Listener Feedback #245

Description: This week, Leo and I discuss ticket-buying bots getting their hands slapped (do they have hands?); a truly nasty new addition to encrypting ransomware operation; a really dumb old problem returning to many recent Netgear routers; Yahoo being too pleased with their bug bounty program; and steganometric advertising malware that went undetected for two years. uBlock Origin readies for a big new platform. What exactly is the BitDefender BOX? We wish we knew! VeraCrypt was audited; next up OpenVPN, yay! We have the definitive answer to the question of where Spock's thumb should be; Steve's new relaxing and endless puzzler; and, finally, questions from our listeners.

High quality (64 kbps) mp3 audio file URL: http://media.GRC.com/sn/SN-590.mp3
Quarter size (16 kbps) mp3 audio file URL: http://media.GRC.com/sn/sn-590-lq.mp3

SHOW TEASE: It's time for Security Now!. Steve Gibson is here. We've got the questions we didn't answer last week. We will get to them this week. Also a nasty new ransomware that enlists you to help them and gives you a real sob story about why you should. Stay tuned. Security Now! is coming up next.

Leo Laporte: This is Security Now! with Steve Gibson, Episode 590, recorded Tuesday, December 13th, 2016: Your questions, Steve's answers - really this time - #245.

It's time for Security Now!. Yes, the question-free Security Now!. No, no, this time questions. Steve Gibson is here, our Educator in Chief, the guy who teaches us all about security and privacy and how computers work - and lives long, thank goodness, for us.

Steve Gibson: And we have the official readout on where the thumb is supposed to go. We'll be getting to that later in the podcast.

Leo: I think that should be the title of the show: Where Is the Thumb Supposed to Go?

Steve: Where is Spock's thumb? So we're going to actually get to Q&A, not because this hasn't been an interesting week. It actually has been very interesting. A bunch of fun

stuff to share. But the volume is down enough to allow us to squeak in questions. I thought, well, let's - I put seven in the Q&A because we had done - I guess we got half of them because I originally had 10. We got five of the quickies covered. Then a couple more have come up, so I've added those. I thought, okay, let's not push our luck. We'll see if we can get this done.

But we're going to discuss ticket-buying bots getting their hands slapped, if they have hands. I'm not sure. Their virtual hands. A truly nasty new addition to encrypting ransomware operation. A really dumb old problem has returned to a worrisomely large number of Netgear routers. Yahoo is being too pleased with the success of their bug bounty program. It's like, okay, I don't think you want to be bragging about how many millions of dollars you've paid out in bug bounties because, you know, those were bugs. We have, the first time I've seen this, really interesting, a working steganometric advertising malware that, due to the way it operated, was undetected for two years.

Leo: Wow.

Steve: Yes. uBlock Origin, our favorite web content control app, is getting ready for its launch on a big new platform, Edge. And the question is what exactly is BitDefender's BOX? And we wish we knew. VeraCrypt, as we covered a few months back, was audited. Next up is OpenVPN. Yay. And we have, as I mentioned before, the definitive answer to the question of where Spock's thumb should go? And I have a relaxing and really nice endless puzzler from the guys that brought us the Infinite Loop endless occupation. And I'm sure we're going to get to some questions from our listeners.

Leo: I know people just love this show and look forward to it all week long. In fact, I didn't realize this, but Rene Ritchie is quite the fan, Steve. He listens to you every week.

Steve: He and I had a great opportunity when we were both up there together two weeks ago to hang out. We spent all of the morning just drinking coffee and...

Leo: Huge fan, yeah.

Steve: And, well, we both enjoy each other's company.

Leo: Well, and by the way, Rene and Steve and Denise Howell will be the guests on our special Christmas Eve, or Christmas evening TWiT, Christmas Day. We prerecorded that so nobody has to work on Christmas Day. And it was a great show.

Steve: And it was funny. I mean, it's not funny, I mean, I'm a huge iOS user.

Leo: Right.

Steve: That's my platform. And for me it was just - it's wonderful to be able to have the

guy who knows to say, okay, now - I wasn't able to find replacement points for my pencil. And he's like, oh, well, they're over here. It's like, oh, okay. And what about this, and what about that?

Leo: I do that all the time. In fact, we'll be on iOS Today, and I'll say, well, I have to ask Rene about that. It is, it's nice to have the expert, isn't it.

Steve: So this week's Picture of the Week is an all-time classic.

Leo: Uh-oh.

Steve: In fact, I tweeted the link, and I have given it a permanent - when I say "lifetime," it's my life, and I've got hopefully a lot left. This is a static link - because this is just too wonderful. Let's see. So you were a sophomore in high school, I imagine, because in May of 1973 I was probably being fitted for my graduation gown.

Leo: Yeah, no, that's when I graduated, too, May 1973, yeah. I graduated a little early.

Steve: I guess you did, you leapfrogger.

Leo: I kind of snuck out fast. I didn't want to be around.

Steve: So that was my - I'm class of '73, high school class of '73.

Leo: Me, too. Yeah, yeah. So this is when we were getting out of high school, this was the Internet.

Steve: And of course, well, I was just going to say, nobody yet who's listening to this is like, what are they talking about? So the Picture of the Week is the Internet, well, and it wasn't even named that at the time. This was the Advanced Research Projects Agency experimental network, in other words, the ARPA Network, or Arpanet. And this is a picture of it.

And it's just wonderful because - and so we've talked about pieces of this. For example, the IMPs, the IMPs, that's the Interface Message Processor. Essentially that's the world's first TCP/IP stack. And it was the size of a refrigerator. And it had, you know, silver can transistors on three legs. And, I mean, it was old school. And then TIPs are Terminal Interface Processors which, as the name sounds, connected terminals to the network. And it's just such a beautiful diagram because there's SRI, Stanford Research, yeah, Stanford Research Institute in the upper left. We've got University of Utah, Illinois, MIT. There's BBN over on the right, what was it, Beranek something and Bolt...

Leo: Bolt, Beranek, and Newman.

Steve: Bolt, that's it, Bolt.

Leo: And Beranek just passed away, in his '90s, yeah.

Steve: And of course we talked about them because they were hogging a huge chunk of the Class A network space. They had, like, two-dot everything, or something like that. Then Harvard. There's the Aberdeen Proving Ground. So you can sort of see, oh, there's Carnegie Mellon, USC. Oh, UCSB, UCLA, UCSD, Rand Institute, Stanford University, Ames Research. Anyway, it's just so - oh, there's Xerox, of course. And so what we don't see is the kind of spider web of redundant connections that we know we have to a larger degree today. But this was all proof of concept. I mean, I think the first message went from Stanford to UCLA. I remember it was along the West Coast. And like, you know, when a character came out on the screen at the other end they were like, oh, my god, this works.

Leo: It's so funny to see. And by the way, my alma mater is not on this list. This is where I was going to go that fall. That just shows you, I mean, you and I went to college predating really the Internet.

Steve: Yeah. And I love the connection to Hawaii. It's kind of got like a squiggly line. It's like, okay. Was that radio? I don't know what that was.

Leo: Must be, from NASA Ames to Hawaii, yeah.

Steve: Yeah, maybe a satellite [crosstalk].

Leo: Hawaii only had a TIP. There are a few PDPs on here.

Steve: Oh, my goodness, yes. And that's the other thing. I'm glad you brought that up because - and that's what many people noted is that it's just covered with PDP-10s. There's an 11. There's a 1. BBN had a PDP, oh, I remember that they had a PDP-1. I think they got THE PDP-1. Harvard had one also. And then there are some 11s that came a little bit later. And a few IBM 360s, different classes of machines, 360/65, and there's a 70 around there somewhere, 370, 145. So, I mean, this is old, you know, big iron mainframe, the dawn of all of this. And anyway, just a very cool chart.

And I meant to also say that, as I have been doing, the show notes are already on the Security Now! page at GRC because there are a number of links here, especially the "is your Netgear vulnerable" test link, that I know that our listeners are going to want to be able to get immediately. So the notes are already present, public. I tweeted them, and they are on the Security Now! page so that people can see this picture that we're talking about. And the JPG of it is like a quad resolution. It's really nice. So it's just - it's a wonderful piece of work. And a whole bunch of our listeners made sure that I knew about

this. So thank you to all of them who informed me.

Okay, so I love this. This is arguably one of the most graceful acronyms or abbreviations we've run across. Some of them seem really painful. Somehow the military often gets really good ones. But the NSA's, of course, are just abominable, well, because we think it's just a random word chooser that assigns names to their various projects. This one, though, this jumped on my radar last week because Congress passed an interesting law called the BOTS Act. And the reason I love the abbreviation is that it is a BOTS Act. And the abbreviation really works well. It's the Better Online Ticket Sales, BOTS, Better Online Ticket Sales Act of 2016.

So I'm just going to read from the text of the bill. Section 2 reads: "This bill prohibits the circumvention of a security measure, access control system, or other technological measure on an Internet website or online service of a ticket issuer that is used to enforce posted event ticket purchasing limits or to maintain the integrity of posted online ticket purchasing order rules for a public event with an attendance capacity exceeding 200 persons. The bill also prohibits the sale of or offers to sell an event ticket in interstate commerce obtained through such a circumvention violation, if the seller participated in, had the ability to control, or should have known about the violation.

"It shall not be unlawful, however" - and thank goodness for this - "to create or use software or systems to, one, investigate or further the enforcement or defense of alleged violations; or, two, identify and analyze flaws and vulnerabilities of security measures to advance the state of knowledge in the field of computer system security" - this is a really good omen - "or to assist in the development of computer security products." And then two last sentences: "Violations shall be treated as unfair or deceptive acts or practices under the Federal Trade Commission Act." And that's what caused me to wonder if it isn't just a hand slap. But then it says: "The bill provides authority to the Federal Trade Commission (FTC) and states to enforce against such violations." So, yay.

Okay, so many people probably know that there has been a systemic and pervasive abuse of online ticket selling because famously Ticketmaster and others would put tickets online. A swarm of automation would descend on those, purchasing the tickets, and then typically marking them up, depending upon popularity, at least 50%, and oftentimes much more. And so the problem has been that ticket control pricing was - essentially that this was automated third-party ticket scalping, which had just become a sub-industry. And people decided no, you know. And the point is that there was active measure to circumvent this. So the initial response was one of using technology, as we often see. It's like, okay, something like the - I want to say "chapkits," but that's not it.

Leo: I don't know where you're going with that.

Steve: You know, the chap - it's not chapka. It's chap - I haven't said the word for a long time. You know, the things that you prove that you're not a bot, the...

Leo: Oh, CAPTCHA.

Steve: CAPTCHA. I was starting with the wrong syllable, thank you. CAPTCHA.

Leo: I was really - I apparently am a bot. Couldn't do it.

Steve: So, you know, all the CAPTCHAs. And in fact I've been meaning to dig into the technology of the newer ones we're seeing because I'm seeing some now...

Leo: They're terrible. I hate them.

Steve: Well, no, but there are some now that are just a checkbox.

Leo: Yeah.

Steve: Where it's like, just...

Leo: I am not a bot; right.

Steve: I am not a bot. And I'm thinking, okay.

Leo: But sometimes you check that, and it generates a CAPTCHA. I don't know why.

Steve: Okay. Maybe it's using a reputation-based system based on IP or something.

Leo: Something's going on, yeah.

Steve: Yeah. Because I'm seeing that now, and I go, this is not - does not sound like really robust [crosstalk].

Leo: Doesn't seem too hard.

Steve: Although it sure bits staring at some strange, like, barely legible street address photographed from the side in dim light on a full moon. It's like, oh, is that a seven? Because of course we've discussed this at length. This is like the problem is bots are getting pretty good. And in fact many of these CAPTCHAs, as we've also discussed, use basically sweatshops in...

Leo: Humans.

Steve: ...in foreign lands to solve them on the fly and then, you know, for circumvention. So the point is the first attempt to thwart this kind of abuse was to do bot protection. And then of course they got circumvented. And so I think this is the right

thing. It's that this is not meant for bots. And it's been hugely abused and raised enough pain that finally Congress said, okay, legislation.

And however, as I said, the silver lining here, I was so glad to see an explicit exclusion for research. It's like, oh, please let this be the trend. Frame this paragraph and drop this into all subsequent similar legislation because I do recognize that law enforcement needs to have something to use, that is, it needs to be made illegal so that law enforcement can knock on the companies that are doing this and say, you know, we're sorry, but you need to go find something else to do because you've had a nice run, but it's over because you are doing something that is expressly forbidden by the terms of service of the seller from whom you are purchasing tickets. And they can't prevent you from doing it; but we can, now that we have a law. So, you know, the BOTS Act, Better Online Ticket Sales. This is perfect.

Okay, now, this is almost stomach-turning. There is a new twist in the file-encrypting malware environment. This one, the first one to emerge, is Popcorn Time. And remember when we first saw this a few years ago, and it was immediately clear, oh, we're going to have more of this because, if this is making money, you know, the files people care about are now no longer available to them. But if they pay up, they can get their files back.

So, yes, as expected, there's been an explosion of file-encrypting malware, unfortunately. Well, now there's a new twist. This one, the first one of these to do this is called Popcorn Time. And when the warning message comes up, you have not one thing you can do, but a choice. The first thing you can do is the standard, which is pay one bitcoin, which today is $780. So that's, you know, it's been creeping upwards. It was 450 for a long time. Now it's about 780. So that's Choice A. And get a load of Choice B. If you want your files back, you also receive a custom infection link...

Leo: Oh, no.

Steve: ...from these people. And if you arrange, using your custom infection link, to get at least two other people infected who pay...

Leo: Oh, that's so evil.

Steve: ...you then get your files back.

Leo: Oh, like multilevel virus marketing.

Steve: Yes.

Leo: Oh, that is nasty.

Steve: it is just - it's brilliantly, horribly awful.

Leo: Nasty.

Steve: Nasty. It really is. So think about what this does. I mean, think about what it means. First of all, $780 current bitcoin, that's expensive. And the other problem with the whole bitcoin thing is it's still not easy. I mean, even if you had the will to pay someone a bitcoin, you know, my mom doesn't know how to do that. I mean, it's...

Leo: I know.

Steve: It's not an easy thing to do.

Leo: You have to have some to begin with.

Steve: I mean, right.

Leo: Or just go buy it, yeah.

Steve: Right. So there's a significant conceptual and performance hurdle just associated with arranging payment, if you wanted to do that. So that creates additional pressure, aside from the $780. But now you've got an alternative. And it's awful. I mean, wow. Now, okay, now. As if these guys were not cretinous enough - and Leo, I put a link to the full-size warning message because there's a lot of fine print in it, and the show notes are very - make it difficult to read. But as if these guys were not cretinous enough, they then in this message proceed to claim that they are Syrians, and that the proceeds from this extortion, quote, "will be used for food, medicine, and shelter."

Leo: Oh, you're helping refugees.

Steve: Those in need.

Leo: Not nice.

Steve: Then, quote, "We are extremely sorry" - uh-huh - "that we are forcing you to pay, but that's the only way that we can keep living." And then I said: "And in case you still thought they might be good guys, reverse-engineering of the code has appeared to indicate that four wrong guesses, four mis-entries of the key will trigger permanent deletion of the still-encrypted files."

Leo: Wow.

Steve: So, yikes. And unfortunately, I mean, and they even say in their message:

"Restore your files the fast and easy way." Unfortunately, it's not that fast. Bitcoin, as we were saying, isn't. And then the alternative is, and they say this, "Restoring your files the nasty way."

Leo: At least they know it's nasty.

Steve: Yes, send the link below to other people. If two or more people install this file and pay, we will decrypt your files for free. Wow.

Leo: That is just - wow.

Steve: And unfortunately, it has a creepy feeling of success in the same way that, when we first encountered the concept of file-encrypting ransomware, it was like, oh, this is going to be bad. As you said, the multilevel marketing and spreading, I mean, because what will happen is people - presumably you're not going to do this to someone you like.

Leo: Well, that's true. It actually could be weaponized for people you don't like. That's a good point.

Steve: Well, but so there are people you don't like, but who you know enough about. And so this is always the challenge, of course, with social engineering is that it's a scattershot. Well, this is now a sniper rifle because you know who you don't like and who you want to help you get halfway off the hook. And so you can use your knowledge of this person to incent them to click this link, to synthesize an email which will be especially relevant to them. It's just, you know, so it's like a social engineering, multilevel phishing, file-encrypting malware. It's just incredible. So, and again I have to say diabolically clever because, oh.

Leo: Yeah. I'm sure people will send it along. I'm sure it happens.

Steve: So everybody be careful because, yikes.

Leo: Oy.

Steve: Now, our next up is the discussion of this very sobering Netgear router problem. It was discovered by a hacker whose handle is Aceworm - I didn't dig into his background any further, I thought that was enough - who initially reported two Netgear router versions. Netgear has responded. When I looked late last night they had updated firmware in beta for three of their router models. So, and unfortunately these are like all the good ones. The R6250 is known to be vulnerable. The 6400, that's the AC1750. The R6700, the Nighthawk, very popular; the R7000, which is both the AC1900 and AC2300. The R7100LG, R7300, R7500. That's the Nighthawk X4. The R7800 Nighthawk, the X4S; R7900, R8000, R8500, that's the Nighthawk X8; the 9000, and the Nighthawk X10. So a broad swath of recent Netgear firmware, which of course demonstrates that it's a common codebase.

Here's the problem. And the reason I consider this a dumb blast from the past is around the turn of the century - and isn't it odd that we can use that phrase now, and it actually means some time ago. Around the turn of the century this is the kind of dumb stuff that was still happening. That is, we were so nave back then that it didn't occur to us that your web interface could be used to execute things in the cgi-bin directory. Well, of course that was fun for a while, and then it all got locked down. Well, and it's back. And that's the point.

Now, the good news is this is a local exploit. So this is not like other problems that we've discussed where, for example, there would be an undocumented backdoor exposed on the WAN interface which would allow people aware of it or who had discovered it to just sweep the Internet and sweep up all of the devices that were vulnerable. So this is only on the LAN side. The problem is that it's, just as we were discussing, it's not that difficult to get someone to click on a link. And when you click on a link in your LAN, you are on the LAN side.

So the exploit - and what's sobering about this is it is so simple. It is http:// the gateway IP of your LAN. And so that's going to be 192.168.0 dot something. A Netgear - I hadn't thought of this until just now, but generally manufacturers all default to the same 192.168. It's like .0.1. I think Netgear tends to be dot one dot, like maybe 1.0 or 1.1 or 1.254, depending upon which end of the block of DHCP 1913 RFC addresses they go in. Anyway, so it does need, you do need to be making a web query to your router's browser interface. And so it's the IP/cgi-bin/; and then whatever you want. Like a command.

So, and the problem is, for example, there are now proof-of-concepts of this. There is a command that will start up the Telnet service and open it on a WAN-facing port, which then of course makes the router instantly vulnerable to remote access, essentially, console commands on that port. CERT immediately issued an advisory stating that multiple Netgear routers are vulnerable to arbitrary command injection. And they describe the problem as "improper neutralization of special elements used in a command."

Now, the good news is - okay. So first of all, you can easily check to see if you're vulnerable by using the command Reboot. So it would be http:// the IP address of your router, probably 192.168.1. whatever. And of course that's your gateway IP. In Windows you could open a command window and say Ipconfig. I think that's what it is. It's been a while. And Microsoft, the exact command has drifted around a little bit from time to time.

But anyway, whatever the IP is /cgi-bin/;reboot. And if your router reboots, you're vulnerable. You can also instead use the command cgi-bin/;uname$IFS, which essentially is a space, and then a hyphen lowercase "a." And if your browser shows you a whole bunch of stuff, you've just executed that command. And because it's a cgi command, the output from the command is returned in the response to the browser's query, and you'll see a whole page of text of stuff from your router. So again you know you're vulnerable.

Now, there is a temporary workaround, if you find you're vulnerable; and that is, you can shut down the web interface essentially to, while you get your firmware updated, to prevent your router from being accessible through the web interface. So, for example, you'd have to do a local telnet or reboot the router or whatever. So I don't know if you can disable through the UI the web interface. But you can kill the process. So same command again, cgi blah blah blah, cgi-bin/ - and all this is in the show notes, by the way - /;killall, and then the little string that expands to a space, which is the $IFS in caps and then a single quote, httpd, and then close the single quote. So you're telling - that's a command telling the firmware in your router to kill any processes with the name httpd, which is the http daemon, the service running in your router. So that immediately

protects you, although when the router reboots it will start up again. And it's possible that the router could notice that the httpd daemon has died and then relaunch it itself. So it's sort of a sort workaround.

But for those who are concerned, there is firmware on the way. I don't know how quickly Netgear will respond. They already did have beta firmware that they were testing. The list of vulnerable routers that I just presented is the most comprehensive one I found because I assembled it myself from several different sources of people that had since been verifying additional, you know, using crowdsourcing in order to get people to test their routers using these various approaches and then logging their responses. So it looks like many Netgear routers are in trouble. So if you are a Netgear user with one of these late model nice routers, check in with Netgear and/or use your router itself to check for additional firmware.

It may be that, while they're in beta - well, I'm sure that's while they're in beta they're not officially released and probably not available for automatic or automated install online. So you'd probably have to grab them yourself. And it's, again, not a remotely exploitable, not a huge problem, but Netgear's on it. They've acknowledged the problem. And they had, for the R6400, the R7000, and the R8000, they had beta firmware for those when I last looked. And so you can check to see. And again, the link to that page at Netgear is in the notes, too.

Leo: Would be kind of - is it remotely exploitable if you click a link that launches the cgi script?

Steve: Correct. So what would happen is essentially bad guys - and they're already getting very creative. Bad guys can cause one or more commands to be executed in the context of the web service which runs with root privilege. So you're executing root privilege commands to the firmware. And so one of the things it could do is instruct it to fire up a Telnet service on one of the public-facing 64,000 ports, so whatever port the bad guy wants to use. Then Telnet is just sitting there waiting for connections. And the bad guy could be then scanning the 'Net for those ports which are open and accepting TCP connections, supporting Telnet, and then of course they have remote access to your router.

Leo: Okay. So they don't have to have physical access to your house.

Steve: No, no. They, you know, you click on a link in social media, you click on a phishing link, and this thing then executes the command and alters your router in a way that's useful to the bad guys.

Leo: Wow.

Steve: Yeah. Not good. And, you know, I've got a Netgear router, right up there. It's got a whole bunch of little antennas all over it.

Leo: Yeah, yeah.

Steve: So Yahoo, I call this "Yahoo's too-successful bug bounty program" because I was reading something from a Yahoo spokesperson bragging about this. And I thought, you know, honey, I don't think this is really what you want to be bragging about. So okay. The beginning of this was, speaking of classic bugs that had long been fixed, Yahoo just paid $10,000 to a Finnish security researcher who found and responsibly reported a means for anyone to access anyone else's Yahoo mailbox, simply by sending them a specially crafted email.

This was a cross-site scripting bug. The security researcher discovered that an attacker could sneak malicious JavaScript code past Yahoo's mail filters by abusing the way Yahoo Mail displays links to sites such as YouTube. All that was necessary was to embed JavaScript within a specially crafted email containing a YouTube video link. No action was required on the recipient's end. Nothing to click, just viewing the email would execute this embedded JavaScript in the trusted web context of the user, thus the cross-site scripting problem, that would then allow this JavaScript code to execute exactly as if it had come from, as if its origin was Yahoo, which meant that it had full access to all of the Yahoo-specific resources like session cookies and so forth and could do whatever it might want to.

So anyway, as I was digging into this, I ran across this comment, that a Yahoo spokesperson said that the company "has developed one of the largest and most successful bug bounty programs in the industry. 'We've paid out more than $2 million in bounties, resolved more than 3,000 security bugs, and maintain a "hackership" of more than 2,000 researchers, some of whom make careers out of it.'" And I'm thinking, okay, wait a minute. You have so many bugs at Yahoo that people's entire career is just cashing in on finding your bugs? Well, okay. Wonder what's wrong with this picture?

So, I mean, I guess it's better than them still having 3,000 bugs that hadn't been found, and it's their money. And, I mean, it's great that people are being responsible and reporting these. But still, you know, I'm not sure you want to brag about what a golden trove of treasure your service is, being that it's so bug-ridden that it's like, yeah, it's not that hard to find a bug. You can make some money.

This is just really, really, really, really clever. And the takeaway from this one is, boy, you know, where there's a will. We've talked about steganometry before, and it's never really impressed me that much. I mean, certainly it's probably a real thing. So steganometry is this clever way of hiding something, essentially hiding a secret in plain sight. And that can be done in many ways. For example, you could embed a very soft whisper in an audio file, for example. Technically that's steganometry. And if you then took the original audio file and subtracted it from the altered one, you would be just left with the whisper, which could be the secret message. Or what is the more common approach is, for example, to use the unused super-high resolution of images to bury another image or a digital message.

So, for example, we've got 24-bit color where we have red, green, and blue gets a byte of resolution each. But so you have 256 different intensities of red, from none, that is, black to bright red. The same thing for blue and the same for green. Well, you can't see the least significant bit of those. So for every pixel, every colored pixel you could steal the three least significant bits, one each from the R, the G, and the B byte, for your message. When we look at it, we're just going to see the picture. We're not going to notice that there's any specific message.

Now, maybe if the secret was like vertical stripes or something, like that was exactly lined up with the scan, the raster scan of the image, then maybe you could detect a little flutter in the least significant bit. But it's probably going to look like pseudorandom noise.

It's going to be code, or it's going to be something. So you just won't detect it at all. But the point is that somebody who knows that a certain picture posted somewhere on the Internet, looking completely innocuous, if they know that has got their message in it, then they take it, and they strip out all but the least significant bits of each R, G, and B, and they've got their message. And it wasn't - and then, I mean, and it could also be encrypted, then they decrypt it and so forth. Or it just might be in the clear.

Anyway, that, steganometry, was used for two years before ESET found it in the wild in ads. And not Spy vs. Spy, people using it to send secrets to each other, but a very clever means of sneaking malicious JavaScript past all ad networks' filters. Because they're looking at the code in the thing you embed, not at the image. And so what they did was they had innocuous code that looked just fine. Nothing wrong here, I mean, kind of screwy, maybe; but why does it care about the images? Well, what it's doing is it's processing the image, extracting the malicious content steganometrically from the image, and then executing that.

So it's just brilliant. Again, it's just like, okay, we might as well just give up, if this kind of stuff - the problem is we have - and this is a perfect example of just so much rich technology that can be used in an inherently multipurpose fashion, that there are different ways to remix the same components and get a different outcome. And, wow. So congrats to ESET. Somebody must have just asked themselves the question, okay, what is this advertising JavaScript doing? And so they reverse-engineered it and thought, and why is it processing the images like this? And they dug into it, and it's like, oh. That's why. And this thing had been going since 2014.

Leo: That's amazing. Wow.

Steve: Yeah. Just so cool.

Leo: You're using the term "steganometry." But I've also heard "steganography."

Steve: Yeah.

Leo: Same thing?

Steve: Steganometry, steganography, yes. Well, steganography probably...

Leo: Is the creation of those images, maybe steganometry the reading of them.

Steve: And maybe "graphy" refers to image, whereas steganometry is more buried content. So it may not...

Leo: Yeah, the "steg" means hidden; right?

Steve: Correct, yeah.

Leo: Okay.

Steve: And in fact it was called the "Steganos exploit kit."

Leo: Right.

Steve: Was the way that they named it when they found it. And if you needed any, I mean, we know we don't need any other reasons for keeping really annoying visuals off of our browser. But from time to time I am using a browser that has just - is wide open. And I'm just...

Leo: I know, it's terrible.

Steve: It's just - it's awful. It's like, how did - and especially now, these autoplaying videos? Oh, goodness. It's like, no, no, no, come on. And of course they're using your bandwidth for their own purpose. And so...

Leo: Everybody's started doing autoplay videos. It's driving me nuts. I just hate it.

Steve: Yeah. It's really wrong.

Leo: Unfortunately, you know, you can turn off Flash, or say Flash has to be loaded. You know, you ask before running. But now they're using HTML5 video, and that loads no matter what you do.

Steve: uBlock Origin.

Leo: Love it. Still use it.

Steve: Yes.

Leo: All the time.

Steve: Absolutely. It is on my browsers. It's what I recommend. Occasionally somebody will complain, that is, a site will complain. I go, yeah, okay, fine, you know.

Leo: More and more, yeah.

Steve: Yeah. It has never been available for Microsoft's newest Edge browser. It is now

in final preview. Microsoft has it available, and I've got links in the show notes. It's on the Microsoft store, uBlock Origin preview, and should be - they're just nailing it down. In digging around a little bit more, I was interested to see that, while it did need some customization, 95% of its codebase is unchanged and shared by Firefox and Chrome.

So this is really nice. We're seeing a consolidation of plug-in backroom or backend technology, which is really, I think, a good sign. We went through a phase where everybody had their own solution and approach. And we're now seeing that pull together as people recognize, as Microsoft did with Edge, they could have certainly rolled their own from scratch. They did roll their own browser from scratch, thankfully. It was time to give IE its final dirt nap.

And now, rather than just going their own way, they clearly made an effort to have their plug-in API as Chrome and Firefox, that is to say, industry compatible as possible. And so this is the kind of benefit that we will reap is that they will end up with plug-ins they would not have otherwise had, if there just wasn't enough incentive enough or time for a developer who was going to first support the dominant browser in the industry, which is now Chrome, to go and do a whole complete different implementation for Edge. So uBlock Origin coming soon to the browser you may be using, Microsoft Edge.

And we've spoken many times now about Moxie Marlinspike and his work, the Signal protocol. The New York Times - I thought, okay, when this surfaces to The New York Times, you know you're getting real traction. They had a weird picture of Moxie, though. They arranged to hide his hair, which as we know is sort of strange. But he was, like, tucked into this corner of a room that was otherwise apparently empty, with windows on either side of him, looking kind of forlorn.

Leo: Aw.

Steve: And I thought, Moxie, what's wrong? You know, The New York Times is talking about Signal. So you should be smiling more. If you click, the link in the show notes, Leo, you'll see The New York Times story and the picture of Moxie.

Anyway, so here's what The New York Times had to say, which I thought is like, wow, this is going mainstream. They said: "By the time you finish reading this column, you would be foolish not to download the messaging app Signal onto your smartphone and computer." Okay, this is the paper of record saying this.

Leo: Yeah, but it's also Brian X. Chen writing it, who's pretty savvy. Really savvy.

Steve: Yes.

Leo: Yeah, he's good.

Steve: Oh, yeah, yeah. "The free encrypted messaging service has won the acclaim of security researchers and privacy advocates, including Edward Snowden. All have said that Signal goes above and beyond other chat tools in keeping electronic communications private." And of course we've given it a podcast, and I said the same thing. It's like, wow, you know, this is the difference between doing everything you can think of to -

well, and remember when I explained Signal, I was initially puzzled by some of the things they had done because it looked like a ridiculous level of over-engineering until I really sat down to learn it in order to explain it to our listeners, and it all kind of came together. And I ended up with a great deal of respect for it.

And so their coverage continues, saying: "Signal is one of many encrypted messaging services, but it stands out for its uncompromising security and ease of use. The chat service retains virtually no information from users, including messages and address books, on its servers. What's more, messages remain encrypted when passing through Signal's servers, meaning that the app's creators cannot read them."

And so that's their coverage. And then I had added a note here to myself to remind myself to say that we know that Facebook's WhatsApp also adopted the Signal protocol. But I think sophisticated users are encouraged to use Signal, and to definitely activate - remember that always for these things you don't actually have security unless you have authentication. You have to know who the other person is and that that not be spoofable, that their identity is authenticated through some mechanism. Otherwise nothing prevents a third party from being a shim, a man in the middle, and essentially inserting themselves into your conversation. WhatsApp has the problem that Facebook has now officially said that they're going to do an information sharing between what the WhatsApp side of their properties and Facebook. And there's a commercial interest.

So I guess the way I think this falls is there's a lot of Facebook users. It would be better to use WhatsApp than nothing. But if you really - and so, you know, why not? It's there. It's supported. But remember that, for a reason I don't understand, the option to notify if the fingerprint of the identity of the other party changes, that's not on by default. You absolutely want that on. And the same thing goes for Signal. You need to just once arrange to verify signatures. And you can just do it through some out-of-band exchange, either on the phone or email or just tweet. Just send some of the characters of the signature, and it's the output of a hash. So it'll be impossible to deterministically spoof that. Or, okay, I just used a forbidden word. Not impossible. Computationally infeasible.

Leo: Challenged.

Steve: Yes. So anyway, I just think it's great that we're seeing this kind of coverage. And, boy, Signal is the one to use if you really do want to have yourself be secured.

Leo: And I'm using it. Now, the only thing I didn't like about Signal was that it uses my phone number for the account.

Steve: Ah, right.

Leo: So it does leak a little information. I mean, you've got to do something, I guess. Threema doesn't because you have to meet and exchange information. But that's not practical. The other thing I like about Signal is there is a Chrome extension for it, so you can use it on your desktop computer.

Steve: Oh, nice. That is one of the things that is chafing for me as an iOS user, I was mentioning earlier, is that Apple doesn't want me to be communicating with my iOS

things over on Windows.

Leo: Yeah. So Android lets you use Signal as your main SMS app. You can designate it.

Steve: Nice.

Leo: But obviously that's - the limitation of iPhone is, you know, you can't make anything but Safari your browser. You can't make anything but Messages your messaging app.

Steve: Right.

Leo: I mean, Apple does encryption on Messages; right?

Steve: Yeah, yeah, yeah. They absolutely do. But they also manage the keys for us.

Leo: Yeah.

Steve: So there's the convenience of it just works. Yes, it's encrypted. But if the FBI said, "We really need to be monitoring this communication; slip our key into the group," then the user would have no idea. I mean...

Leo: Well, and law enforcement could subpoena it because they have the key.

Steve: Correct. And it was funny. Charlie Rose interviewed Bill Gates a couple weeks ago. And Charlie was really interested in AI, wanting to pick Bill's brain about AI, and also this whole issue of encryption of our communications. And Bill didn't go into any detail because, you know, it was Charlie Rose. Just, you know, that audience doesn't want details. But Bill was adamant in saying Apple can get your content. Apple can see your data. Apple can do this if they want.

And I appreciated just the simplicity of that communication. He stated, first of all, it was Bill Gates who you're going to tend to think, okay, he probably knows what he's talking about. And so it's not easy to kibitz that. And he's, like, saying, no, Apple can. Let's just, like, take that question off. And it was funny because Charlie was confused because he was like, well, wait.

Leo: No, Apple implies, oh, no, it's...

Steve: It's encrypted. It's encrypted.

Leo: Yeah, they don't - they really - they do a lot of hand-waving around that.

Steve: Well, and of course I've also watched him interview Cook, who's like, oh, no, everything's encrypted. We can't see any of it. So it's like, okay. And of course we know Bill's right. Yes. Apple ultimately controls the software in your phone, end of story. They can give you anything they want. They know who you are.

Leo: Right.

Steve: You know, you individually, uniquely. So, yeah.

Leo: And of course I think any, you know, January 20th on, a newly empowered FBI will be knocking on Apple's door, I'm sure.

Steve: Well, and I actually skipped that in The New York Times article. There was a paragraph about the incoming...

Leo: Well, that's why Signal's downloads are up 400%. It was right after the election. I mean, you know, that's a little alarmist to say suddenly we've got storm troopers.

Steve: Precisely. And it wasn't relevant to the technology.

Leo: Yeah. But it's just prudent to do it, and do it now. And then you don't have to worry about it.

Steve: So arguably one of the stronger antivirus systems, BitDefender, has been around for years. It ranks very highly in various third-party cross-AV analysis of, like, false negatives, false positives, how many signatures are found, and so forth. Many people this week shot me a link to a new product, a hardware product that the BitDefender folks have produced, called BOX because that's what it is. It's a box. So it's BitDefender.com/box. And it's very pretty, and it glows. There's sort of an apex on the lower edge and a nicely sort of cyan light can be seen. And apparently that can blink red if it's not happy and change colors and do things.

So, and I was curious. It's like, okay, what is this? Because here's the problem. And we've discussed this before, partly in the context of Cujo, which was another, an earlier box which did not impress me for two reasons, our listeners will remember. One was that it was doing a deliberate ARP attack on the person's LAN in order to hijack the interface associated with the gateway IP, to solve the problem of how it could be able to intercept your communications. So that just was clunky.

I mean, I understand if you decide you want to solve the problem, and the architecture of the system won't let you physically connect Cujo inline. For example, if you don't have a separate router and a cable modem or a DSL modem where there's a wire connecting

them, that allows you to put Cujo in the middle then this is a way to solve that problem. But the bigger problem is everything's encrypted more and more. And so something like a Cujo sitting there saying we completely protect your system, and we're going to scan you for viruses, and it's like, no, you're not. You can't. You can't see into the traffic anymore.

And so when BitDefender did this, I thought, okay. Do we have the same problem here? So I read through their 46-page owner's guide, and it looks like this is the real deal. However, it comes at a cost. Sort of icky, but if this is what you want. So I'm not sure this is a solution for us all, but for many people. They solved the going dark problem because essentially they've got the same problem the FBI has. But this time they're on your side, definitely looking for anything, wanting to intercept anything that they believe would be malicious, to do AV stuff.

Now, I'm assuming they are doing traffic filtering. And the big problem is how would we know? There is no technical explanation anywhere. It's just look how pretty the box is. Look at the rounded corners. Buy it. I think it's normally 199. At the moment it's discounted to 129. But of course it's also got the annual subscription service fee. Now, for that you get a lot. But the problem is we don't know exactly what you get because it's all just nice-looking glossy surface, both on the box itself and on the website.

So it's impossible for me to be as definitive as I would love to be because, very much as is the case still with BitTorrent Sync, they won't tell us. They won't share what they're doing. So it's like, well, it might be secure. The point is the icky part, kind of, but the tradeoff, is they require you to run agents in every device in your house. So they've got Windows, Mac, iOS, and Android. I don't know if they have a Linux agent, but of course a lot of our listeners are running Linux that would like that protected. They don't have a TiVo agent nor a light bulb agent. So there's the problem of coverage if their solution depends upon something on the endpoint helping them to get around encryption.

And in fact that is necessary, that is, if your light bulb or your TiVo is going to have an encrypted connection, there's nothing they can possibly do to intercept it. I mean, well, except synthesize a certificate which would be signed by an already trusted certificate authority. And no certificate authority of any repute would ever give them a certificate that had resigning, you know, certificate signing privileges itself. So we've covered stories of where other commercial devices have been found doing this; and then the people who signed their cert for them said, oh, well, that was just meant to be contained in the lab. That was never supposed to be out in the wild. It's like, uh-huh, okay, well, so.

So here's the problem. BitDefender I'm sure wants to do a good job. They want to make money. They're telling you that this solves all the problems that you have. And it's a compelling - it's like, wow, you know, I just stick this here in my network, and now I'm secure. All of my stuff is secure. Except, okay, it's not all, and we just don't know what they're actually doing. So I got so many people said, hey, you know, what is this, I wanted to address it. And unfortunately, I have to say they're not telling us. I'd be happy to know.

But there are problems with doing both what Cujo, I mean, aside from the Cujo's approach, which was the ARP hack, there are problems with any solution which purports to just be a drop-in box on your network that's going to secure you because it has the same problem that law enforcement has out on the Internet, which is it cannot see into the traffic to an ever-increasing degree. And maybe these endpoint agents give them a tap prior to encryption, which would then allow them to see it. It's just not possible to know. And they could be doing DNS and looking for malicious websites, I mean, there

are many good things such a box could do. But again, it's just trust us, plug it in, and now you're secure, and make sure you don't miss one of your annual payments. So we'll see.

The people who brought us the VeraCrypt audit a few months ago shot me a tweet three days ago, actually to you, Leo, and me, and Security Now!. You know, Security Now! does have a twitter account. Somebody got it.

Leo: It does?

Steve: Yeah. Somebody grabbed it in the very early days. And I want to say they offered it to me.

Leo: Oh.

Steve: I don't remember what the deal was.

Leo: They probably did.

Steve: But anyway, so this is the OSTIF...

Leo: [Crosstalk] "This account name's been reserved for Steve Gibson on GRC.com."

Steve: Yeah, okay. So if anybody here...

Leo: No, no.

Steve: I don't know how to - who are you?

Leo: Yeah. Huh.

Steve: Yeah. Anyway, it's receiving tweets from people. But I'm not seeing them.

Leo: Yeah. Don't tweet to that one.

Steve: So this is the OSTIF. The official account said to us: "You guys did a great piece on our VeraCrypt audit. We are at it again with OpenVPN." To which I say yay because it's OpenVPN. It's like it's the one that has won. So the effort was launched by OSTIF exactly three weeks ago. It was Tuesday, November 22nd. OpenVPN is currently at v2.3, with 2.4 in beta. So it's 2.4 that is the focus. We haven't had a version change in OpenVPN in quite a while. It's got tons of fixes and feature changes and improvements.

So it's going to be the first major release in years.

These guys hope to raise $71,000 to fund an audit of OpenVPN, much as they did fund the audit for VeraCrypt. iPredator has contributed 10 bitcoins, $7,700; OpenVPN Technologies, $5,000; Perfect Privacy, $3,500; nVpn.net, $2,650; ExpressVPN. And so these are providers; right? ExpressVPN, $2,500; SmartDNSProxy, $2,500; iVPN, $2,100; SecureVPN.to, $1,500; VPN.ac, $1,500; and so forth. And then it goes into a whole bunch of others. So they're about halfway to their goal. And so first of all I wanted to bring it to everyone's attention. They are looking for contributions. So it's OSTIF.org is the outfit.

Now, then something odd happened that they're not happy about. But as long as it doesn't derail this effort, I am happy about it. And that is that Private Internet Access - which is another major VPN endpoint provider. Private Internet Access apparently, after knowing of this OSTIF effort and being contacted by them and solicited for participation, you know, to be named among the supporters of the audit and so forth, they just decided to go their own way. They've hired Matthew Green to independently and separately audit OpenVPN, this forthcoming v2.4.

Leo: Well, that's just him.

Steve: Well, exactly. So, I mean, so this is not bad.

Leo: He's the guy who did the TrueCrypt; right?

Steve: Well, yeah. Yeah, Matthew is the...

Leo: Is the guy.

Steve: Yeah, the cryptographer. He's spoken in front of Congress, and he's at Johns Hopkins. He's in the middle of all this. And so, as long as this independent effort doesn't prevent OSTIF from succeeding with their fundraising goal, then I think the more eyeballs that are focused on this next version of OpenVPN, the better. And this is not like someone saying, okay, we need to absolutely verify once and for all that two plus two equals four. Okay. I can do that. You know, we don't need a group. We don't need a committee or a team or a big effort.

But as we know, unfortunately, the way software exists today with its complexity and features and so forth, especially as it gets older, and it's bringing a whole bunch of legacy stuff along that no one's willing to let go of, which is the case with OpenVPN, although it's a good solid offering, you know, those things are more like, okay, we need the exact count of stars in the Milky Way. Okay, let's assemble a team. Because that's not something one person can arguably answer.

So this does require - nailing down software requires more effort. And I would imagine different things will be found. And actually, if it was ever possible to compare the - if there was no leakage between the two audits, be really interesting to compare what they both independently found, how much overlap and how much non-overlap in the Venn diagram of the two audits is there.

So I wanted to let everyone know that, first of all, the good news is that we are apparently definitely going to get some good audits of OpenVPN: one privately financed by a provider using Matthew Green; and the other is the OSTIF, who brought us the VeraCrypt audit, going to do the same thing, bring us the OpenVPN audit. And if any of our listeners are interested in contributing anything to that effort, it's OSTIF.org.

Leo: It's kind of a high-risk proposition, though, because if you miss something big, that looks kind of bad; right?

Steve: Yeah.

Leo: There'll be overlap, but I'm sure there'll also be unique findings on each person's part.

Steve: There'd have to be, yeah.

Leo: Because you're not going to find everything.

Steve: I mean, because it is a big - well, and also, too, it is absolutely the case just that there is a matter of opinion involved. Like someone could say, you know, I've never felt comfortable casting 16-bit values to 32 because on some platforms where you might have a compiler that does this, blah blah blah. So it's like, so even code has gray areas which are about style and taste rather than - and feedback from longstanding real-world experience, where you just go, oh, you know. That can kind of come back and bite you in the ass, you know, sometimes.

So, yeah. If you're in a situation where you've got big gray areas, you're going to have different outcomes. And which is why, actually, it'd be great for them both to do their thing. And then, of course, share their, you know, cross-share their results so that we end up with the - I'm sure we will end up. Certainly the OpenVPN project is the ultimate arbiter. It will receive the results of both audits and automatically amalgamate them into what finally becomes 2.4. So it'll be fascinating as we watch this evolve to see how it goes. And I think probably, you know, as we're seeing, these things are not taking forever to happen. The TrueCrypt audit was only a couple months. And VeraCrypt, same thing. So, neat.

A little bit of miscellany. I have a new puzzle, Leo. The good news is I like them when they're free, and I like them when they're cross-platform. So this is for both iOS and Android. And I really enjoy this. This is, you know, as I've often talked about, the requirements for me for puzzles, I don't want to have to, like, you know, a reflex test where something flashes by, and I have to nail it with my virtual blaster. I want to have something that's more contemplative. This is called Square it!. And much as they - yup, there it is on the screen. And much as they used an infinity sign as the first character of Infinite Loop, these guys have used the unicode block character as the first character of the name Square it!.

So it's called Square it!. It is free, although it is ad supported. After you finish a level, you get an inter-level ad. And if that's annoying, $0.99 and that disappears. So mostly

because I like these guys' work, and I want to support them, I immediately shot a dollar, I mean, it's a dollar, shot a dollar at them so that I would not be facing those ads. And I really like it.

It's difficult to explain. It's unlike anything that you've seen before. Very minimal, well, and Infinite Loop was this way, too. Remember, Infinite Loop was you got a grid of squiggly shapes, for lack of a better term, which had been rotated arbitrarily. And when you tapped on them they rotated 90 degrees per tap. And so you just had to figure out, had to essentially reassemble this fragmented sort of jigsaw by rotating pieces until all the pieces connect to each other with no frays, no loose ends. So that was completely novel.

This is, too. You just push up, left, right, or down; and you move one or two cursors around. And the goal is to fill all of the area. Very simple. No instruction manual. The first few puzzles show you everything you need to know. And it's algorithmically generated. I'm on, I think, on the pad I use most for reading - I'll pause reading and do a few levels and go back to reading - I'm like on Level 94 or something. And it's just nice. None of them are super hard. Some of them are a little trickier. But they also go forever. So I just - it's Square it! for iOS and Android. And it gets the full recommendation. No timers, no strikeouts.

Be nice to have an undo, that is, instead of a restart, because a couple times - the nature of the rules are that, as you're painting this region in order to fill it with the color of the cursor, you can't go back. You can't ever go over what's already been colored. And sometimes I've, like, swiped to the left and realized, ooh, I don't want to do that. And I may be well into solving the puzzle, but there's no undo one step. You've got to start from scratch. So that's - it'd be nice to have that.

Leo: I made TraptionBakery my Pick of the Week on iOS Today.

Steve: Oh, good.

Leo: Keep up those picks. You're saving me a lot of work.

Steve: Perfect.

Leo: Yeah.

Steve: Yes. And that's had some great responses.

Leo: That's a wild one, yeah, yeah.

Steve: So you know that you're in a seriously geeked-out podcast when we're seriously considering the proper position of Spock's thumb for the Vulcan whatever that is - the Vulcan hi, how you doing, mahalo, live long and prosper. It always goes along with "Live long and prosper." And so we discussed this a couple weeks ago. Somebody sent me a screenshot of a very serious-looking Leonard Nimoy making it very clear. And the

moment you see it, it's like, oh, of course. And in fact I remember I was talking about the best man in my wedding who used the red orthodonture bands in order to do his fingers right because he wasn't able to do it otherwise. So the answer is thumbs out.

Leo: Thumbs out.

Steve: Yup.

Leo: Interesting.

Steve: Thumbs out.

Leo: I wouldn't have thought that. I think the chatroom was guessing thumbs in.

Steve: And when you see it, you go, oh, I mean, it's just like so familiar. If Spock didn't have his thumb out like that, it would just be wrong. So thank you.

Leo: It's the Boy Scout salute if you - yeah, out, yeah. Out looks right, you're right.

Steve: Yeah.

Leo: I can't do it. He did it with his left hand, though, I think. Or did he do it with his right hand? I can't remember. I can't do it with my left hand.

Steve: Oh, yeah, I've got both hands under control. But of course I've had a lifetime of experience.

Leo: Yeah, practice.

Steve: And, finally, last week's discussion of using the smart data that SpinRite was feeding back generated a couple questions. And one of them reminded me that there was a third way of using the information that I had forgotten about. And that is to say, and so the tweet was from a Justin Alcorn. He said: "@SGgrc, SpinRite Level 4, 500GB drive, 350 million ECC corrected." So that's, like, during the Level 4 scan that drive had 350 million read errors which ECC successfully corrected.

Leo: Wow.

Steve: Yeah. "And 21 million seek errors." He says, then, "Throw it away? Or did we just fix it?" And I talked about how the problem is that these numbers only mean something in context. And unfortunately the manufacturer doesn't provide us the context. And the

manufacturer has also been sort of forced to provide this information against their will. They want them to be a black box. They want it to be just trust us. And it's only the users of the drive who've said, eh, not so fast there. We need something. Give us some feedback about what's going on. And so the SMART, the Self-Monitoring - SMART. Self-Monitoring...

Leo: I always want to say And Reporting Tool, but it's not.

Steve: You're right. There's another...

Leo: It's almost like they've tried hard not to do the obvious.

Steve: Exactly. And I always stumble back into the obvious. Self-Monitoring - what could A be?

Leo: I'm sure the chatroom will tell us.

Steve: Anyway, someone will tell us.

Leo: You'll have it soon.

Steve: Analysis.

Leo: There you go.

Steve: Self-Monitoring Analysis and Reporting Tool.

[SMART = Self-Monitoring Analysis and Reporting Technology]

Leo: There you go. That sounds right.

Steve: I didn't cheat, it just took a minute...

Leo: Yeah, but it sounds right, yeah.

Steve: ...to dredge it out of my rusty old neurons. So you need - and come on, analysis? Yeah, we wish. No, 350 million ECC corrected errors. Where is my analysis? I mean, that's what Justin is asking for...

Leo: Yeah, right.

Steve: ...is can I have an analysis, please?

Leo: No.

Steve: What does this mean?

Leo: Yeah, mm-hmm.

Steve: So it's the context. And I proposed two ways last week of establishing some context. If you had multiple drives of the same, like you bought five at once so that they are same model, same make, same manufacturing batch even maybe, but run SpinRite on all of them. Now the other four provide the context for the fifth one, and vice versa. So then you'll easily see if one of them looks like a sour apple. It's like, ooh, boy, you know, that's - I don't think I want to put my data on that sucker. It'll stand out from the crowd, probably not in a good way.

The other way, if you don't have that option, I suggested, well, one thing you could do is you'd like to see a uniform rate of errors being corrected. That is, not some region where that number just shoots up skyward so that the maximum - so that there's a region where there's a maximum rate that is much larger than the rest of the drive, which is more near the average or minimum rate. So that's something you could do.

But there is one other thing that I forgot to mention. And that is, when you are deploying drives for the first time, run SpinRite on the drives. Note the level and the numbers. And that allows the drive to establish its own starting context. Then a year from now, or six months later, or a quarter, maybe just burn the drive in and do it again. The point is sometime downstream, when you rerun SpinRite, you're rerunning exactly the same code on exactly the same drive. Well, you know you're not going to get exactly the same error counts on any of these. But you'd like to see it similar, not suddenly wrong. And write those down.

So the point is you can, you know, this should be relatively static. And if any of these suddenly start going, I was going to say south, but technically north, more is probably bad in the case of errors, then that's context. That says something is softening, something is weakening in here, well before any alarm bells go off, well before any data is in trouble. I mean, this is why I've always been so bullish about this smart real-time probing that SpinRite does.

And I put that in, SpinRite 6 is the first version that ever had that, because it is just - it is incredibly sensitive in terms of what potential value it offers. But unfortunately it comes at the price of more user involvement that we would require because the manufacturer doesn't want to say, ooh, this isn't looking real good here. It'd be nice, you know. In that case we would have context. But we have to create our own. At least SpinRite gives you the raw numbers from which you can do that.

Leo: I thought of a way you could get that Security Now! Twitter account back.

Steve: Okay.

Leo: DM the guy. I tried to DM him. He's not following me, so I can't DM him. But I bet you, if he thinks it's your account, he's following @SGgrc.

Steve: You know, okay, maybe he is. I'm sure I tried that once.

Leo: Yeah. Then that won't work.

Steve: I definitely [crosstalk].

Leo: Everybody DM him. Oh, you know what, you can easily see who he's following. I think he's only following one person, so, yeah. Maybe the person he's - actually, I bet you the person he's following is him. I'm sure we can track that account down, get it back [crosstalk]. Do you want it?

Steve: Well, it's been 12 years, you know, it's...

Leo: Eight years.

Steve: Doesn't seem to be a big - oh, yeah. Wait, no.

Leo: Twitter's only been around for 10 years, so it can't have been 12 years.

Steve: The podcast predates Twitter?

Leo: Of course it does.

Steve: Well, no wonder that I wasn't a member in the beginning.

Leo: Yes. Oh, no. Goodness, gracious.

Steve: I'll tell you who's really in pain is the @stevegibson guy. He's like, he's not happy.

Leo: Yeah, he's sorry, he's sorry he did. You know what, he doesn't follow any - this

account doesn't follow anyone. So there's no way we can DM him. So if you're listening, Mystery Person - it's got your album art on it. If you cared a lot, we could go after, you know, we can go to Twitter. But I don't think you care.

Steve: No. And my point was, you know, SGgrc, that's five characters. That's, like, fine.

Leo: It's you. It's you, man, you. Steve, we've got questions.

Steve: I don't believe it.

Leo: Steve's got some answers. And I'm ready with the first one if you are.

Steve: And it looks like seven may just be the perfect number, too.

Leo: Just about the right amount. We've got 45 minutes.

Steve: Yeah.

Leo: Bruno V. kicks this off, this question fest, with a tweet. @SGgrc Hey, at you. OpenVPN with Raspberry Pi. Now remember, it's 140 characters, so it's going to be a little terse, a little succinct. Open VPN with Raspberry Pi. Assuming the Pi can be hacked once exposed to the 'Net, would you recommend placing it on the DMZ? Thanks.

Steve: No. Next.

Leo: That was easy.

Steve: No.

Leo: Don't do that.

Steve: No. So, but this is a great question. So he's saying we're using OpenVPN running on the Raspberry Pi. And we've talked about, for example, the Pi VPN, where one of our listeners built a script which completely installs OpenVPN on, what, a $35 Raspberry Pi. You then plug it into one of your router's unused ports, and you're golden.

The problem is, the point is you want to have OpenVPN as a server running at your home for remote access. That is, when you're out roaming in, for example, in an unencrypted open WiFi, you use the OpenVPN client, which exists now on all platforms, to connect to your OpenVPN server at home, and then it does the decryption, and your traffic emerges

onto the Internet from your home. And/or you have access to your internal network. That is, when you OpenVPN in from the outside, you get an IP address on your LAN. So you then have seamless access to your entire home network, as if you were there. So it's very cool.

But all of this requires that the OpenVPN server or service is reachable from the outside. That means you have to open a port through the router so that it can get to the OpenVPN service. And so what Bruno is saying, well, one way to do that is by using the router's DMZ, the demilitarized zone feature. And essentially what that does is it is the IP address to which any unsolicited incoming traffic is sent.

Normally, as we've discussed often with a NAT router, anything not expected coming in, that is, any traffic that is not a response to something that initially went out is dropped, which is why a NAT router is a natural firewall; so that, as the traffic leaves, a table entry is made to note the exit of that packet. And when a packet returns with the source and destination IPs exchanged and the source and destination ports exchanged, what that is saying is I'm coming back from somewhere you just sent me. And so there will be a match in the table which then tells the router which internal IP, which machine inside your LAN originated that packet. And so the packet destination address is rewritten to the IP of the original sender, and the packet goes into your network and so forth.

So what that means is that unsolicited traffic is ignored. It just dies at the WAN interface because there's nowhere for it to go. It's trying to come in, but nobody's expecting it. There was no expectation created by an outbound packet first, to which that's a reply. So we have two choices. One is you use the DMZ feature as the default destination for unsolicited traffic. Now, yes, you could put your Raspberry Pi IP as the DMZ. It would then be receiving all unsolicited traffic.

But this is exactly the cause for Bruno's question and concern. He says, "Assuming the Pi can be hacked once exposed to the 'Net, would you recommend placing it on the DMZ?" And that's why I so quickly said, well, no. Because essentially what that means is that anything incoming, like all the Internet background radiation that exists, all those unsolicited packets, which we're relying upon our router, that isn't very intelligent, I mean, it's a router, it just says do I know what to do with this, no, and drops it. Now we're saying, oh, come on in. Make yourself at home. And we have a Raspberry Pi ready to consider what to do with you.

Leo: I'll be your host today.

Steve: Yes. It's going to say, well, what do we have here? And so who knows what is going on in the Raspberry Pi. The nature of these things are that you want to minimize the attack surface. What we've just done is spread it all out there for anything to, like, come on in. Maybe you can find a problem. No. So the solution is static port mapping. The idea is that - so the DMZ is the - I'm not sure today, in today's world, if there is a justification. Well, there is one justification for it, and I think it's in the next question. But it's hard to find one for exactly this reason. It was always there. It's just the de facto, sort of the get around the NAT problem. Except it turns out NAT wasn't a problem. It was a blessing. Which is why everyone should be hiding behind one.

So this is the right way to do this, is that the OpenVPN will be listening on ports which you configure. And here's a default one, but it's been so long since I used the OpenVPN default port. That's the first thing you want to do. Don't use the OpenVPN default port. It doesn't matter which one you use. And I have, for my application, I use a bunch of

different ones. And, for example, 110 is nice because that's POP, which is generally accessible everywhere, the Post Office Protocol, because people tend not to block it. ISPs block 25. You won't have any success there. Or 80, for example, that's the HTTP port. Well, you probably don't have a web server running at your house, so you could put OpenVPN there, although that sort of exposes it to anything else that might be probing around the 'Net for web servers. So maybe stick it up in the higher port numbers. But just, you know, make up some numbers. You want it to be sort of obscure, not to rely on that, but why not also have that?

And so the idea is that you tell your router, any traffic coming in to this port, whatever it is, 123123, should be sent to this IP. And that is the IP of the Raspberry Pi. That way, rather than opening the floodgates, you're just poking a little pinhole through your NAT saying, if any packets are smart enough to try to come in at 123123, at that port number, we're going to let them go because they're probably us over at a Starbucks somewhere, or in an airport, trying to reach our OpenVPN router.

And the good news is that that port will come through only going to that port on the Raspberry Pi. So nobody outside can scan around. Essentially, if you use the DMZ, they can do a port scan of your Raspberry Pi and find everything that it might have open, rather than just a little pinhole that allows only the packets that know the secret, well, or obscured port number to sneak through the NAT boundary and get into your Raspberry Pi. And I think we've pretty much covered that topic.

Leo: Mike Chapman is next. Hi, Steve. I'm seeing more websites ask for a username before a password instead of together. I'm thinking it's less secure. Is it? Love the show, et cetera.

Steve: You know, that's a great question. So he says he sees more websites ask for a username before a password.

Leo: Google does it, too, yeah. I've seen [crosstalk].

Steve: Yes [crosstalk].

Leo: And so that makes me think it's more secure, frankly.

Steve: Well, it's a function of the way they handle it. And this is why Mike's question made it into our Q&A. We know how rare these are. So it's got to be a good question. Because if the website does the wrong thing, then the separate handling of username and password lets it be probeable. Remember that the famous weakness in WPS was the web auto configuration. It was an eight-bit token, but you were able to guess the four - or, I'm sorry, it was an eight-digit token. But you were able to guess the first four digits separately from the last four, and technically the last three, because the eighth one was always a check digit, a checksum.

And so what happened was that meant that you could only - the designers intended for you to have to guess them all as one, seven digits with a check digit. And that would be enough combinations that it just was infeasible to crack it. But an error in the protocol allowed you to independently, because it was a multiple handshake, and we covered this

in detail at the time, you were able to separately guess the first four. That reduced the possibilities to 10,000 guesses, which became feasible. Now you had the first four, you only had to guess the next three, which was only a thousand possibilities. So by cracking the problem into two separate pieces, it dramatically weakened the strength.

And exactly the same thing happens with username and password. A site which allows you to enter your username and verifies it for you in any fashion, without then always asking for your password, think about it, that allows you to separately probe for usernames. If instead a site either asks for them both at once, or if receiving your username never changes its behavior and always asks you for your password, then you're not able to probe the username space separately from the password space. Now, some people might say, oh, that's really not such a big deal. It's like, okay. I wonder if WPS designers thought that at the beginning when they allowed the protocol to handle the eight digits in two separate pieces, and it completely destroyed the security of their solution.

So he says, "I'm thinking it's less secure." He's right. It's technically less secure. Maybe it's not enough to matter. But it is an issue that web designers should keep in mind, that is, it does allow usernames to be tested for validity without needing to be accompanied by a password. And if instead the site responds always by asking a password, or asks for them both together, then you don't know which of the two you got wrong. And that's the point. You don't want to have this problem capable of being broken down into smaller pieces. It's just not good. So great question, Mike.

Leo: Question 3 comes from Joseph W. Barlow, a.k.a. @sobokuone, his Twitter handle. Hey, @SGgrc, I bought the Eero router, but I'm now finding they don't support stealth ports. Should I be concerned? All ports are closed.

Steve: Okay. So this is why I was talking about the DMZ again, because what this means is that, for whatever reason, the router - well, okay. I said for whatever reason. Technically, stealth is a violation of the RFCs. That is, and as far as I know, I'm actually the person who coined the term, because of course it came from Star Trek and ShieldsUP! and stealth ports and so forth, because that was quite a while ago.

The RFC states - the formal specifications for the Internet - that, if a packet is attempting to go to a server, and it's certainly the case back in 1973 in the chart that we discussed at the top of the podcast, is as a courtesy, because after all we're all on the same team, we're all pulling in the same direction; right? We'd like to tell somebody, oh, sorry. Your data arrived. We looked at it, but we don't have a service listening for incoming traffic on that port. In other words, there is always a reply to an attempt for a packet of this sort that might receive, for example, if it's a TCP packet, it might receive a TCP ACK as part of the setup process.

What you don't want is a black hole; especially, for example, in the case of DNS. DNS is, as we know, doesn't have a delivery guarantee as part of the protocol. So you make a DNS query over UDP. And it's the UDP protocol that is not guaranteed to itself deliver the packets. So if you don't get a response, you assume that the response either was lost in returning to you or lost on the way there, per the original definition of the Internet, where everything always responds. And it's nice that everything always responds because then, if you do make a query over UDP, for example, for DNS, and you get a response back saying, sorry, no service here, there is actually a port closed response.

In this case UDP doesn't have it, but ICMP has it. So the ICMP protocol is used to send

back a, sorry, that port is not open. Well, then you affirmatively immediately know that, okay, that there's no point in retrying. It's not that there was a lossage somewhere, or a routing problem, or a congestion or something that lost your query or your reply. You know you're never going to get one from that IP at that port number. Which the designers intended. The problem is it also makes the entire Internet probeable. That is, you then, if this were still the case, if stealth didn't exist, then people would be knowing that they would get a response from any IP and any port which exists and did receive their incoming traffic. And the decision was made, very informally, this is not endorsed anywhere, it's like, eh, maybe we should just pretend we're not here, and thus this concept of a stealth port.

So Joseph is saying there's no option on the Eeros. Now, there are some routers that automatically drop packets that are unsolicited. There are some where you can tell them I want to respond. Sometimes ping has a separate checkbox in the UI, where you can have the router respond to pings, but not other ports. Sometimes these things are configurable. But the fallback position, if you do want stealth - and so my point was that there are purists who would argue that stealth is by its nature a violation of the way the Internet works. Technically NAT is, too, because you're not supposed to go around rewriting, addressing information on packets. They're supposed to be inviolate. But sorry about that. It's not the way reality and the Internet intersected.

So I don't think it's a huge problem if your router is not stealth these days. But if you want it to be, and if the router does support a DMZ, then just as we were talking about in response to Bruno's question, the DMZ is by definition where to send anything that is not otherwise expected. Well, where do you send it? You send it into outer space. You configure the DMZ IP to something on your network that doesn't exist. And normally the DHCP configuration, the dynamic host configuration protocol address range on your LAN, normally it's not from zero to 255. Sometimes it's like one to 100. So you know that no machine on your LAN will have the IP ending in 101 because the router's not automatically allocating addresses in that range.

So, for example, set the DMZ IP to something outside of the range that your router is using for LAN addresses on your network. And anything coming in, the router will rewrite the packet's address to that nonexistent IP and stick it on the LAN. Well, there's nowhere for it to go. If it hits a switch, the switch will have never seen any device with that IP. It'll just discard it. And so you can create stealth by using this little DMZ hack.

Leo: That's clever. I have Eeros at home. I'll try it. I didn't even think about that. Of course I should have run ShieldsUP!. First thing I do with any new router. I wonder how many of these simple mesh routers do stuff like that. I know I can do - surprisingly, the Eero has a lot of things like DMZ and port reservations, port forwarding, stuff like that. So certainly your solution, your workaround is possible. But I will try it when I get home tonight. I also have the Google mesh WiFi solution, so I'll try it with that.

Steve: Yeah. We're seeing now router firmware is mature enough that, even though they may hide it under advanced setup and advanced settings and things to sort of keep regular users from being freaked out by it, there's really no excuse not to have all that technology available in routers. This just may be buried a little bit.

Leo: They can do it. I think they don't, they choose not to, mostly because they

don't want to overwhelm people.

Steve: Exactly.

Leo: It's not a geek router, that's for sure.

Steve: Right.

Leo: Does the job, though, in a way that the geek routers don't seem to. So Reston Wiles in St. Louis has titled his rant "WiFi Lite Bulb Morons." Shall I do it like that? I'm mad as hell, and I'm not going to take it anymore. What I cannot understand is the perceived need, use, desire, or the stupidity required to purchase expensive idiotic WiFi lamps. I don't use WiFi and as - I realize now as I read this I've got the wrong voice. I don't use WiFi and as little BluT as possible.

When I transfer data to a device I plug it in. When I need to turn on a lamp, I reach out and turn it on. If I want to fake lamp use in unoccupied home while traveling, I put some on timers and a couple of photocells. I have hardwired security cams well hidden. Motion sensors turn on porch lamps and security cams. I've missed the entire IoT brain wash. Explain to me, Steve, why people use this stuff. How lazy, cool, and trendy do you have to be? At what diminished IQ level does IoT become a necessity? And while you're at it, get off my lawn. Sorry.

Steve: With apologies to Reston.

Leo: I'm sorry, Reston.

Steve: Thank you, Reston, that was...

Leo: WiFi light bulb morons.

Steve: And Leo, thank you.

Leo: You're welcome. A dramatic reading.

Steve: And I'm so entertained I have no idea what the question - what the answer to the question is.

Leo: I don't think it's a question. The question is, "Why does one do that?" he says. Why would you do that? What's wrong with your IQ? You kids today and your fancy doodads. In my day.

Bob Raffety in Florida is a bit puzzled by ShieldsUP! with a VPN: I recently purchased a VPN service, Windscribe, and decided to see what happened when I tested it with ShieldsUP! at your site.

Steve: Uh-oh.

Leo: People still use ShieldsUP!. This is so awesome. The results - oh, boy - were many open ports: 20, 21, 80 - well, 80's always open - 143, 443 also should be open; right? Well, not, no, only if you're running a server, I guess, in inbound ports, on the Windows 10 Pro computer. Their support indicated the results showed the status of their remote server and not my computer. Huh. The software client has a firewall-enabled switch which is enabled when the VPN is active and turned off when not enabled. My background experience tells me they're pulling my leg. Get off of my leg. Has there ever - sorry. Shhh. Your part is over, sir.

Has there been any test with the many VPN services and ShieldsUP!? How can I reliably test my computer while the VPN is active? My experience is over 30 years of tech support. I was introduced to SpinRite back in 1990 when working with a computer support company and purchased it back in the early 2000s. I listen to your podcast, but I'm a few months behind. So if you've talked about this, just let me know which podcast.

Steve: So this comes up all the time. And I've even seen some very rude VPN tech support people saying, oh, you can't trust that ShieldsUP! thing. It has no idea what it's talking about. And that's like, okay, let's all take a deep breath. First of all, Bob, these guys were exactly right. But they may not have understood what was actually going on.

When you contact GRC's website or - and we can broaden this to the Internet, any website. Those websites, mine and others, see your IP as the IP of the VPN provider. That is, when your packets emerge from the VPN service, Windscribe in this case, the source address of the packet, which used to be you at home, is now changed to the VPN service. So that the packet then goes out onto the Internet, does whatever it needs to do, and answering packets return to the VPN service.

So what that means is that Internet websites see you as the VPN's IP, as the VPN service. They don't - your actual IP is obscured behind their own lookup table, which knows where to send that VPN packet to get it back to you. But out on the public 'Net, we all just know somebody using Windscribe is at our website. So when you as a VPN user test ShieldsUP!, I'm not seeing, GRC is not seeing your Windows 10 Pro computer. I don't know its IP. And that's one of the reasons you use the VPN is to hide your IP. So I'm telling you about the ports that Windscribe has open because that's how I know you. I know you by your public IP on the Internet, which because you're behind Windscribe is their service.

So now we know Windscribe's VPN servers have ports 20, 21, 80, 143, and 443 open. And so it's interesting to know what ports a VPN service may have open. And those look reasonable. Yeah, I don't see anything really wrong with that. Of course, it's a function of what they are doing with them. And it may probably have other high-numbered ports open, as well, which ShieldsUP! could test for you. You just have to ask it to do that explicitly because it otherwise just does the service port range from zero. And I actually do test zero because there are such things, up to a little past 1024.

So anyway, many people wonder about this. They kind of like want to check their security, thinking, ooh, you know, I'm behind a VPN. It should be better. But then they see these open ports, which is confusing. And, yes, it's because I'm no longer able to see your computer, which is what you want. And so I'm showing you what we do see, which is the VPN service. Which is all anybody can see of you when you're behind a VPN.

Leo: That's the point.

Steve: Yeah.

Leo: You're testing the VPN, not you. If you could test you, you'd obviate the point of the VPN. Jeff Gros, Yorba Linda, looking for - wow, ShieldsUP! day today - a ShieldsUP! command line: Steve, I'm planning on setting up a Raspberry Pi behind my EdgeRouter X - love these Ubiquiti routers, BTW. Can't stop buying them. They're the 50 buck routers that I still have just sitting around. I've got to figure out something to do with it - to remotely monitor and control my 3D printer. That means I'll have to port forward to the device. However, I would like to double-check my security once I get the device set up. I haven't set it up yet as I don't have the Raspberry Pi yet. But I'm assuming it will command-line only.

That's not true, actually. You can have a browser in a Raspberry Pi, so you can do all of this. Is there a terminal version of ShieldsUP! that I can run? Or perhaps there is a clever way to do this with Linux commands? Any help is appreciated. Thanks. Jeff. Just a side note, the Raspberry Pi does in fact have a full GUI with it. Raspbian is a Debian distribution, and you have a browser and everything else.

Steve: Nice. So we've never really talked about user-side port scanning.

Leo: Yeah, because you could do everything VPN, I mean, that ShieldsUP! does yourself; right?

Steve: Yes. And so the, well, yes, you can.

Leo: Can't really say what the outside world sees unless you go somewhere - yeah.

Steve: Correct. Correct. So the advantage of ShieldsUP! is that it gives you a public-facing view, that is, it's GRC looking back at you from the public to see what you're exposing. So that exists. Now, the alternative is to know what's going on internally. And GRC can't, and it shouldn't be able to see because you're behind a router, which as we just have been discussing is blocking all of that. So there are all kinds of tools that have been written for local port scanning. But there's one that stands out from all others. It's the granddaddy known dearly and "indearly" to every hacker, and that's NMAP, N-M-A-P. NMAP.org/download.html will take you there.

Leo: And if you need help with a mnemonic, that's what it stands for, network mapping tool; right? [Crosstalk], yeah.

Steve: Correct, correct. Now, the caution is don't freak out because it's going to show you stuff that's going to curl your hair. The point is you're hiding behind your NAT router. And all kinds of stuff in your network has ports open. Apple's got Bonjour talking to everybody, and TiVos have a bunch of ports open. And there's like, there's this amazing amount of stuff going on on your LAN, and NMAP will find it all and show you what's going on.

The key, though, is that would be a problem if, well - and this is one of the reasons why I'm a little worried about IPv6 giving us 64K IPs per subscriber is that, I mean, this router is a very good thing to have. You want no one to just be able to reach into your network. NMAP on your LAN has full visibility to all the devices and all their ports on your LAN. You want to keep that to yourself. You want to keep that behind a router.

But with that understanding - oh, and there is now a GUI frontend for NMAP. It used to be, I mean, for the longest time it was magic incantations and a lot of help information available about how to do this and that, what IP ranges to scan, what port ranges to scan, how many times to look, how long to wait between. I mean, it is a feature-rich, massively mature port scanner/network analysis diagnostic. I mean, it'll find every device on your network, things you forgot about, things that are, like, plugged in behind the sofa that you - it's like, oh, that's still there? I mean, it's great. But again, understand that ShieldsUP! will show you what you care about if you're worried about exposure to the world.

What any local scanner will show you is how much you need that router to be in between you and the rest of the world; why I'm so bullish about it as a native firewall; and why, even after we start getting blocks of IPv6 space allocated to us, you absolutely want stateful packet inspection in between your home and the public Internet. It's just - actually I would argue that LAN-based systems take that for granted now. They don't even have to worry that much about their security because they're going to be behind a router that's going to be the thing responsible for keeping them safe. So NMAP.org.

Leo: NMAP. Lee in the U.K. says he's going to work around the Snoopers' Charter. Hey, Steve. Long-time listener, love the show. You've mentioned a few times now about masking your Internet activity by using a VPN to connect back to your own home pfSense - that's the firewall Steve uses - or equivalent system, then doing your Internet searches from your home. This has been my solution for some time now using the Sophos UTM, formerly the Astaro Gateway, former advertiser on the show. However, I'm based in the U.K., and so this isn't sufficient any longer since our government last week passed the Investigatory Powers Bill or Snoopers' Charter. Now even the Food Standards Agency has access to the list of domains I visit.

So my current thinking is to ditch the Sophos UTM - it cannot unfortunately be configured to route external traffic through a VPN unless it routes to another UTM - in favor of pfSense, which can be configured to route traffic to any OpenVPN connection. My current choice is NordVPN. I'm thinking that's out of Norway, or certainly not out of England. I want to get your thoughts on this, also let you know that the notion of connecting home so your employer can't see what you're browsing is now scuppered by the U.K. government - if, of course, you reside in the U.K. I

hope you get time to cover this on the show. All the best. Lee.

Steve: So I just, you know, this sounds like, I mean, first of all, we've discussed and covered the Snoopers' Charter. This sounds like he's interested in a standard VPN application to hide his traffic and get it out of an area where it is subject to surveillance. And so he's doing, I mean, this is what he wants to do. And as long as he chooses an endpoint that is outside of the U.K. or wherever the legal regime is that he's trying to thwart, then yes. Law enforcement will know that his IP is routing encrypted traffic out of their boundary. And as long as that's allowed, that system will work.

My guess is that this strategy may not have much life because you can certainly imagine, if the decision has been that the government needs to be able to see into everyone's traffic, then one way or another they're going to arrange to make that happen. And I don't know what that means, if that means maybe you have to, like you are required to use local-ish VPN services so they can see the decrypted traffic. Remember that, even then, even with a VPN, as we've been saying, the bulk of traffic is still HTTPS tunneled. It is in a TLS tunnel that is encrypted and authenticated. But DNS queries are not.

And I've even seen some VPN configurations that did not tunnel DNS. Which is to say they tunneled traffic over TCP, but not necessarily the machine's own DNS queries, which of course is a privacy leakage for that VPN configuration because, as we know, DNS does not have an encrypted flavor, and you would need to be using something like DNSCrypt with a service like OpenDNS in order to obscure where you were going. So to me, I get what Lee is trying to do. Unfortunately, he's operating within a government that has decided they're going to legislate themselves the ability to see what everyone is doing. And they're bigger than he is.

Leo: Mm-hmm. Yeah. Yeah, that's encouraging. So thanks for ending on a high note there, Steve.

Steve: We should have done Mr. Grumpy as our final question.

Leo: Mr. Grumpy should have ended the show. Well, you know, the nice thing is if you don't use the Internet, no one knows what you're thinking. Or something.

Steve: Yeah, that's actually true.

Leo: Yeah. It's a lot harder to figure out.

Steve: And if you do, then everybody does.

Leo: Yeah. Steve is on the Internet, GRC.com. He's also on the Twitter, @SGgrc. You can go to GRC.com to get SpinRite, his bread and butter, his daily bread, the program he makes available to you at a fee well worth the price for the world's best hard drive maintenance and recovery utility. Go there. And while you're there you

can also get a copy of this podcast. He has the audio plus transcripts, which is nice, for every show ever, all 590 of them. He's also got other stuff, free stuff, lots of free stuff, good stuff there. In fact, it's kind of a fun place just to wander around through the archives of Steve's mind as placed on the Internet: GRC.com.

We have the show, too, audio and video, if you want, at TWiT.tv, and in this case TWiT.tv/sn for Security Now!. And of course you can find it on every podcast client. We do the show every Tuesday, right after MacBreak Weekly, so that's about 1:30 p.m. Pacific, 4:30 Eastern, 21:30 UTC. Please join us live. Join us in the chatroom. Chill with us. Hang with us. You can even visit us in the studio. Email tickets@twit.tv.

One more day to get your TWiT Army shirt, great design by Anthony Nielsen. Go to Teespring.com/twit, T double E, Teespring.com/twit. And we are going to do a TWiT Store, and that will be one of the designs in the store, but it might cost a little bit more because we will be mass printing them. So if you want it, and you want it at this price, and it's a good price, go now, Teespring.com/twit. I think tomorrow is the last day, maybe the day after tomorrow.

We are also looking for best-ofs. Not for this show. We've decided, Steve's decided to reprise one of the most popular shows he's ever done, the story of the Portable Dog Killer. No dogs were injured in the making of this show, I promise you. But that'll be the Tuesday after Christmas. The rest of the shows, though, we will be doing best-ofs, and we'd love to hear from you. If there's a moment from this year that you would like to recapitulate, go to TWiT.tv/bestof. Give us whatever information you can. We ask for a lot of information, but you don't have to know it all. Just give us whatever you know, and we will help our editors put together the best-ofs, as we do every year. It's always fun. Steve, we've got one more show before the holiday break.

Steve: Wow.

Leo: I know, can you believe that? Next week.

Steve: Cool.

Leo: And I will see you then.

Steve: Thanks, buddy. Talk to you then.

Copyright (c) 2014 by Steve Gibson and Leo Laporte. SOME RIGHTS RESERVED

This work is licensed for the good of the Internet Community under the Creative Commons License v2.5. See the following Web page for details: http://creativecommons.org/licenses/by-nc-sa/2.5/