SECURITY NOW!

Transcript of Episode #518

HORNET: A Fix for TOR?

Description: August's annual DefCon and Black Hat conferences never fail to surprise, worry, and entertain. This year is no different. Though still two weeks off, reports of interesting security troubles are beginning to surface. This week Leo and I examine the week's news and take a close look at a topic the Internet press got completely wrong: HORNET, a new design for an Internet Anonymity network.

High quality (64 kbps) mp3 audio file URL: http://media.GRC.com/sn/SN-518.mp3
Quarter size (16 kbps) mp3 audio file URL: http://media.GRC.com/sn/sn-518-lq.mp3

SHOW TEASE: It's time for Security Now!. Steve Gibson is here. We are going to explain what StageFright is all about. And then he's going to take a look at what was covered by the mainstream media as an alternative to Tor. Turns out it's not quite right. Steve, as always, has the explanation, next.

Leo Laporte: This is Security Now! with Steve Gibson, Episode 518, recorded Tuesday, July 28th, 2015: HORNET: A Fix for Tor?

It's time for Security Now!, the show where we protect you and your loved ones online with the guy in charge, the Explainer in Chief, Mr. Steven Gibson. Hello, Steve.

Steve Gibson: Yo, Leo. Great to be with you again, as always. It's funny, we're approaching Black Hat and DefCon, which is always producing a lot of news. But it's also my reminder that we're approaching the end of another year of podcasts. So I thought, when exactly was that? And it's like, two weeks from now we'll be wrapping up Year No. 10 and heading into No. 11. So, yeah. And it's funny, too, because I went all the way back to 2005, it was August 19th was the first podcast. Eighteen minutes. And I said, "Ah, those were the days."

Leo: That was Honey Monkey. Was it Honey Monkeys? Honey...

Steve: I think it was the Honey Monkeys. Or it may have been the Windows Metafile. Those were like the first two.

Leo: Oh, yeah, yeah.

Steve: And because I remember our original concept, well, let's just sort of sit down for 30 minutes and talk about the week's events. It's like, okay.

Leo: It's evolved. It's evolved over time.

Steve: It's not like Security Now! is an exception to the rule. All of your podcasts are now multi-hour, relaxed, multiparty discussions. And that's sort of - that's what's evolved. I mean, unless you really hold it, like with TNT, where it's expressly just a half-hour news show.

Leo: I have to fill time because, I don't know why, because I have to. I like to. I like to do long shows.

Steve: Yeah. Yeah, I know.

Leo: Yeah, actually it makes sense that your 10th anniversary would be in two weeks because this is Episode 518, so it'll be 520 will conclude Year 10. We missed one show in that 10 years.

Steve: Yes. Yeah. And we used to be on...

Leo: One.

Steve: God darn it. And we used to be on Wednesdays. And so the 19th is, no, the 18th is a Wednesday back then. 2005, August 18th was a Wednesday. And so we'll be hitting on the 17th.

Leo: Well, what do you want to do for the 10th anniversary?

Steve: I want to say, "Hi there."

Leo: Okay. That's it, huh?

Steve: Okay. So...

Leo: Hi there.

Steve: Yeah, I have no big plans.

Leo: Okay.

Steve: I just wanted to say, hey, look.

Leo: Oh, 10 years. But that's like a big one. That's not eight, it's not 11, it's 10.

Steve: That's true.

Leo: We should do something. I'll get cupcakes. Any excuse for cupcakes.

Steve: Eat them on the air, that's right.

Leo: Yeah, there you go.

Steve: It's funny, too, because when you were mentioning them on MacBreak Weekly, I have a buddy who's, like, he just loves cake. But he's just - he's single. He's by himself. And I thought, you know, I wonder why Mark hasn't thought about cupcakes? That's just, like, perfect. Little mini...

Leo: Happy cakes, yeah.

Steve: Yeah.

Leo: You don't have to eat a whole cake.

Steve: And actually, he's really good, too, about not overdoing. He can have like a jar of cashews, and just have four. That just doesn't work for me.

Leo: No, I can't do four cashews. Can't do it. Unh-unh.

Steve: Anyway. So we have, because of Black Hat and DefCon, lots going on. And a lot will be going on. We have a little brief follow-up for the Fiat Chrysler catastrophe that we broke the news of last week.

Leo: And that was actually a DefCon/Black Hat announcement, wasn't it?

Steve: It is.

Leo: Charlie Miller's going to do his - yeah.

Steve: Yup. So that's an example. The next topic is also, and that's StageFright, the news of a problem with Android phones that affects just shy of a billion of them, which we'll discuss, much in the news now.

Google did an interesting survey that I thought our listeners would find really interesting of the practices of security experts versus non-experts. And I've got a link in the notes for people who won't be able to see the charts because the charts are really interesting. So I thought you and I would have fun scrolling through their multipage document and going, oh, look at those. Oh, look at that. So we'll do that.

I picked up on a little bit of DMCA news, the Digital Millennium Copyright Act, which has caused so much concern because it's just so onerous. However, there was a suit that involved it that was appealed. And the appellate court ruled in an interesting way, and in a hopeful way, frankly, that I hope sets some good case law.

Also, the Anti-Phishing Working Group has a report that sort of surprised me, and it's interesting, about the distribution and the nature of phishing sites. We're always talking about phishing because it's, like, the way of either just spraying out exploits and getting some percentage of people to click on links in email, or so-called "spear phishing," which is where you're really trying to penetrate a specific organization where you use your specific knowledge of an organization or of an individual to whom you send carefully crafted email designed to get them to click on a link. Typically, these often involve then bouncing the person to a malicious website that they don't notice isn't where they're supposed to be going.

Leo: Yeah.

Steve: So some interesting statistics come out of that. Following up on the really cool privacy page that we talked about a few weeks ago, I found a terrific, sort of a security news and information aggregation page that I'll share. Also I've got the right way to silence Windows 10 Upgrade pesterings, since...

Leo: You're a little late. I mean, tomorrow they go away. But all right.

Steve: No, no, no. No, all of the people who have 7 and 8...

Leo: I guess not. I guess not.

Steve: ...they're going to keep getting pestered. And in fact...

Leo: Until they upgrade, yeah.

Steve: ...there's already hidden directories appearing in the users of Windows 7.

Leo: Oh, boy.

Steve: As they're, like, beginning to move components and getting ready. And then we have, of course, a little bit of miscellaneous stuff. And we're going to talk about HORNET, which made the news a lot, and all wrong, in all of the news coverage, because it was being hyped as the fix for Tor. We know the many things that are wrong with Tor. And unfortunately, well, I just stepped on the line. But, you know, so the podcast title is HORNET: A Fix for Tor?

Leo: Question mark?

Steve: Question mark.

Leo: A Fix for Tor? Hmmm? I think not. But we'll find out.

Steve: Yes.

Leo: A little later.

Steve: And why not, exactly.

Leo: Why not.

Steve: Exactly why not is what makes the podcast fun.

Leo: Yeah. Well, and frankly, it's what people tune in for because I read all those articles, and I thought, oh, good. Unh-unh. All right. We'll find out.

Steve: Just because we're up at the first page of the notes, I'll point to this week's Picture of the Week is interesting. It is of the top brand names on the Internet. It is a chart showing the phishing attacks, the number of phishing attacks per brand.

Leo: You mean emails purporting to be from that brand, but not really from that brand.

Steve: Correct. Correct. And it turns out that - I have the numbers down below in my notes, but I think it's 75% of all phishing emails are for the top three brands.

Leo: Wow.

Steve: And...

Leo: Well, why not? Why would you - if you're going to send a phishing email, you're not going to send it from brand 1,000. You're going to send it from the Bank of America or Coca-Cola or something people recognize.

Steve: Well, yeah. But...

Leo: Or Apple.

Steve: But what was interesting to me was, well, Apple is one of the top three.

Leo: Yeah.

Steve: Yeah. What was interesting is that the distribution is as sharp as it is. I mean, it's like three quarters of all phishing email is just three brands that are under attack. So I thought, that's like, wow. Anyway, we'll have full coverage of that. But I wanted to note that picture.

Okay, so just a quick follow-up on last week's big car hacking story. Shortly after we finished the podcast, Chrysler went from go to our web page and download a patch to load into your own thumb drive and good luck to you, to issuing a nearly 1.5 million-car recall. And so apparently you can do three things. You can go to that page, put in your vehicle identification number, the VIN number, and they will tell you if you're vulnerable. And I've seen a number of descriptions. Most of the articles summarize, it's like, you know, these vehicles and others. But it's older cars with an 8.4" touchscreen seems to be the best way to know. But of course you can go to the website and find out.

There are also, I've seen reports saying that they're sending thumb drives to all the owners of the affected cars. But you can also, if you wish, go to the dealer and just say, "I don't know where to plug it in, help me," and they'll do it. But in the meantime, the opportunity to attack the cars has been completely foreclosed because the cellular provider - I'm trying to remember, was it Sprint? I think it was Sprint.

Leo: Yeah, Sprint.

Steve: They blocked the port. In the same way that our ISPs "protect us," unquote, well, they actually do protect us when they're blocking the famous Windows file and printer sharing ports, 137 through 139 and 445. But, for example, they also block 25 to prevent SMTP servers from running inside their networks and so forth. So there are some things they do for themselves, some things they arguably do for their customers. And in this case Sprint blocks the port that was open to the Internet. And there's a - they had to submit a dot.gov PDF containing a timeline of the steps that they have taken, a chronology, basically.

And so in that timeline they said: "A communications port was unintentionally left in an open condition, allowing it to listen to and accept commands from unauthenticated sources. Additionally, the radio firewall rules were widely open by default, which allowed external devices to communicate with the radio. To date, no instances relating to this vulnerability have been reported or observed, except in a research setting." So the good news is - oh, and in fact on Thursday, Chris Valasek, who was one of the two - I'm sorry, on Friday he tweeted, "Looks like I can't get to Charlie's Jeep from my house via my phone. Good job, FCA" - that's Fiat Chrysler Automotive - "/Sprint." So the network availability was shut down to all those. And then, you know, patches are forthcoming.

So, you know, everybody acted as well as they could, given the fact that there was this problem in the first place. And I think this is just the beginning because everyone wants to have everything connected to everything. And security is hard.

Leo: Yeah, yeah.

Steve: And speaking of security being hard, so again pre-DefCon/Black Hat, this will be another presentation made in two weeks, a group who does, like, enterprise-level mobile security, called Zimperium, took a close look at one of the modules that is ubiquitously available in Android. It deals with cellular media, but it's also in other places. For example, Silent Circle had it in their Blackphone. And it was even in Firefox, and all versions of Firefox since v38 are already patched to this.

But they took a close look at the code and, just in scrutinizing it, found 12 problems, about six or seven of which are remote code execution exploits. And what's extra scary is that this thing is running by default. As I mentioned at the top of the show, it's about 950 million Android devices are believed to be at risk from this. Google has already patched this some time ago. But we're back to the problem of back-patching older phones. And that whole process seems to be problematical.

In the coverage that I'm reading, it's generally felt that phones older than about 18 months are relatively abandoned by their carriers. And in fact the ACLU has now filed a lawsuit to get the cell phone industry to face the fact that they're selling computers to people which are now, we're finding, just as full of problems as we've been covering for all of the last 10 years of this podcast, even before smartphones happened, and for decades before as viruses were causing problems. Yet there isn't, because they're sort of consumer devices, and the carriers don't want to continue to take responsibility for them, they're just being left in a condition where they're vulnerable.

And in this case you send the phone - all you need is the phone number of a vulnerable Android device. You send it a specially crafted, multimedia message, an MMS message. And without the user doing anything, just the phone's receipt of the message, Android gets it and does enough parsing of it without it ever being selected or viewed, that there are exploits in that early receipt and pre-parsing path that allow code to be executed. Anyway, so they're calling it, I mean, the module is called StageFright, which I thought, well, that was - did they name this with some sort of prescience of some sort?

Unlike a lot of Android, because performance is crucial, this is not written in Java and therefore able to get any of the advantage of the Java virtual machine protection, where essentially you're writing interpreted code. This is pure C++, where the code is essentially at bare metal, with full access to the system, and full responsibility being placed on the programmer. So Android and derivative devices since v2.2, since and including v2.2, are vulnerable, although those prior to Jelly Bean, which is a smaller portion of the whole, only about 11% of the devices, are at the highest level of risk because there have been other mitigations that have been put in place, things like DSLR or ASLR and DEP, you know, the things that we've talked about that make these exploits more difficult. But, as we know, "difficult" is a reach from "impossible."

So about seven CVE designations have been reserved. Right now they're still blank because everyone's running around, scurrying around and patching these. The guys at Zimperium will give a presentation, show a video of this being exploited, and they do plan to produce and release proof-of-concept code after their presentation. So essentially, we know what'll happen. Proof-of-concept code will quickly be weaponized. And what I'm hoping that will happen is that some good guys will take that and create a benign proof-of-concept test, as we have seen in the past for other things.

For example, the Flash exploit that got loose from the Hacking Team, it contained a proof of concept that launched the calc.exe app. That's typically what they do is they launch the calculator on your desktop, just to show that, look, we just ran executable code that we should not have been able to run. So it would be great if someone would produce a benign, like someone with reputation so we knew they were benign, produced a proof of concept so that people could send their phone a test MMS that would allow them to proactively verify that they either were or were not vulnerable. I ran across, actually it was Simon Zerafa who tweeted a couple hours ago, the patches. And they were in the - boy, I'm blanking on the name. What's the alternative...

Leo: CyanogenMod.

Steve: Yes, thank you, the CyanogenMod.

Leo: I'm getting good at reading your mind.

Steve: You are. Perfect.

Leo: What's the - oh, yeah, CyanogenMod, of course, yeah, everybody knew that.

Steve: And for anyone who's interested, I will tweet this after the podcast because it's your typical code diff where it shows remove this line, add this line, remove this line, add this line. And they're just - it's like three simple little things that, if they were there, this wouldn't have happened. But of course that's the way code patches are, and the way these problems are, is that it's just the person who wrote it, I mean, like one of them is creating a new array of a certain size. And the fix is of a certain size plus one. So it was just one byte too short, probably a null terminator, so that when you filled the array with size objects, you had one extra byte of null that would guarantee null termination, and that would prevent that string from being overrun. But that's not what the coder did. So they forgot the plus one. Whoops.

So anyway, this is, I guess - so what we need is we need to have some, I think an industry-wide look at how cell carriers are going to be dealing with these older phones because they are computers. And, I mean, and it's not like Android is alone. Even Apple has, like, eh, well, you know, we're not going to go - we're fixing things, but we're not going to go all the way back into the dark past because we don't want to. Well, but if the phones are still in use, and they're vulnerable, then they really do represent a target. And, boy, I mean, talk about a target of opportunity. I mean, basically people could just send text messages blindly to blocks of phone numbers that were known to belong to a given cell phone carrier and take over phones. And in fact you can even delete the MMS message after you've sent it. So the malware, it would leave a notification that you'd received one, but what it was could be deleted as part of this.

Leo: Wow.

Steve: So, yeah.

Leo: We don't know, do we, what versions of Android have been patched, what carriers have patched them, which phones are patched?

Steve: This has just happened. And so I think - in fact I think this was yesterday that the news happened. We know that Google has responded, and their code...

Leo: Google says, yeah, that they've put out a patch; right?

Steve: Right.

Leo: And so any Google phone should be okay, I would think.

Steve: Right. Although this does affect nearly a billion. They're saying 950 million Android phones.

Leo: I know I got something last night from T-Mobile on my Galaxy S6. I don't know if it's related or not, though, because they don't tell you. Nobody tells you what the fix is for.

Steve: Yeah. And that's why we need somebody, maybe somebody listening to this, you know, this is the kind of thing I used to do before the podcast and before SQRL and before SpinRite 6.1. I can't take time out to go do that. But all of the freeware that GRC was offering was my quick responses to horrible things Microsoft was doing back in the day. And so hopefully somebody will do this. And, if so, make sure I find out about it, listeners, and tweet it to me so that...

Leo: Something benign, so you push a button, and it sends you an MMS that says, yeah, if I'd been a bad guy, you'd be hacked by now, yeah.

Steve: Correct. Correct. And, then, see, and that will empower users to then go to their carriers and say, hey, this is - you've got to fix this because, I mean, somehow we - I guess it will either be a combination of the legal process demanding that carriers take ongoing responsibility for really bad security vulnerabilities in products that they've sold, even during their useful life. I mean, they're connected to the person's phone. They're receiving revenue from this person. And that connection that the person is paying for can infect them with malware through a widely known path. Seems to me there's a strong argument for saying that the providers have an ongoing responsibility. Oh, and these guys provided Google not only with the news of the problems, but they gave them the patches. They said, "Here, patch this." And Google did.

Leo: Yeah. Or it may be an argument for, if you're going to use an Android phone, getting an unlocked Android phone that you could put your own firmware on, like CyanogenMod.

Steve: Right.

Leo: Not getting a phone from a carrier. Carriers don't really have much to offer.

Steve: Yeah. And I think that, yeah, certainly, as we know, the Android users are a broad class of people. And so listeners to this podcast could certainly do that. But, you know...

Leo: No, no, there's millions of phones sold every day in India and China that...

Steve: Hundreds of millions out there that are just, you know, they're as vulnerable as those that are carried by people who could replace the firmware in their phone.

Leo: The 5.1.1 is fixed. Or not. We don't know.

Steve: I don't know. It just happened.

Leo: Yeah, okay.

Steve: Just happened. And what we need is a test. We need somebody to do that. Maybe these guys will do that. If, in fact, the proof-of-concept code they're coming out with is a benign test, it may be that what we're going to get after DefCon and Black Hat will be all we need. Send this MMS message to your phone. And so it may be easy to make that happen.

Okay. So interesting research from Google. You want to grab that PDF in the second link on this next page, Leo, under "New research: Comparing how security experts and non-experts stay safe online." This was fun. There are some, I guess, not surprising things. But it's nice to see the charts and graphs on the exact numbers. So Google took, through an online questionnaire, they profiled 231 security experts who are defined as having at least five years of self-reported experience in online security, so a pretty good sample size, and 294 regular non-expert computer users, and asked them all the same set of questions. And the results in this very nice PDF, this is being presented, or actually just was.

We just got through with the USENIX Symposium on Usable Privacy and Security. That conference was up in Ottawa. And the key is "usable privacy and security," the idea being that some advice which is just too burdensome - and of course as I was reading this, I was thinking, like, yeah, like turn off scripting. You know, we've been covering for the last several podcasts the problems that people have with running with scripting turned off because it breaks too many sites, and people don't want to selectively turn scripting on, and I don't blame them. But there's a perfect example of asking people to do something which many people just, even though they recognize the danger, smart, you know, listeners to this podcast who understand, it's like, yeah, but it's a pain. Give me something else. And so we've talked about Sandboxie and alternatives.

Leo: I should point out, though, that I don't see that turn off scripting or Sandboxie is in the experts' list, either.

Steve: No. And this is - right. So that gives you a sense for the level at which this is operating.

Leo: Yeah.

Steve: This is, you know, the importance of a password manager, changing your passwords often, sort of more generic security advice.

Leo: By the way, the security measures that are in this chart are ones that at least 5% of each group mentioned.

Steve: Right.

Leo: So, you know, if it's a small number of people mentioned it, it doesn't make it into this chart. But I do think that what this really reflects, if you look at it, and I'll let you do your analysis, is how security concerns have changed over time.

Steve: Yes. And in fact, if you scroll to the top of the next page on the left, they take that chart, which shows various concerns, experts and non-experts, and shows the differential between them.

Leo: Ah, yes.

Steve: Which is a really nice presentation. So, for example, experts, the thing that experts recommend and understand most more than non-experts is the importance of keeping your system updated. That is, the number one thing that experts get which non-experts don't get. And in the text covering that particular issue, I was a little puzzled because it actually said that non-experts were reticent to do updates. And I thought, wow, I wonder where that comes from?

Leo: I know that fact. I know.

Steve: Yeah. But why, Leo? I mean...

Leo: But mostly non-experts think, if you use an antivirus, you're done.

Steve: Yes. And in fact, at the other end of the chart is that. That is the exact inversion. That is, the thing least recommended by experts and most believed by non-experts. And that is that antivirus, as you said, Leo, just have an AV, and now - but, wait, aren't I protected? Isn't that all I need?

Leo: Yeah, yeah. And that's old school. Right? That was, you know, in the day of the zero-day exploit, that doesn't make any sense. But it used to be that was the thing to do because exploits didn't spread like wildfire. You got them from a floppy.

Steve: Yup. Very good point. And if we were to - I'm sure if we were to listen to our own advice 10 years ago on this podcast, we would have been saying, oh, yeah, you know, like - I mean, I remember. An antivirus company was a sponsor.

Leo: ESET, yeah.

Steve: Yeah. NOD32, I think.

Leo: Yup. And Dvorak and I were always talking about Kaspersky in the day. I mean, I'm talking 20 years ago.

Steve: Right, right.

Leo: Up to when Melissa came out. And that was the first email virus. And once they started spreading, you know, fast enough that you could have a zero-day exploit, I think all bets were off.

Steve: Right.

Leo: Most of the people I know, including you, don't use antivirus. Most of the security experts I know don't use antiviruses at all.

Steve: Yup. I don't. Okay. So in the order of going from what experts advise that non-experts don't, down to the reverse, that experts don't advise and non-experts do. So we have update system, most advised by experts, least performed by non-experts. Then, number two, or coming down toward parity, is use two-factor authentication. And in fact, if we look at the individual chart of that, that's something that the reason it's number two in the differential is that non-experts just do not do it at all. That first chart that shows both side by side, experts are strong on using two-factor authentication, and it is way down at the bottom of what non-experts do. That just isn't something that non-experts are into is, again, people who are just using computers and want the computer to stay out of their way, please, they just want to get their job done.

Then a password manager. That's another thing where the experts are really strong on it because we understand that we need unique passwords, and we need high-entropy passwords, neither of which, or both of which, thwart our memories so we cannot remember all of our passwords. So then your only fallback is use a password manager.

And what was interesting is password managers are not trusted. Non-expert users don't like them. They don't trust the idea of putting their passwords in a password manager. It's like, oh, you know, why am I not asking for more trouble by doing that? They don't get it that it is so important not to reuse passwords and to have high-entropy passwords, which thus cannot be memorized, that the gain in doing that is wholly offset from reasonable concern that you've got all your eggs in one basket.

The other thing that experts do that non-experts don't is look at the address bar. Non-experts don't know what that is. I've got a good friend who thinks the Internet is Google, doesn't understand that there is any difference.

Leo: Wow.

Steve: That Google page comes up because I set that as her home page years ago. And she just says, you know, well, where's the Google? It's like, okay, Judy. Yeah. So we understand about looking at the address bar. But typical users don't. Also, we look for HTTPS. Typical users don't know what that is. There's no concept of, like, they're just, I mean, we have to understand, they don't even know what that gobbledy-gook is up there. They click links. They click buttons. It's like, that just, I mean, it's part of my annoyance with the way the Internet has evolved is that should not be even seen. Unfortunately, we know you have to see it. But you shouldn't have to see it. But you do have to see it. But we who understand it look at that. And I'll often, before I start to fill out a form, I'll look up there, manually check the domain, look to see that the security's in place, and then go, okay, you know, these guys took some pains.

Also, be suspicious of everything. That's the last thing on this differential chart where the experts have one of these properties to a greater degree than the non-experts. And that is, we're suspicious. And, I mean, and that's one of the mantras of this podcast. How many times, like, you know, you have to be suspicious. The first lesson from that we've often talked about is never download or accept a download for something you didn't go looking for. Well, if suspicion wasn't your watchword, you wouldn't have a rule like that. Like somebody says, oh, you need to update your Flash. No, don't, don't. You didn't go looking for it. Someone offered it to you. Don't do it. Because why? We are suspicious of everything.

Leo: Yeah.

Steve: And it's unfortunate, but all of our experience teaches that. So now we get into the second half of this differential chart. What are those things that experts are less focused on than non-experts? And now we're going in increasing direction towards the end, which is antivirus. So we're sort of near zero with something that I'm surprised about, which is verified - is it use Linux, or verified software? I'm not seeing where that lines up. But so...

Leo: Yeah, that's kind of weird. I don't know how that lines up, either. I think Use Linux was still more expert.

Steve: Oh, yeah. It has to be, right.

Leo: Than less.

Steve: Right, right, right.

Leo: So we flip when it goes to verified software.

Steve: Yeah, there's no way Use Linux could be over on the non-expert side.

Leo: No, no. Non-experts are not - they're not going to use Linux, no.

Steve: Okay. So delete cookies, for example. That's something that non-experts still sort of think is like the way you solve these problems. Unfortunately, we know that cookies, while, yes, that's a good thing, that's not solving our problems because there's, like, there's supercookies and sticky cookies and come back cookies and boomerang cookies and all kinds of other ways that we have of being identified and tracked, with fingerprinting and, you know, Panopticlick shows us that we have all kinds of trails we leave behind. Then we've got "Don't share info." So the non-experts believe that just sort of keeping your head down and pretending it's not you...

Leo: Tell nobody anything.

Steve: Exactly, just don't do it, that that's going to help them a lot. So then "Use strong passwords." Now, the reason that's there is that that's the only idea they have. They don't really understand fully about entropy.

Leo: Right.

Steve: But it's like, if there's a password strength meter, they're like, oh, okay, I guess I'm trying to get that into the green and get it out of the red zone. So whatever that means, it's like, okay, stronger passwords are good. Unfortunately, they use the same one password pretty much everywhere. So they're defeating some of that. Also, "Visit only known websites." Now we're getting to the stronger end, where the experts are, like, shaking their heads, like oh, okay. So these are people who just don't go. They're, like, they're so afraid of the Internet that they're like, okay, well, I go to Amazon and NBC, you know. And of course we know that good websites can also be distributing, for example, malware through ads. And so that's really not helping you that much. Also, next to the last, next to "Use an antivirus," is "Change password." Which I just have never understood, and everyone knows that I scratch my head at that one.

Leo: And they get that from the IT department; right? The IT guy told them to do that, so it must be secure.

Steve: Yeah. Yeah, exactly. So anyway, if anyone's interested, there are pages and pages of really interesting stuff, and various graphical displays of these parameters. You know, nothing really surprising here. Unique passwords is sort of the same. Strong passwords is pretty much the same. As I mentioned, in the text it explains they don't really trust password managers, which is of course completely flipped upside down. On the other hand, I would argue that password managers imply unique and strong passwords. Otherwise you wouldn't need one. And that's beginning to be sort of in the strongest response to the problem of websites being breached and losing your passwords, thus the only reason to make them all unique, and hopefully making them uncrackable. Thus you need to make them strong. But there we're at the, you know, I mean, that's like the advice of the experts that are willing to exert more effort in the direction of securing themselves.

Anyway, so the whole point of this was to understand what it was that non-expert users felt and compare that to what expert users did, and sort of the idea being to come up with a rationale for what advice should be because clearly there are some misconceptions among non-experts. Those misconceptions arguably, if they don't require extreme measures, fixing those misconceptions would be a useful thing to do, and relatively inexpensive because on some basis it's about the economics of what these things cost users to implement. And in fact the study shows that, if it requires too much effort, and it doesn't obviously help them, they just sort of say, eh, you know, okay, I'm not going to bother with that. It was fine yesterday. So I think it would be fine, you know, later today and tomorrow.

Leo: Yeah. It's pretty amazing.

Steve: Interesting.

Leo: Yup.

Steve: So the DMCA judgment. Whoever tweeted this to me, thank you, because I wouldn't have found this otherwise, and I thought this was interesting the way the court ruled. So this is the dismissal of a claim made under the DMCA. And these are some big companies. This is General Electric. So Judge Emilio Garza, who's the Fifth Circuit Court of Appeals judge in New Orleans, his ruling said that: "Merely bypassing a technological protection that restricts the user from viewing or using a work is insufficient to trigger the Digital Millennium Copyright Act," which is - now, so what that says is, up until now we've believed that messing with the encryption at all, just the act of decrypting something or studying it, for example, like universities want to be able to study this. And many, many researchers have been chilled and in fact stopped, thwarted, by the DMCA.

This ruling says, no, those were not breaking copyright.

So the idea is that he is saying that the anti-circumvention provision only holds if the use would then be a breach of traditional copyright. So the way this all happened is sort of interesting, too. General Electric was involved in this. So the ruling stems from a lawsuit filed by a power company, MGE UPS, as in Uninterruptible Power Supply systems, which was bought by GE in 2001. To fix the machines, technicians have to use MGE's, that's this UPS company, copyrighted software programs. The software can be unlocked with an external hardware key, a dongle. And the dongles have expiration dates, passwords, and a maximum number of uses. Wow, this sounds like a pain-in-the-ass dongle. Anyway, and thus what happened. Years after MGE introduced this technology, hackers posted information on line on how to bypass the hardware key.

Leo: Hmm.

Steve: Yes. Once a key is cracked, the software can be freely used and copied. So in its lawsuit against GE, MGE claimed that a group of PMI employees, that's the company that GE purchased, had at least one copy of their software that was running on a hacked machine. In other words, the PMI guys had used the information from the Internet's hackers to bypass the dongle protection. So this UPS company sued GE, as the parent of the company whose employees did this, for copyright infringement and DMCA, and a claim under the Digital Millennium Copyright Act. And so what the judge ruled was that, "Without showing a link between access and protection of the copyrighted work, the DMCA's anti-circumvention provision does not apply. The owner's technological measure must protect the copyrighted material against an infringement of a right that the Copyright Act protects, not from mere use or viewing."

So in fact there was no - what the judge said was that, since they had a legal right to use the software, and they were not breaking traditional copyright, that the additional protection that the DMCA is trying to bring to bear doesn't apply. So I thought, wow, that's good news for researchers and for people who, I mean, we know about the notion, and we've talked about it often, of fair use under the Copyright Act. It's just like, you know, if I own this, then I have a right to read it or to listen to it or watch it or whatever, under fair use. And so this is saying that the DMCA doesn't prevent us from decrypting something that we want to use that would still fall under fair use of copyright.

Leo: Moving on.

Steve: So we know that phishing is a problem. It is arguably the major attack vector these days is getting people to click on links that take them to a website. My PayPal email address somehow got loose to a phisher, and I get daily attempts. And I'm just amused and bemused. I look at them, and I go, really? You think so? I mean, they're doing everything they can...

Leo: I know, yeah.

Steve: ...to try to, you know, trick me into this. And of course these all involve domain names which are, I mean, some of them they're not even trying. It's like, why? You know, I'm getting this phishing email, and it's giving me a link to something that doesn't even pretend to be PayPal. Although generally they do. It'll be PayPal and some different TLD, or PayPal dot and then a short domain and dot and then a standard TLD. Again, so this is why, as I was saying, experts need to see the address bar, is it does matter where you are. And it's good when an email client will show you where you're going when you click on a link, you know, make that clear, rather than allowing the email just to have a visible click here.

Anyway, there is something called the Anti-Phishing Working Group, the APWG, which sort of keeps track of this. And I found out about this because I follow DigiCert, and they tweeted about this. Being a certificate authority, they're intimately involved with this whole issue because the major attraction that phishers get is by fraudulent registration of domain names, which as I said, look close to, like A-M-A-Z-0-N rather than O-N. And it may be in all caps so that the "O" and the zero look pretty much the same. And a lot of people go, oh, Amazon, you know, and don't notice that it's a numeric zero rather than an "O."

And responsible companies are, for example registering their domain in all of the other top-level domains, when they're able to, specifically to prevent malicious reuse of their domain name in other top-level domains, just to preempt that; and also often times registering all kinds of variations, which may be the result of typos, but can also be, like, a one rather than a lowercase "L." You know, all the sort of numeric and alphabetic substitution that we see going on.

Anyway, so these guys, looking at the size of the problem, through whatever mechanisms they have for collecting these, are able to assert that in the latter half, the second half of 2014, there were at least 123,972 unique phishing campaigns. So just shy of 124,000 during six months. So, I mean, if you divide that by 180 days in six months, and you're shy of a thousand per day, but you're in that ballpark. So, and they define an attack as a phishing site which targets a specific brand or entity on the Internet. And of those just shy of 124,000 attacks, a little more than 95,000 of them used unique domain names. So they're tending not to reuse domain names between attacks, and thus the nature of an attack is a domain gets registered for the purpose of confusing people, of a malicious intent, and then it will be used in a phishing campaign.

One of the things that I found most interesting is the duration of these. It's much shorter than I expected. On the other hand, since I'm seeing a different one every day, I guess I should have expected they wouldn't last long. The average attack uptime is only a little greater than one day, 30 hours. And the median uptime...

Leo: You mean before, like, the bad...

Steve: ...that is, there are as many shorter than as longer than, is 10 hours. So half of them are less than 10 hours, and half of them are greater than 10 hours. So these things are not long-lived. Apparently they bring up a domain. They generate a piece of email and send it out to whatever email target audience they have for that attack. It either works or it doesn't work. And, well, the other thing that happens is it is recognized by those who are actively working to combat phishing, and it's taken down. So these registrations that are being made are short-lived, not only because the campaign has a short life, but it has that short life because as soon as it's recognized to be used for malicious phishing purposes, it's deregistered. So these bad guys then have to go do another one. And they use it as long as they can, but it's not going to be long.

In fact, one of the things the working group found was there seemed to be some slacking off relative to history in the rate at which these were being taken down, sort of as if people are just getting tired of, I mean, think about the number of these domains, 95,000 domains, more than 95,000 domains in a six-month period are malicious, used for phishing. They're being registered, and then they're needing to be proactively deregistered by people who are working to combat this.

And then we see skewing, which is interesting. This is not just a flat distribution. There's a distribution such that 75% of the malicious registrations occurred only in the top five TLDs. And I guess that's not surprising because that's where the attacks focus. For example, the same 75% of attacks, or I meant the same percentage, 75% of attacks are occurring against only the top 10 targets on the Internet, of which the top three are Apple, PayPal, and a Chinese site, Taobao, T-A-O-B-A-O, dotcom. I'd never heard of it, so I looked at it, and it's a Chinese page with looks like ladies fashion apparel or something. Not being a Chinese reader, I couldn't tell, and there's no English to help us out there, but lots of girls and apparel. And so those are the top three: Apple, PayPal, and this Taobao are the targets.

And so the top five TLDs correspond to the same amount, the same fraction, three quarters, the top three quarters. And that's .com - again, not surprisingly, because that's where Apple and PayPal are - .tk, .pw, .cf, and .net. So even though there's been a huge explosion of TLDs, what, it's like 272, 272 different TLDs have seen attacks in them. And in the second half of 2014, one quarter of those, about 56 of those were brand new. Actually one fifth of those, sorry, were brand new. So new TLDs are, as we know, are being created. We have .info, there's .guru, there's dot all kinds of things. And those are not immune.

But generally people are focusing on the top 10 properties on the Internet, of which the top three are Apple, PayPal, and this Taobao. And those three companies experienced independently 20,000 phishing attacks against their services and brands. So it may be that - and I've never looked at the inside of how this operates. But it might be that Apple is reporting the domain, you know, like Apple has some visibility into the phishing attacks against their own Apple.com domain and are doing constant takedown notices in order to remove domain names from the registries in order to get these things out.

And then I have a chart here about attacks by industry, where in general about 40% of all attacks are e-commerce, with the next largest being banking at 22%, and then next after that is money transfer at 20.7. So, and together, what's that, that's 42 plus 40, so 82% of the attacks are e-commerce and banking and money. That's, of course, in my own little example...

Leo: Makes sense. Go where the money is.

Steve: ...of the ridiculous PayPal phishing that I get every day. That's what they are, they're just all kinds of random things trying to get me to somehow give up my PayPal credentials by any hook or crook that they can come up with. So I thought it was interesting to put some numbers on something that we're talking about all the time because that's, famously, that's how RSA had its huge breach is some administrative assistant clicked on a link in email, and that let the bad guys in. And we believe that's how the Office of Management and Budget, the OMB, in the U.S. federal government got the same attack. It was a phishing email targeted at them.

Leo: Uh-huh.

Steve: So a fabulous site, Leo, you're going to want to bring this up...

Leo: Okay.

Steve: InfoSecIndustry.com. This is now the companion to the one I found a couple weeks ago, or I found only because somebody was nice enough to tweet it to me, remember, that was PrivacyTools.io. Anyway, this one, InfoSecIndustry.com, is a really well-assembled security, as it says, information security news aggregation page. So they've got a section for all kinds of alerts that they're following - US-CERT alerts, current activity, WordPress security advisories, Amazon Web Services security advisories, Microsoft security advisories and the Microsoft security response center postings, Apple security advisories...

Leo: Wow.

Steve: ...Linux security advisories.

Leo: This is great.

Steve: Cisco and Oracle and Adobe. So it's just, like, all in one place. Those are all scrollable so you can go back in time. Then they follow - a bunch of security tweeters are being followed: US-CERT, Symantec Security Response Center. They follow Bruce Schneier's blog; Brian Krebs, who of course we talk about all the time; Graham Cluley, Tripwire, Kaspersky, and the Hacker News tweet. And then a number of news sites - Krebs on Security, Bruce Schneier, Naked Security, Dark Reading, Graham Cluley, ZDNet, and on, you know, just like another 20 of those. And podcasts - mine, Security Now!...

Leo: Oh, good.

Steve: SANS Security, Southern Fried Security, and others. Oh, and then, finally, is Events. They maintain a calendar of specifically North American events and also those from around the world. And of course we've got, right here at the top, is Las Vegas, Black Hat and DefCon coming up. So anyway, I just wanted - so many people liked and raved about the PrivacyTools.io page, where in one place was a bunch of really good privacy focused tips. I know that our listeners who are focused on information security would get a kick out of this page. And I've been tweeting back and forth with the guy that put this together. I told him I loved it, and I would mention it this week. So bravo for this, InfoSecIndustry.com. Definitely another keeper.

And we've got, of course, Windows 10, as Leo, you mentioned at the top of the show, officially coming out tomorrow. It made some news today because yesterday they released another security update which broke Windows 10 for a bunch of people. The news was that disabling a network adapter could cause a problem; and that, if you went to the classic Add/Remove Programs page, double-clicking on something which would normally remove it would instead crash Windows Explorer, and that that was reproducible. I tried it on mine because I did update yesterday, following the news of the update, which I tweeted about. And it did not crash mine. I had a Lenovo laptop, and I thought, I can safely remove the modem. So I double-clicked on that, and Windows 10 survived without any trouble. So it doesn't affect everybody, but apparently it affects some.

And people were concerned that, well, this whole notion of updates happening all the time. Last week we talked about the troubles people have had removing the upgrade notification. And there are people who are just not ready to go to 10. I mean, I'm not actually going. I've got it on a laptop just because I'm curious, and I want to see what it's like and decide whether I'm going to go to 7 when I leave XP, or make the leap to 10. I'm not sure yet. I just sort of have to get used to the new look and feel, give myself a chance to do that.

But anyway, Microsoft's acronym is GWX, which stands for Get Windows 10. And so there have been various postings, we talked about it last week, where initially you could uninstall the "upgrade," unquote, the Windows update to Windows 7 and 8 machines, which was annoying people. But then we had the news that having done that, the second Tuesday of the month, which was now three weeks ago, or three Tuesdays ago, counting this Tuesday as one, so two weeks ago, it came back, and that there wasn't something obviously possible to disable.

So the good news is, for those techies here who do want to disable this, who don't - oh, and I guess my advice of hiding it doesn't work, or doesn't work robustly. So you cannot just use the Do Not Notify me option down in the tray. There is a registry key and value that can be created. Apparently, if you've got this, you've got something under the HKLM\ Software\Policies\Microsoft\Windows\GWX. So that's the path in the registry, HKLM\Software\Policies\Microsoft\Windows\GWX. Under this GWX key you create a DWORD value named Disable, with capital D, lowercase "isable," so Disable, all caps GWX, and give that the value of one, and you are done. You have disabled Get Windows 10. And whatever Microsoft chooses to do, apparently they will honor that. It works today. We can hope it continues working.

And I did get a tweet from someone who found in his root directory of a Windows 7 machine a hidden folder that contained a setup EXE and other Windows 10 upgrade things. So Microsoft is already installing stuff in people's Windows 7 and presumably 8 machines, in preparation for the big event, which many people are a little less excited about.

Leo: Yeah. I don't think you'd get that folder unless you accepted the invitation.

Steve: Ah, good, good. Let's hope.

Leo: He probably accepted the invitation. But they said they will start preloading, you know, because it's 3GB. You don't want to have 1.5 billion people downloading 3GB all at the same time.

Steve: Yeah, I think you probably can't. I think it's a safe bet that that would even strain

Microsoft a little bit.

Leo: Yeah, yeah.

Steve: Yeah. So, miscellaneous stuff. I finally read "The Martian." Which our listeners have been tweeting me about, that we've talked about on the podcast often.

Leo: But I know you don't like to read a book unless, like an operating system, it's been out for a while and fully tested.

Steve: That's correct. We want to get the bugs out. Actually, there was a bug in this one.

Leo: Yeah.

Steve: I need to say, though, that when I say "read," I mean raster-scanning a printed image. So, that is, a two-dimensional image of text. I raster-scanned the entire thing in order to read the book the old-fashioned way. But I found a glaring mistake. As I mentioned, there's a bug. And this is not a spoiler, for those who haven't yet read it, because everyone knows, everyone knows the premise is that Matt Damon gets - oh, I mean, sorry, Mark Watney...

Leo: You're jumping the gun a little here.

Steve: Mark Watney. Actually, this is "Bourne in Space," or actually it's more like "MacGyver in Space."

Leo: The movie is "Bourne in Space," maybe.

Steve: Yeah.

Leo: Andy Weir is going to come on the show. The author's going to come on The New Screen Savers, this week or next, to preview the movie because he's...

Steve: Oh, neat. Neat.

Leo: ...overseeing the moviemaking.

Steve: And I didn't understand, did you know that this was initially just something he was sort of doing, he was, like, posting the chapters for free on his website or blog. Just sort of as he wrote them, he would post them.

Leo: Yeah.

Steve: And he began to be surprised by the number of people that were reading them and raving about them. And actually he made the comment that it was an interesting process because the feedback on each chapter helped him to, like, was really useful for helping him to refine where his writing was going, and even to go back and fix some things.

Leo: Andy was on "Triangulation 163." He tells that story, if you want to watch the interview.

Steve: Oh, neat.

Leo: He was great, yeah.

Steve: Neat. So the problem I had was that he had, Mark Watney, who is the mission's botanist and engineer, because they're both cross-trained and have multiple training so that they can be more useful for the mission. It was a six-man mission. He was one of the six. And he of course got left behind, and this is his story of survival on Mars. So he's incredibly inventive and understands how to grow things and how to fix things. And but there's this glaring mistake, and that is that he for some reason is not aware that the 14 satellites orbiting Mars are imaging him all the time, and that of course Houston, NASA, will know that he's alive because he's moving stuff around outside.

And so it was interesting because this was like a revelation that he took with some surprise when he did arrange some communication. And I won't talk about that because that would be a spoiler. But it was like, oh. You know, I was like, okay, wait a minute. You know? We know who he is. And he absolutely knew that there was this huge network of satellites that was imaging Mars. Yet he didn't think to, like put stones out in a pattern of, like, somehow, like scratch in the sand, "Hi, guys," or "I'm still here" or something. Anyway, I sort of thought, well - but other than that, and I have to say to all of our listeners who have not read it, it was great. I really enjoyed the book.

Leo: Yeah. It's really good, yeah.

Steve: And not a long read. I read it over the weekend. I started on maybe Friday, and as often happens, it sucked me in. It is definitely a page-turner and hard to put down.

Leo: So let me get this clear. You're so averse to holding a paper book that you scanned the pages in?

Steve: With my eyes.

Leo: Oh.

Steve: In a raster scan.

Leo: Oh, I get it. Oh, I - it's called "reading." I've heard of that, but I didn't know anybody did it, yes.

Steve: It's called reading, yeah. The problem is that we seem to have repurposed that word. Instead of it being read to me, I read it myself.

Leo: Oh, okay. The raster scanning was performed by your eyeballs.

Steve: Correct.

Leo: I get it. I get it.

Steve: That is the time-honored tradition know as "reading," is raster scanning.

Leo: It is a great book. I don't know what it would be like to read a book. I haven't done that in so long.

Steve: Yeah, it's the same, Leo.

Leo: Oh, all right, all right.

Steve: It's like, yeah, I think, you know, I grew up reading. We didn't have videogames, and I wouldn't have been a videogame person back then except for the technology, anyway. I just - I had such a thirst for knowledge, Mom would drop me off at the library, at the San Mateo Public Library. That's where I would just spend my days, just down in the stacks, reading. And that's what I did. So I think, you know, I get it that there are people who are not readers, and there are people who are eBook listeners. Anyway, this is a great book, so.

Oh, and a piece of errata. I know better. I knew better. I do know better. When I said last week that you could use System Restore to back out of the CryptoLocker problem, it's like, what was I thinking? Of course not. System Restore only stores the system files, not all of your personal documents. I was confused and didn't apply my own knowledge because someone had written to us, we covered it on a Q&A, that he had successfully used System Restore to recover his files. Well, maybe we got confused about what he was saying, or he was confused. Anyway, I ended up being confused.

Many people corrected me, so I wanted to officially correct the record and say, yes, of course I know that System Restore doesn't restore, you know, it's not a full rollback of everything on the machine, only the things that may have gone wonky in Windows, the Windows system files. So thank you, everybody, for your corrections. I certainly stand corrected.

And a quick SpinRite story. This is from a listener of ours, Jason Robohm, who's in Allen, Texas, who wrote last week on the 15th of July, Wednesday. He said: "Home Security DVR HD Failure/Recovery." It's interesting, we've had a few DVR testimonials recently. He said: "Steve, I wanted to pass along a testimonial on SpinRite. We have an alarm company" - ADT Pulse is his alarm company - "provided DVR for the cameras outside our home." Wow, that's - so he has that for his residence. He said: "Recently the DVR's alarm was periodically sounding, all hours of the night as one would expect - thank you, Murphy," he says, "letting us know that the hard drive was in an error state. Rebooting the DVR only seemed to make the hard drive error alarms more frequent. So we just unplugged it to get some sleep. After three failed ADT service appointments to replace the DVR unit" - and when I was reading that, I thought, what? They couldn't fix it?

But it must be that somehow they just missed the appointments. Because then he says: "I figured it couldn't hurt to crack the DVR open and see what was inside, turns out a simple 1TB WD Green SATA hard drive. Seeing that, I quickly grabbed my trusty copy of SpinRite, attached the DVR's hard drive to my PC's SATA cable, and booted up into SpinRite. 90 minutes later, running on Level 2, SpinRite declared it was done. Interestingly enough, SpinRite did not show any errors or corrected sectors. But having heard that this is common when SpinRite induces the drive to fix itself by showing it its own problems, I shrugged and placed the SpinRited HD, the hard drive, back into the DVR unit and powered it up.

"Guess what? No hard drive alarm, and a fully functional DVD, or DVR. All of the recorded video streams are still there, as well. Thank you, Steve, for such a great product, and I know we all look forward to SpinRite 6.1+ versions in the future. Sincerely, Jason."

Leo: Nice.

Steve: And so for what it's worth, and for everybody who's wondering, yes, 6.1 is the next thing I do. As soon as I get SQRL to a point where I can put it down, I will, believe me, believe me, I am looking forward to getting back to 6.1.

Leo: You don't mean "put it down" like put it to sleep forever. You mean - never mind.

Steve: I mean, yes, good point. Not put it down in the sense of we're sorry, but your dog is now...

Leo: Yeah, no, not that put it down, yes.

Steve: ...no longer functional.

Leo: Steve Gibson, Leo Laporte, and the Green Hornet. Bzzz. Actually, it's not green, it's just a HORNET.

Steve: Not a Green Hornet. I wish it were green. Maybe that would be good. I'm not sure. So we've talked about Tor a lot. We did a podcast about it, The Onion Router [SN-070], and how it operates by successively encapsulating encrypted data as it moves through this network. Normally you have an entry node, a middle node, and an exit node. And the idea of onion routing, just real quickly to run through it, is you obtain the public keys for all three nodes. Then you take what you want to send anonymously to some distant point. You first encrypt it in a way that only the last node can decrypt. Then you encrypt that for the middle node. And then you encrypt that for the first node. And so we think of this successive encryptions as layers of an onion because now you've got something triply encrypted where the last encryption is for the first node you will encounter.

So you sent that to the first node. It and only it knows how to decrypt it. So its being sent is secured by, like, way secured, you know, your data is down, buried under three different layers of encryption, where the keys are known only to three different routers, onion routers, out on the Internet somewhere. So the first one decrypts its outer shell, and only when it does that is the destination of the middle node known to it. So nobody seeing that original packet going to the first node would know where it was going to go to after that. It decrypts it, finds out where to send it only then, and sends it there. Now you're at the middle node, and only it knows how to decrypt that middle layer. It decrypts that. Now it knows where to send it to the exit node, so it decrypts that. I mean, so it sends it to the exit node. The exit node decrypts that third and final innermost layer. Finally it's the data that you originally wanted to send, and that contains the IP address of where that should be sent, so the exit node sends it off.

Well, if all that sounds like a lot of work, you're right. And that's why Tor is notoriously slow. And in fact, that's why we're only using three nodes. If you use many more nodes than that, which could - and circuits can be built. This three-node thing is called a "circuit" through Tor. You can build circuits with more nodes. But, oh, lord, does it slow down because, first of all, the system - Tor is now being heavily used. Two million users a day use the Tor network. It's grown dramatically.

And we talked about it a couple podcasts ago, when we were looking at some researchers were trying to figure out how to improve its anonymity. Because what we've learned is that, if you really scrutinize this kind of onion routing network, there's just no way to prevent the traffic association problem, that is, to prevent associating incoming and outgoing traffic. And there's two stages of that, that we've discussed in the past. One is where you don't have any idea where someone's traffic is going, so you need to look at a lot of nodes. But then the improved version is the confirmation attack, and that is very powerful. When you believe you know where the data is transiting, if you mess with it, like deliberately delay a packet going through, and then let it go, and look at it come out the other end, it's much easier to confirm a suspected route through this kind of anonymizing network than it is to just, without any knowledge, try to guess where it's coming out.

The problem, of course, is that in order to do this, you have to have visibility into the entrance and the exit. But it turns out that the existing circuit builders are not very good about deliberately choosing nodes in, like, really disjoint autonomous networks, autonomous systems. Like, for example, if you have Level 3, which is a big Tier 1 carrier, if they happen to have the server running both a Tor entry node and, somewhere else in

Level 3, an exit node, because they're one network provider, they can see all the traffic on their network. And so, presumably, could law enforcement, if they said, hey, we want to look at traffic here and here. It's easy to do. Whereas if it goes across the world somewhere to a different provider, and across a different portion of the world to still a different provider, then you've got stronger anonymity guarantees. On the other hand, boy, is it slower. Again, this really runs slowly.

So, HORNET. The name HORNET is an acronym for High-speed Onion Routing NETwork layer, H-O-R-N-E-T. What these guys did, and what all the press got wrong - the press was all jumping up and down, it's like, yay. They were talking about the speed, which is impressive, and saying that this, like, solves the problems with Tor. Well, first of all, this isn't deployed. No one's ever built one of these. It doesn't even exist, first of all.

Leo: So they're just assuming it's a lot faster.

Steve: Well, what they did - yes. What they did was they built a software router. There's a project, DPDT, I don't remember, it's a four-letter acronym dot org. And it's a software router SDK, essentially. And it is a high-speed packet-switching platform where you can develop, you know, it's based on user space Linux, or I'm sorry, it's BSD. And there is a FreeBSD version. And what this allows you to do is, in user space, to write a little bit of code and get very fast routing. What they realized is that interrupts, the interrupt service overhead of - unfortunately, I know too much about this because I spent a lot of time looking deep into this years ago.

The interrupt service overhead, which is how normally operating systems handle I/O, is so significant that, for fast packet routing, just dealing with interrupts is where you lose all your time. So they shut down interrupts and use polling, which is like the old-school way. No one does that anymore. But it turns out, if you do polling with proper buffering, you can write very high-speed routers yourself in software. You don't need switching fabrics and all kinds of state-of-the-art stuff.

And what these guys got was 90-some, 96 or 92 gbps of routing speed using this technology. But they just built one. And then they gave it - they had, like, a packet generator going on the input, generating synthetic HORNET packets, and measured how many it was able to handle. So that's where they got this. So this doesn't exist. And so what they did, as the acronym says, high-speed onion routing at the network layer, that's what's different.

So what's that mean? We've got to step back a little bit and talk about layers because the way networking is organized is in a hierarchy of layers. And this is one of the beautiful things about the way this architecture was established from the beginning. The very bottommost layer is the wires, the so-called physical layer, the actual, like the definition of the electrical signals, and like the actual wiring that carried the signals from point to point.

Then the level above that, above the physical, is the so-called "data link layer." And we all have those. An example is Ethernet, ATM is another one, where that's the protocol that uses the wires, the physical wires, and carries the payload for the higher layers. And so we have physical layer. Then we have Ethernet, where Ethernet is a protocol standard. And that's where we have things like ARP and MAC addresses.

And what Ethernet typically carries is the IP layer, the Internet protocol. And in fact, as our listeners who've listened for a long time will remember, what ARP does, the Address

Resolution Protocol, is it's what associates IP addresses and Ethernet endpoints. It maps IPs to MAC addresses, which is how network adapters are identified at the Ethernet protocol level, at the data link layer.

So now we have IP packets, which are carried by Ethernet packets, which are carried by the physical layer. And the IP packets contain the - and that's the network layer. Then they'll contain the transport layer, which is something like TCP or UDP, which are the protocols which run on top of IP. And then an example of the layer above is the so-called "session layer." And that would be like TLS or SSL. And in fact, SSL's name, Secure Sockets Layer, what that says is it's a layer to provide security to the underlying layer, which is TCP. And then above that is the so-called "application layer." An example of the application layer is HTTP, the Hyper-Text Transfer Protocol, where we move web pages around. So that's this hierarchy.

Now, Tor is written, essentially, at the application layer. That is, you have all of that other stuff underneath it, and then you're running the Tor protocol like you would be running HTTP. In this case it's the Tor protocol. So what these guys asked, and at the academic, let's find out layer, or level, is what if we did onion routing, not at the application layer, not up at the top, on top of all this other stuff, where Tor does it. What if we did it at the network layer?

Now, that's way low. Remember that we have, going downwards, we have the application layer; then the session layer, where we would apply security; then the transport layer, where we provide connectivity; then the IP layer, which is the Internet protocol, the actual packet routing layer. So what these guys are saying is, what if we implement onion routing there, at the IP layer?

Now, this is interesting for a number of reasons. I've already given away one, and that is 96 gbps? So what these guys showed was they could design a secure onion routing protocol that suffered none of the performance bottleneck and problems of Tor. That's their achievement. So again, this doesn't exist. You can't go get it. Someone, it was Mashable who said, oh, yeah, use the HORNET browser instead of the Tor browser. It's like, there's no such thing. But, you know, maybe someday. But probably not, and we'll explain why in a second.

But so what these guys did was basically the onion routing protocol, as I described, what they did was they came up with a new packet type or payload for the IP packet where the payload contains a succession of onion routing headers. They came up with their own onion routing protocol because what they wanted was, in order to get speed, they needed to minimize the work that each node did. Because now we're used to thinking in terms of Tor nodes as a server, an Internet server running the Tor protocol. Remember, because it has to be application layer. So it's like a web server. This is a Tor node server. And so it's running on top of everything. And as a consequence, it's doing lots of asymmetric, which is to say public key crypto, which we know is slow, but the current onion routing protocol requires that.

These guys flipped all that around. First of all, they are using elliptic curve crypto, and they're using the one I chose, the good one, the 25519 Edwards curve, which is the one I use for SQRL, and that we're seeing more people adopt because it's a win, has both short keys and is very fast. And bulletproof security, as far as anyone knows. What they do is they still establish a circuit, using a variation of the Tor protocol. They reach out to the various nodes. But we need to think differently now. Now we're actually talking about a hardware router. I mean, it's got - there's software there, or firmware, if this ever happened, that would implement HORNET. But it's a router in the way we think of Internet routers, meaning lots of interfaces coming into it, the routing table.

And right now, today's routers do nothing. They do no processing except they look at the destination IP, they look at their routing table, and as quickly as they can, they stick that in the queue of the interface that the routing table tells them to send it to. That's all they do. So imagine if the Internet's routers were enhanced. We've talked about how security is not part of the Internet protocol; that it's, as we can see, layered on top of the existing low-level protocols. These guys asked the question, what if we upgraded, updated, had second-generation routers? And of course, unfortunately, it'll never happen. But it's an interesting thing to think about. What could we do?

So they designed - now, what HORNET is, is an academic exercise in the idea of onion routing at the IP level, not four layers up at the application level. And the idea is you need minimal work per router, meaning that it's only doing symmetric crypto on the packet. The packet comes in. It has the key because it's established that. It uses that to do fast symmetric crypto on the packet, and it forwards it to the - oh, and it takes the result of that, obtains the IP address where it's forwarded it to, which isn't otherwise visible, and then sends it on its way. That router at the IP level, again, with symmetric crypto, runs that across the packet, obtains the IP address, and forwards it. So each phase is taking a wrap off of the onion to obtain the IP address and forwarding it at the network level.

So what they have successfully shown is, in a different world than we live in, unfortunately, where all the Internet routers have some smarts and some time - because, I mean, this isn't zero overhead. Right now we're worried about IP fragmentation ballooning the size of the routing tables and that not only won't they fit, but it requires more time in order to find what you're looking for in larger routing tables. I mean, so in general, routing is going to be a challenge. And of course IPv6 is going to be a mixed blessing. Hopefully we get some routing table consolidation, sort of by having a chance to redo this after the Internet is now several decades old. But if routers had this protocol, what they could achieve is much higher onion routing performance. And they worked out all the details.

The problem is, if you have near real-time, then there's even more ability to do traffic pattern analysis. And they don't address that at all. Now, one of the things you do get is you can have many more nodes because your per-node overhead is dramatically reduced. So no more, like, three hops. You can have 14. In fact, they mentioned 14 - that's why I had the number in my head - in their paper. I mean, you could afford to have this thing jump around a lot.

But the other thing is that you're inherently now routing encrypted information where looking at it at any point along the way tells you nothing about its future. So if the Internet's routers had this, then you don't even really need to set up a circuit. You just route this where, I mean, you do need to determine where its destination is so that you're able to build the header, the onion header, in order to contain all of the keying material for these routers. But you're able essentially to bounce this around a lot more without incurring the significant per existing Tor node overhead. The problem is that, again, you still need to form your circuits intelligently to minimize observation.

So, much as this is - so what they've done is they've solved the performance problem. If all routers had this, then they would dramatically solve the surveillance problem because right now, even though there are, what is it, 3,000 Tor nodes, well, okay, that's still nothing compared to the number of routers we have. Imagine if every router on the Internet were also capable of doing this kind of onion routing. Then you just have, just by scalability, you'd have a vastly bigger problem trying to track down and deanonymize the traffic.

What these guys did was they designed an onion router that can run, essentially, at full routing speed, without introducing substantial overhead that the Tor network has. And if something were to change, so that we had routers that did this, then we'd have a next-generation Tor network that ran much faster. But unfortunately, you know, we're not even moving to IPv6, let alone adding dramatic protocol level crypto stuff into the packet management and switching of our routers. So nice idea, you know. Now we know it's there.

Leo: It's an academic exercise. It's not...

Steve: Right, right. Exactly. It was never meant, I mean, it doesn't exist, one of them. They made one.

Leo: That's pretty funny. I didn't realize that.

Steve: No, out of an SDK. It's like, okay, good, yeah. Wow, is that fast.

Leo: Proof of concept, yeah.

Steve: Now we have a paper, yeah.

Leo: Well, I'm glad you explained that, and I won't wait around for a HORNET.

Steve: No.

Leo: Steve Gibson is at GRC.com. That's his website. That's where you get, oh, everything. Information about SQRL. Of course SpinRite, the world's finest hard drive maintenance and recovery utility, soon 6.1. Well, someday.

Steve: Yup. Yup.

Leo: I don't want to say "soon." But soon could be many things to many people.

Steve: As soon as I can. As soon as possible.

Leo: As soon as possible. There you go.

Steve: Really, truly.

Leo: Yeah. What else? Lots of good free stuff there, and including this show, 16Kb audio, 64Kb audio, written transcriptions from Elaine, and the whole works there, kit and caboodle, GRC.com. Next week, good lord willing, the creeks don't rise, we're going to have a Q&A. No?

Steve: With DefCon...

Leo: DefCon, it might not be.

Steve: DefCon and Black Hat. We may squeak one, we may squeak a Q&A in before the avalanche hits the week after.

Leo: No reason not to ask a question, if you've got one, at GRC.com/feedback. Or go to Steve's Twitter account, ask them there, @SGgrc. And he's very active on the Twitter these days, which is nice. You will be...

Steve: And let me just say, oh, my god, it's the people who send me stuff. I'm, you know, the podcast, I can't quite say that it writes itself because I still have to do a lot of work pulling it all together and tracking all the stuff down. But, boy, I don't have to go looking for this information anymore.

Leo: That's nice.

Steve: We've got a bunch of really great tweeters who, like, they must spend their time doing that. I'm glad I'm able to work on SQRL, and they feed this to me so I'm able to pull it together for a weekly podcast. Works perfectly.

Leo: You'll be on The New Screen Savers Saturday to do a preview on Black Hat, and that should be interesting - and DefCon. You don't ever go to those, do you?

Steve: I don't. And I wonder, I mean, I'm glad someone does. But we have the Internet now. I'm able to look. All of the papers are published. All of the itinerary of everybody. I mean, we know what the show's going to be. It's like, you know? And everyone would rather have me working on SQRL so I can get back to SpinRite. And get SQRL done. So, yeah, I think that's the best use of my time. I will absolutely be covering it. But it's fun to go to conferences; but, you know, like I went to the RSA conference. Thank goodness I met Stina and Yubico. So good things can happen there. But, nah, I just think you get almost as much these days, you know, just staying glued to the 'Net.

Leo: Yeah, that's why I don't go to conferences much anymore, absolutely.

Steve: Yeah.

Leo: They're over-covered these days.

Steve: Yeah.

Leo: Well, next week our DefCon/Black Hat coverage, maybe questions and answers. We also have high-quality audio and video of the show, if you want to watch Steve gesticulate. You can do that at TWiT.tv/sn, the new TWiT website, TWiT.tv/sn. You can also subscribe on YouTube. We have, I think it's YouTube.com/securitynow. We also, you know, it's everywhere podcasts are, so it's easy to get. Use Stitcher, Slacker, your podcast app, that kind of thing. But do get every episode. You don't want to miss anything. You know, you miss a week, and who knows, your whole system could be hacked. Just like that, boom. Thank you, Steve. Happy 10th Anniversary in two weeks.

Steve: Coming up on it, yeah.

Leo: I shall get the cupcake order in.

Steve: Okay, my friend. Thanks, Leo.

Leo: Bye-bye.