the system is able to submit a query.

The query goes to a local node and is then replicated across the entire XKEYSCORE system. And then the results of that query flow back. There are - they call them "micro-plugins," which anyone can write. They have a custom programming language called Genesis which generally performs the sorts of tagging that they need. Basically, somebody can use Genesis to create a simple piece of code which does string and item offset and value matching on known packets, and then compile that into a micro-plugin, and then propagate the micro-plugin across the entire XKEYSCORE system to install that in order to perform traffic analysis.

So this isn't - so as the traffic comes in, it is analyzed, packet by packet, by these micro-plugins, which tag the traffic as what it is. It's email, it's Yahoo!, it's Gmail, you know, like what network it's from. Basically, as I remember, like up to 10,000 different tags can be applied to packets. And so you're able to surf on pre-identified tags in order to pull data that is still in the database before it's pushed off just due to its age based on metadata. And the metadata is being acquired on the fly. So it's like a complete X-ray

into this XKEYSCORE system and exactly how it functions. Wow.

PADRE: Now, the interesting thing about that is they've been saying, to justify the system, they've been saying, look, we can store the metadata, and we'll only go back if we get a court order for that particular set of metadata because it's attached to a suspect that we're looking at.

Steve: Right.

PADRE: What we're seeing, if you actually look at the documents and understand what they mean, they can't actually do that. If they only have the ability to store X days of data, and a court warrant takes X plus Y days, then there are going to be many, many cases where they're storing bits of data that they shouldn't have, or they're analyzing bits of data that they shouldn't have, waiting for a warrant, assuming they're going to get it, which they may not always do. So, yeah, that's - hmm. It's no longer just a technology problem here. We're talking about - this is the logistics of spying in oversight.

Steve: Yes. And this is indiscriminate collection of everything. Basically, the entire Internet, we now know, the entire Internet is now tapped. There's 150 - and remember, this is six years ago. This thing has probably grown dramatically, even since then. So some sites are receiving as much as 20TB of data per day, storing it, and it's getting pushed off the end. Maybe they've increased their retention period from what it was six years ago, so that they are now able to store more. We know that, you know, hard drives are cheap, and servers don't cost anything, and they have a big budget to work with. So, but this is the shape of the system. The entire Internet is tapped.

PADRE: Six years ago they didn't have a million-square-foot building in the middle of Utah that's basically a big hard drive. So they've probably expanded that a couple of times already.

Steve: Where the big problem is keeping it cool, you know, running a river through it in order to keep it cool.

PADRE: It's not just keeping it cool. They've actually had arc lightning in the datacenter because the densities are so high, and there's so much power going to cabinets, and they're so squeezed together. One of the issues they had at the very beginning is they'd get these arcs between cabinets that would take out an entire row.

Steve: Wow.

PADRE: So there's actually lightning in the NSA datacenter. Well done.

Steve: Oh. And the news just keeps on coming. WikiLeaks published, has been, over the course of the last week, been releasing a series of details about the NSA's intercept of various foreign governments. France and Brazil were earlier leaks. The most recent one was Germany. And our friend Bruce Schneier, who's a well-known cryptographer and, I mean, famous in the security industry, he blogged on Friday, he said: "On Friday, WikiLeaks published three summaries of NSA intercepts of German government communications. To me, the most interesting thing is not the intercept analyses, but the spreadsheet of intelligence targets." And there's a link there, Padre, if you want to bring it up.

"Here we learn the specific telephone numbers being targeted, who owns those phone numbers, the office within the NSA that processes the raw communications received, why

the target is being spied on - in this case, all are designated as 'Germany: Political Affairs' - and when we started spying using this particular justification. It's one of the few glimpses," writes Bruce, "we have into the bureaucracy of surveillance."

And, wow. Again, this is just - it's one thing to sort of, again, have an Edward Snowden slide that says, oh, you know, yeah, Germany is on the list of people we're spying on. And, oh, Angela Merkel is one of our targets. But here is a list of phone numbers, I mean, the last four digits were blocked out with X's in what was published.

But, wow. I mean, it's - there's no way to deny, when Germany looks at the phone numbers they know of that their officers have, and here they are on a list of communications that the NSA has been actively tapping for some length of time - ouch. And Germany's Der Spiegel is also very unhappy. They've got a lengthy article where their main argument, or their main unhappiness, aside from us, is that we have in the past told agents of Germany when we have encountered problems within the German ranks of people who were leaking communications. In other words, we've told Germany that, well, okay, don't ask us how we know, but there's somebody over here who you really need to put in a less sensitive office. So they've relocated their own intelligence officers when we've told them that their officers are leaking sensitive information that they shouldn't.

And anyway, the Der Spiegel article, I've got a link in the show notes, is really eye-opening because, in fact, their main opener says: "Revelations from WikiLeaks published this week show how boundlessly and comprehensively American intelligence services spied on the German government. It has now emerged that the U.S. also conducted surveillance against Der Spiegel." And in there they talk about a specific instance where information was provided to Germany by the NSA about a German officer who was then relocated to some department studying the history of something. Anyway, very eye-opening. But wow. I mean, this is, I mean, this is the world we live in. And I guess we have to assume that governments are now doing this to each other equally. But, boy, to have this leaked like this really does seem like a black eye. Again, I feel sorry for our own agencies.

PADRE: There's going to be people in the chatroom who are saying, "I don't feel sorry for the NSA." I mean, this is horrible. But as you said, this is new for us because we're getting an inside baseball look at what our intelligence services are doing. But you've got to figure, if any government has an intelligence service, this is what they're doing. That's their job. That's what they were created to do. So that part's not shocking. I can't imagine that when the American representative went up to his German counterpart and said, "Oh, and by the way, so and so, you might want to give him a less sensitive position," that's sort of like a wink and a nudge type thing of we're doing our job better than you are, just FYI.

Steve: I know. I know. Actually, I really do, I recommend this Spiegel article to listeners. I've got the link here in the show notes. It's - wow. Yeah. And it must have come from Bruce Schneier's blog entry. So it's definitely one to look at because it's like, woo, here's what's really going on. And again, you know, just to be clear, it's just a detail. It's like, okay, wow, you know, it's one thing to just have it be said. But it's another thing to see a list of the phone numbers.

[spiegel.de/international/germany/the-nsa-and-american-spies-targeted-spiegel-a-1042023.html]

PADRE: Right. To know it actually works.

Steve: That's beyond plausible deniability, to have that level of detail.

PADRE: It's one thing to be told that all governments spy on all other governments. It's another thing to actually see the intelligence that's coming out of the spying.

Steve: Right, right. So in another controversial move, Windows 10, which continues to get nearer to release - the last date I saw was end of this month, end of July it was supposed to drop - Windows has expanded their WiFi Sense, which is what they call it, so that it now, apparently with our permission, but not with any granularity, shares users' WiFi passwords with their Facebook, Outlook, and Skype contacts, and vice versa. So this has really concerned people who are concerned about privacy.

I mean, this is why, you know, one of the things that I always had was my cable modem was on - my cable modem connection was connected to my wireless router, and my inner sanctum wired network had no WiFi because you really want to keep those things separate. And I'm glad I did that, and it's why I'll be bringing up a FreeBSD UNIX router in order to create similar disjoint networks that have absolutely no contact with each other because WiFi is getting scary. I mean, we already saw this with iOS, where, and I've talked about it on the podcast before where, okay, it's kind of a convenience that I didn't have to give another one of my iOS devices my crazy impossible WiFi password, yeah. But it's also a little spooky when you don't have to give one of your devices your password because it mens somehow they found each other through some cloud service that we hope is secure.

And that's what Microsoft is doing. If you use this, Microsoft's own FAQ says that, if you choose to share this information, it is sent via an encrypted link to Microsoft, who then stores it in their servers. So that is to say, your WiFi password is in the cloud. And there's just no way that I'm going to feel comfortable with that. I guess I mean that if it were a network that I really cared about. You know, I just - I don't care about my WiFi network because it's on a network whose security I don't pretend to have control over, but I don't care because it connects to nothing else within my secure perimeter.

But Microsoft's FAQ for WiFi Sense says: "When you share WiFi network access with Facebook friends, Outlook.com contacts, or Skype contacts, they'll be connected to the password-protected WiFi networks that you choose to share and get Internet access when they're in range of the networks, if they use WiFi Sense. Likewise, you'll be connected to the WiFi networks that they share for Internet access, too. Remember," writes Microsoft, "you don't get to see WiFi network passwords, and you both get Internet access only. They won't have access to other computers, devices, or files stored on your home network, and you won't have access to these things on their network."

Now, of course, to me, those additional perimeters don't feel secure to me. It just, you know, this whole notion of, oh, look, it all just works. Extreme Tech really took Microsoft to task on this and really made a good point of saying that, unfortunately, what this is doing is, it's really softening our notion of security. It's like, oh, you know, look how easy it is. When people visit, when my friends come over, they're just on my network because, well, you know, they're my Facebook friends, or they're my - I have a Skype contact in common with them, and so forth. So, okay, I'm not turning that on.

PADRE: I can see why they think this might be a good feature. Let me speak from the other side. The idea that you would never share the password, you wouldn't write it down, you wouldn't say it out loud, and you wouldn't feel the need to keep it simple so that you could share it with somebody else, that could be a net gain of security.

Steve: Yup, yup, I agree.

PADRE: I think you absolutely put your finger on the two things that make it a nonstarter. The first is the fact that there's no granularity. You're either sharing with everyone on Facebook or nobody on Facebook.

Steve: Right.

PADRE: That's easy to fix. I mean, I'd love to see a feature where you say, I would like to share with these contacts if they're willing to share with me. Because then now you have sort of a community, which is, if I'm in your part of the city, I want to be able to use your WiFi. If you're in my part of the city, vice versa. I also don't like the fact that I don't have enough details on how my network credentials are being stored in the Microsoft cloud. They might be able to make me feel safe about that; but, as you said, right now no. That's something that should stay inside my network and never leave. On the plus side, they did a couple of things right. They did make it opt-in. So it's off by default.

Steve: Oh, good.

PADRE: Unless you turn this on.

Steve: Good. I'm glad to know that because in my notes I noted I did not know if it was default on or not. And it's super dangerous if it was on by default.

PADRE: Right. I mean, can you imagine if it just started doing this for everyone in Outlook, Skype, and Facebook without you knowing it? I mean, that's a disaster. But, yeah, you do have to turn it on. And when you turn it on, it actually warns you. It gives you a little warning of what it's about to do. So that's good. Now, they could fix this. Now, if, Steve, let me throw a little bit of a theoretical here. If they made it granular, which I think that's actually quite easy to do...

Steve: Yeah. That's the first thing they have to do.

PADRE: Right, yeah. And if they gave you - if they showed off a process by which only a hash is stored in the cloud. And if they actually showed exactly how they're protecting your network from someone coming on and breaking out of the VLAN or whatever they're doing to try to keep them Internet-only, would this be a feature that you could say it's usable, and the net gain is you're no longer writing down passwords.

Steve: Yeah, they can't be storing a hash because I think they actually have to give the other machine your password. Now, as we know, a password also has a hex representation. So the ASCII is turned into a hex representation. That's what they're probably storing because that's all they really need to keep. But that is the actual password that they're giving to another machine. And my concern is, you know, it looks like just packet filtering walls they're putting around them, saying oh, no, we're not going to give them file and network access. It's only going to be port 80 and 443 or whatever the Internet means. Maybe it's a routing thing, where they're being sure to only route then out your gateway and only to the gateway and not to other IPs within your network. It'd be interesting to see. But I'll bet you they're doing it with some sort of packet IP-based firewall.

PADRE: I could see this working if they were to create a custom firmware, like something based on DDWRT or Tomato, where there's actually a hook between the OS and the router, where you could specify that, yeah, it's going to set up a VLAN, and you have a

guest VLAN, and you have a private VLAN. And also that the decryption of the hash actually happens inside the firmware of the router, so it's not happening on the OS, so the actual key never hits, the unencrypted key never actually hits the Windows machine trying to connect. But, I mean, that would require work.

Steve: Yeah. Well, yeah. I mean, I guess the concern is, that I'm seeing raised, it just is that, as I said, we're making this look easy, where one thing after another is, oh, you know, we're choosing ease of use over security. And but I take your point well. And that is that, for example, I can't share my password in any practical fashion because it's 64 characters of gibberish that, you know, it's like I use GRC.com/passwords for my WiFi password. And I can't share it with anybody. I've got to arrange to somehow cut, copy, and paste in order to get it to somebody who's visiting. Of course, I've solved that problem by having a guest WiFi that has a much simpler password.

PADRE: Of course, if you want to use the WiFi at my house, the password is 1234. So feel free. Drop on by.

Steve: So, okay. There's something that I ran across this week that I got a kick out of, and this was, I guess it was July 2nd. So, yeah, just middle of last week. A new RTF, or candidate RTF, was submitted by someone at Akamai. And they're proposing it as an extension to the specification that will be part of TLS v1.3. As we know, 1.2 is the current version of TLS that now is like the preferred version for the Internet. That's what you want servers to be accepting connections with and clients to be using. What this is, is something people have talked about for a long time.

And I got a kick out of it because they stole the trick from Bitcoin. The whole idea of mining bitcoins is that you have to do work. One of the longstanding proposals for mitigating various types of denial of service bandwidth floods is somehow require the client to do some work. This has also long been proposed for antispam. That is, the problem with email is that there's no charge, not even a micropayment, not even a penny. It's free to send email.

So instead, imagine if there was some way to make somebody sending you email expend some energy. That is, make it in some way expensive. And it wouldn't necessarily be expensive in terms of money. It'd be expensive in terms of processing time, so that you could not have a spamming server that just blasted the Internet and was able to do it with very low return, taking advantage of just it being a numbers game, that they're just able to get a low percentage, but that's enough to make it worthwhile. So this is a solution that Akamai is proposing wherefore, if you wanted to establish a TLS connection to a server, upon sending that first connection establishing packet, the so-called "client hello" packet that we've talked about in the past, a server so equipped could send back a challenge saying, if you want to connect to me, you've got to solve this puzzle.

And so this is called TLS Client Puzzles Extension. And the server can specify, exactly in the way that this works with Bitcoin, in the same way that Bitcoin's hardness of SHA-256 hashing has grown over time, these puzzles use SHA-256 or SHA-512 or a memory-hard as opposed to a processing-hard puzzle, where the server is able to specify how much work it wants the client to do. The client must then, on its end, crank away for some length of processing time to solve the puzzle and then present the server it wishes to connect to with the answer in order to proceed.

And in the RFC they explain that this isn't something that all servers would always do. But when a server was under attack, it could then switch into client puzzle mode because the alternative is to have to discard packets because it's just in a flooding situation anyway, where it's having to do, you know, a statistical packet discard. So their

argument is, if instead it began responding with every request to solve this puzzle, and only accepting requests that were able to, while it does put a burden on the client, the alternative is the client would have discarded packets and couldn't get to the server anyway.

So I just - it was fun to see this finally actually looking like it might be added as part of the TLS v1.3 protocol. And I got a kick out of the fact that in the RFC they referred to Bitcoin, some Bitcoin dialogue where the way of doing processor hardness or memory hardness is being discussed.

PADRE: This is actually something that I saw in F5's security appliances, their UTM set, which it allowed for three different ways to back off a DNS attack. The first one was it just added waits. So it would add waits to certain clients, just random waits, which would decrease the amount of traffic. The second thing was, as this proposal says, it would add a little bit of work. It would ask the client to do something in order to continue with its request. And the third thing was it would actually ask for some sort of human interaction. And of course, if you're running a DDoS attack, there will be no human interaction.

Steve: Right.

PADRE: So it's nice to see that get folded into TLS.

Steve: At the protocol level, yeah. So one of the things that happened last week was I shared with everyone my discovery, thanks to someone who tweeted it, of the PrivacyTools.io site. And one of the upshots of this is that there was a lot of Twitter traffic, both to me and to the PrivacyTools.io guys, about, hey, wait a minute, you know, why aren't you recommending LastPass? I think they like 1Password better. And also what about Threema? Steve likes Threema as a secure instant messaging client, and you guys aren't recommending it.

So I just wanted to say that, you know, there are many good secure solutions. There are other password managers, certainly, other than LastPass. And I'm always being asked about them. I just don't have time to dig into deep technology for all these alternatives. I know LastPass. I've checked it out. I did a podcast about it. I know how it works. They really do seem to be doing as good a job as anyone could. So I'm comfortable recommending them.

The PrivacyTools.io guys have responded, saying that they don't recommend closed source, U.S., cloud-based password managers since great alternatives exist. And I'm not - I wouldn't take any issue with that. I completely agree. You're welcome to use what PrivacyTools.io suggests. I'm comfortable with LastPass. Threema is closed source, they also replied. And so they're only going to recommend open source tools. I've looked at Threema. I understand the technology. I'm comfortable recommending it. So if you'd like to use Threema, I think it's a great solution. If you'd rather use what PrivacyTools.io suggests, I don't have any problem with that, either.

So I guess my point is that many good solutions exist to solve these problems. The ones I like aren't the only ones. And those guys have explicit criteria for recommending what they are recommending. And I certainly honor those, you know, that criteria, as well.

PADRE: And we tend to like what we like, once we find that it works. So, I mean...

Steve: Right. Just like IPv4, Padre.

PADRE: Exactly. Exactly.

Steve: You and I are staying with it.

PADRE: I'm staying with it.

Steve: So Ron Houk tweeted a nice tip. I mentioned last week that I had switched from Adblock Plus over to uBlock because it's a more aggressive blocker, and I wanted to try it for a while. It's also multiplatform. It's available over on Chrome. Well, I guess Adblock Plus is, too. But I just liked it because it was more aggressive. He noted that, if you turn the advanced options on, which you can't get to unless you dig a little bit - you've got to bring up, go to the add-ins page, and then go to its add-in, and go to its control panel, and there's something down at the bottom where you say, you know, give me the control deck or something like that. Then there's a checkbox. You turn that on. Then when you go back to their little icon on the toolbar, they've added a "+" in front of the - oh, no. You turn on the "I'm an advanced user" checkbox. Then they put a "+" in front of the requests blocked and the domains connected. Either of those expands the panel to show really cool information about the page you're on and what it did for that page.

So I just wanted to pass that along to our users. I wasn't aware of it. I hadn't tripped over it and discovered it because I'd turned on the advanced option and looked around in those tabs over in the control panel and didn't see any difference. Turns out the difference is over - is your ability to get those little plus signs that allows you to expand the dynamic display of what it did for the page that you're on. So very cool.

PADRE: I like the domains connected option. I'm definitely going to check that out tonight.

Steve: Yeah. For sure. And that's our podcast, right at two hours.

PADRE: How about that.

Steve: So, yay.

PADRE: That was an incredible amount of news.

Steve: Whew.

PADRE: So in the last 30 seconds here, maybe we could do some Q&A.

Steve: Uh...

PADRE: Just kidding. That's not happening.

Steve: Thank you, because I'm exhausted.

PADRE: Well, you should be. I mean, considering exactly how much, how many new stories we just covered. And this was an above-average week. There are weeks when we scrape.

Steve: Yeah.

PADRE: We did not do that this week.

Steve: Yeah. We'll see what next week brings. And I'm glad I'll have you with us again, Padre.

PADRE: It'll be a lot of fun. Of course Steve Gibson is at GRC.com. That's where you'll find SpinRite, of course, the tool that I recommend on this podcast, on This Week in Enterprise Tech. We're going to be doing a special on SpinRite on Know How. So if any of you want to find out the inner secrets of how SpinRite can help your hard drive, your SSD, and if you want to make your own SpinRite station - because that's what we're going to be doing. We're going to be creating a low-cost SpinRite box so that your regular computer could do what you have to do.

Steve: I get a lot of requests for that, Padre.

PADRE: We're making a dedicated SpinRite box.

Steve: Very cool.

PADRE: And it's a low-cost dedicated SpinRite box.

Steve: Very cool.

PADRE: You'll also find 16Kb versions of this episode, transcripts, and of course some great information about security, about SQRL, about the upcoming change-the-world-of-authentication software, as well as an active forum discussing everything under the secure sun. If you have a question, you can submit them at GRC.com/feedback. We'll make sure to get them into a future Q&A episode. And maybe your question will be picked for one of Security Now!'s Q&A specials.

You can find all the versions of Security Now! at our website, at TWiT.tv/sn, which of course is a place where you can subscribe to get every episode in the format of your choice, into the device of your choice, automatically, each and every single week. You could also use our apps and watch us live. There are upcoming apps. Since we switched over our website, we're going to be building APIs. In fact, if you watch Coding 101, you'll see exactly how we use the API to build apps for iOS, for Windows, for OS X, and even for Android.

Remember we gather every Tuesday, 1:30 p.m. Pacific time, that's 4:30 Eastern, and 20:50 UTC, at live.twit.tv. Until next time, I'm Father Robert Ballecer in for Leo Laporte. Thanks, Steve, and we'll see you next week on Security Now!.

Steve: Thanks, Padre.