Listener Feedback #236

Description: Leo and I catch up with a fun and interesting week of security happenings, including an expensive Windows update, a worrisome FBI hacking court decision, a fix for slow Windows 7 updating, more Comodo slime, JavaScript cryptomalware, yet another way to exfiltrate data from an air-gapped computer, a worrisome Netgear router flaw, the COOLEST brilliant new idea of the year, some miscellany, and questions and comments from our terrific listeners.

High quality (64 kbps) mp3 audio file URL: http://media.GRC.com/sn/SN-566.mp3
Quarter size (16 kbps) mp3 audio file URL: http://media.GRC.com/sn/sn-566-lq.mp3

SHOW TEASE: It's time for Security Now!. It's been a busy week. I'm really glad to be back. Steve Gibson will have a rundown on the latest security news, including a way to speed up those updates on Windows 7. And then we'll answer 10 of your questions with 10 of Steve's answers. It works out well that way. Stay tuned. Security Now! is next.

Leo Laporte: This is Security Now! with Steve Gibson, Episode 566, recorded Tuesday, June 28th, 2016: Your questions, Steve's answers, #236.

It's time for Security Now!. I'm back, baby. And so is Steve Gibson of the GRC - what?

Steve Gibson: I never left.

Leo: You never left. You never left.

Steve: I never left.

Leo: In fact, by the way, first of all, thanks to Father Robert for filling in. Did a great job. Such a good job, people are saying, why is Leo back? So I'm sorry, I'm back.

Steve: Well, we've solved the problem of you taking vacations.

Leo: Robert's fantastic. The sad thing is, I didn't realize, and you didn't say - you were so cute. Because I'm telling you, I'm going down to Newport Beach, I'm going to stay at the Pelican Hill, and you're going, "Mm-hmm, mm-hmm." It turns out I'm like a mile from you.

Steve: Actually, you never mentioned it on...

Leo: I didn't. Oh, okay.

Steve: You didn't mention it before you left. And I only knew because I think I heard you say it to Paul and Mary Jo on Wednesday. And then I thought, well, now, wait a minute. And I sort of did a little bit of math, and I thought, I think that means that I'm not going to have Leo next week. And so...

Leo: I apologize for not telling you that.

Steve: I sent email to Father Robert, and I said, are you co-hosting with me? And he had a busy weekend, and I didn't hear back from him till the beginning of the week, and he apologized.

Leo: I'm so sorry.

Steve: He said, "Yeah, sorry I didn't get back to you sooner."

Leo: No, that's my fault, I thought I had told you. I totally apologize. But I even further apologize because I'm literally next door, and I didn't realize it before, but I realized it when we got down there, and I'm driving by Irvine, I'm going, "Oh, that's where - oh, we're here?"

Steve: The only thing I could have provided mostly was tips on the right places to eat because...

Leo: Yeah, because we went to all the wrong places to eat.

Steve: ...that's what I do. And so, yeah. I have a nephew who's with Salesforce, and he's now traveling a lot. And so he spent a couple - he's got a couple business meetings down here in Southern California. And so we'll get together, and I've been introducing him successively to one fabulous restaurant experience after another.

Leo: Well, I now know why you live where you live. For some reason, when I hear "Irvine," I feel like it's like City of Industry or something. It's not. But Newport Beach

area, that coastal area is gorgeous. It's out of town. So you have the benefits of being in a city, but not living in the smog-laden L.A. valley.

Steve: What happened was, when I left the company that I had been at for a few years previously, and the president of a company down here in Southern California phoned, and he said, "Steve, I heard you're available." And I said, "No, Paul. I've been working my butt off for a couple years. I'm just going to take some time and see my friends again, who I haven't seen forever." And he said, "No, no, no, no, we want to hire you." And I just sort of said, "Well, no, you're down in L.A." And he said, "No, no, no. Irvine is not L.A." And I was very skeptical. And he said, "Let us just fly you down, and we'll spend a day, and then you can see what you think." And it's like, oh, okay, fine. So they flew me down into what at the time was a shack of an airport. It was...

Leo: It's still pretty small.

Steve: It was, yeah, the Santa Ana Airport. And then picked me up and drove to Balboa Island. We took the ferry over there.

Leo: I love that. Isn't that awesome?

Steve: We drove down the coast to Laguna. And I'm looking around thinking, wow.

Leo: It's just amazing.

Steve: This is not Los Angeles.

Leo: Yeah, yeah.

Steve: Because I'm really - I'm not a city person. I'm a suburb person. And so I spent the day down here and saw the company. And I said, okay.

Leo: You didn't take the job, but you took the town.

Steve: No, I did take the job.

Leo: Oh, you did take the job.

Steve: Stayed for exactly one year. I hated it, and I quit on my anniversary. Get me out of here.

Leo: And of course that little shack is now the John Wayne Airport.

Steve: Oh. And, please, do we have to - of course, you guys have the Charles Schulz Airport, so I guess that's what we do is we name small airports after famous people.

Leo: What's nice is I can fly from Charles Schulz Airport to the John Wayne Airport, and it's a nice, easy flight. So we'll come back. I'll come back and say hi.

Steve: There's now a direct flight. It began on March 26th, actually. It's Alaskan Air that has a point-to-point for the first time.

Leo: We drove because we wanted - I don't know why. There was four of us, and I thought it'd be easier.

Steve: I know, and so I was figuring you weren't seeing that second-to-the-last episode of "Game of Thrones" because you were on the road probably.

Leo: No, we saw it the night before we saw the last episode. So we saw them back to back almost.

Steve: Oh, okay. Right, right. So you delayed that one by six days. Yes, nice.

Leo: We have the technology now. It's kind of amazing. Wow.

Steve: Oh, and thank god we do because I don't know what I'd do without my...

Leo: Lisa's saying, okay, we're not going to watch "Ray Donovan" until the whole season's over, and then we'll watch it all at once. And I said, yes, let's do it. So what did we do? We turn on the TV and, oh, "Ray Donovan" is on." So we watched that. It's hard. It's hard.

Steve: So it's Q&A day.

Leo: Yay.

Steve: Number 236. This one, I have a really good feeling about this podcast. Lots of fun news. We had, I mean, and boy, this was just a Twitter-driven podcast because our listeners were sending me, breathlessly tweeting, things that were happening. And I said, oh, yeah, yeah, it's in the show notes already. Oh, yeah, yeah, I know. So we had one particularly expensive Windows update for Microsoft, a troubling court ruling about FBI hacking computers, hope for slow Windows 7 updates. Comodo drops to a new low level

of slime behavior. We've got new cryptomalware in pure JavaScript. Yet another way to exfiltrate data from an air-gapped computer.

And you're not going to believe this one. A worrisome flaw found in most Netgear routers. And in fact I made it the bit.ly link of the day because people are going to want to check their Netgear router to see if it's among those. And so I just thought, okay, that's the best use for this week's bit.ly link. And what is probably the coolest and truly brilliant idea of the year. Just so clever. Then we have a little bit of cool miscellany I'll breeze through. And then, really, some really great questions and listeners following up on last week, and a bunch of other new stuff. So I think I can promise a really entertaining and informative two hours.

Leo: And jam-packed, as usual, which is great.

Steve: Yeah. I've got a great model that I really like with GRC's old-school NNTP news server because we just have a really perfect community there that...

Leo: This is mostly for SQRL; right? Or is it for other stuff?

Steve: Well, and it'll be for SpinRite, as soon as I get back to it, when SQRL's done. But absolutely for SQRL. And then - but like all the stuff I've done. The DNS Benchmark was the same way. And basically all of the projects I do, I do it in a community where I provide it to them, get feedback, lots of great ideas, and then go away, gin up the next iteration, and say, okay, how about this one?

Leo: Steve's so old-school, I love it. And actually, I mean, we use IRC, which is as old-school as NNTP.

Steve: Yeah. Well, when I...

Leo: Which predates, by the way, the web. That's how old this is.

Steve: Yeah. Well, and what works is that it's just text. There's no rich media. It's a pain to, like, attach anything. But really it's just - it's dialogue is what we really want.

Leo: Right, right.

Steve: When I mentioned recently bringing up a FAMP stack, a FreeBSD, Apache, PHP, and actually it's MariaDB, that's for GRC's forthcoming web forums because I recognize, once SQRL is done, we've got something like a quarter million listeners because we know we had 165,000 in one week, so that's probably two thirds of the total download. The point is there will be a huge number of people playing with it. And I can't accept email from everybody who's got a question or wants to know something. So there needs to be a public place to organize that. And so I'm in the process of bringing up for the first time a public-facing, high-visibility web forum. It won't replace what we've got going on in the

back because that's just so valuable to me. But I just need something where people can ask questions and sort of help each other.

Leo: Yeah. Yeah, in about 20 years you'll get to GitHub and JIRA and, yeah. No, I'm actually 100% supportive of this. It's use the right tool for the job. Whatever works for you is the tool to use. And it's silly to abandon classics just because they're old. I mean, I think that's fine. I don't have a problem.

Steve: Well, I'm looking at my XP desktop here.

Leo: I know, I know.

Steve: And it's working just fine. Speaking of.

Leo: Yes?

Steve: The Seattle Times, Matt Day wrote sort of the reference coverage of this. All the other articles ended up sort of referring back to Matt's. And he also quotes Mary Jo and Paul. So this is about Microsoft settling a lawsuit that a woman whose name is Teri Goldstein brought because Windows 10 was just such a problem for her. Matt wrote: "A few days after Microsoft released Windows 10 to the public last year, Teri Goldstein's computer started trying to download and install the new operating system. The update, which she says she did not authorize, failed. Instead, the computer she uses to run her Sausalito, California travel agency business slowed to a crawl. It would crash, she says, and be unusable for days at a time." He quotes her saying, "I had never heard of Windows 10. Nobody ever asked me if I wanted to update."

"When outreach to Microsoft's customer support didn't fix the issue" - and by the way, she, like, spent days on the phone, so this is not like she's complaining easily - "Goldstein took the software giant to court, seeking compensation for lost wages and the cost of a new computer. She won." Then Microsoft, of course, couldn't let that stand, immediately said that they were going to appeal. However, last month Microsoft dropped their appeal, and Goldstein collected a $10,000 judgment from the company.

And then Matt goes on, but later in the article he says: "Mary Jo Foley, a journalist who has closely followed Microsoft for decades, wrote recently that the company has made saying no to Windows 10, particularly for non-savvy people, quote, 'nearly impossible to implement.' Paul Thurrott, another longtime Microsoft follower, criticized a recent popup asking users if they were ready to get Windows 10. In the prompt, the X in the upper right corner, long known to Windows users as a way to exit a software program or abort a process, is interpreted by the update tool as an agreement to go ahead with Windows 10." He then quotes Paul saying, "The violation of trust here is almost indescribable."

So anyway, I think clearly what happened is Microsoft was concerned that they might lose on appeal, and that would have set just a horrific precedent for all of this Windows 10 upgrade fervor. And so they thought, you know, better not to set that precedent. We'll just, I mean, Microsoft said they didn't want to invest any further in this case. And it's like, come on. Microsoft doesn't even feel $10,000 or whatever their cost of litigation. If they thought they were going to win on appeal, that's what they would have done in

order to foreclose anybody else thinking that this was a good idea. So I think they just thought, look, we can't afford to lose on appeal.

Leo: The good news is our long national nightmare is almost over because it ends July 29.

Steve: Yes. I was interviewed for an article in the San Jose Mercury News last week, and I have an interview scheduled with Consumer Reports tomorrow on the whole - on my involvement in this with Never10. The downloads have slowed from about 35,000 is where they peaked per day, down to 15. And we're at about 1.365 million downloads of Never10.

Leo: Wow.

Steve: So it's been going crazy.

Leo: Wow. That is amazing. And the thing about that is there's no reason to download it more than once. So that means really very close to that number of people have used it on that number of Windows installations. That's really impressive.

Steve: Well, and as you'd expect, all the other download sites have jumped on and are offering it also.

Leo: Right. With their ugly wrappers.

Steve: So it's maybe probably not as much as GRC, but maybe in aggregate again about that much.

Leo: And according to Ghacks.net, this is the new Windows 10 invitation, which is much clearer and has a button that says "Decline" on it.

Steve: Yes.

Leo: And it also has much better prose underneath on what to expect, including that you can roll it back, and how big it is, and it's going to be a big download. I mean, I think this is good.

Steve: Yeah. So they finally, like when they exhausted all other possibilities, they said, uh, okay, fine.

Leo: Black mark. It was just a black mark on their escutcheon.

Steve: We'll do what we should have done. It's, yeah, crazy. Okay. And this is the second most brought to my attention issue was a very troublesome ruling which is tantamount to the court saying that users have no expectation of privacy for the contents of their own computers in their homes. So, okay. The timeline here is back in 2014. The FBI tracked down just a horrific child pornography site, just horrifically called "Playpen," of all things, which was a Tor-hidden service. And their strategy was to also rope up, not just the people who were running the site, but to get the people who were using Tor as an anonymizing service to access this creepy site.

So they kept the site up, and they arranged to inject some of their own technology - it was called "NIT," N-I-T, for Network Investigative Technology - into the machines of subsequent visitors to this hidden Tor child pornography site called Playpen. So one of these creeps, first of all he's alleging that the traffic or the bandwidth that the FBI has obtained from his machine may have been unencrypted at some point, so some third-party man in the middle could have injected this content. In other words, he's saying - he's trying to use the defense that, oh, no, they didn't really get this from my computer. My computer is clean. I'm innocent. This came from someone who somehow thought to, I mean, that's this guy's defense.

So the problem is, and we have zero sympathy for this person and anyone else who's there, but the problem is the ruling of the court in this case and the possibility for creating some very worrisome case law because some of the opinion which the judge has generated hinges around IP addresses and whether IP addresses - because of course that's what the Tor network is hiding. The whole point is by jumping through Tor nodes you're hiding your IP address. So the question is whether IP addresses are private and subject to Fourth Amendment protection or already public.

So there's a senior United States district judge named Henry Coke Morgan Jr., who wrote: "Generally, one has no reasonable expectation of privacy in an IP address when using the Internet." This, he posits, is because we all voluntarily give up our IP addresses to third parties every day, such as to our ISPs and any websites we visit. And when it comes to Tor, users have to connect to and disclose their IP address to the initial node of the network. Well, of course, that reasoning is a little flaky because the whole point of using Tor is to hide your IP. And we know that the Tor system does the best, is the best technology we have currently for making that happen.

And then there's something else interesting that I encountered as I was digging around into this, which sort of factors into some of the reasoning, which is known as the "broken blinds test," which allows passing law enforcement officers to legally peer into someone's home if their blinds are closed, but broken in such a way that there's some visibility allowed. So there was clearly some case law in the past where this was argued over somebody had their blinds closed, but one of the louvers was a little bit ajar, leaving a crack, and so somebody was able to see in through the broken blinds. And so that generated some case law about whether or not the intent of the owner was to have privacy and so on.

So then, continuing, the FBI's investigative software then grabbed more than just the suspect's IP addresses. It also obtained their username and some other information from their machine and sent it to the FBI. And that information is undoubtedly within the user's computer. So there was some argument here that, well, the IP address is not actually in the computer. It's the way the person connects to the Internet. So that's

really not theirs, and it's not their private information. And so the argument was, oh, but what's inside the computer is private. Well, but what the FBI got came from inside their computer.

So then again, in arguing against that logic that this person's defense attorney was coming up with, Judge Morgan said that - he said the defendant has no reasonable expectation of privacy in his computer. He wrote: "The NIT [Network Investigative Technology] only obtained identifying information. It did not cross the line between collecting addressing information and gathering the contents of any suspect's computer."

But then, finally, most horrifyingly, he writes - and this whole case is full of horror. He says: "It seems" - get a load of this. "It seems unreasonable to think that a computer connected to the web is immune from invasion. Indeed, the opposite holds true: In today's digital world, it appears to be a virtual certainty that computers accessing the Internet can, and eventually will, be hacked." He then references a series of media reports on high-profile hacks and posits that users of Tor cannot expect to be safe from hackers.

Now, of course our friends at the EFF are apoplectic over this. And I'll just share the beginning of a longer post where the EFF wrote: "In a dangerously flawed decision unsealed today, a federal court in Virginia ruled that a computer defendant has no 'reasonable expectation of privacy' in his personal computer located inside his home. According to the court, the federal government does not need a warrant to hack into an individual's computer. The implications for the decision, if upheld, are staggering," writes the EFF. "Law enforcement would be free to remotely search and seize information from your computer without a warrant, without probably cause, or without any suspicion at all. To say the least, the decision is bad news for privacy. But it's also incorrect as a matter of law, and we expect there is little chance that it would hold up on appeal."

Leo: Oh, that's interesting.

Steve: Yes.

Leo: The broken blind defense isn't necessarily apropos.

Steve: Yes. It doesn't - I think that's where the judge went too far is to say that experience demonstrates that computers are not secure. Therefore it's like saying all blinds are broken.

Leo: Your blinds are broken; right.

Steve: And it's like, eh, come on.

Leo: I mean, there's even a larger issue about the FBI running a child porn site for any length of time. I mean, maybe that's legal, but that's creepy as hell.

Steve: Well, certainly it's creepy. I guess, I mean, someone I guess would argue, and I

don't know the law, but where does entrapment begin and end? I don't know where that boundary is.

Leo: Well, it's not like they set up the site themselves.

Steve: Correct, correct.

Leo: But they kept it running, which I think is kind of creepy. I mean, again, it may well be legal. I'm not saying [crosstalk]. I just think it's creepy. They ran it for months.

Steve: Yeah, they ran it for a long time.

Leo: For months.

Steve: In order to snare the people who were going, which no one is going to say that's not a good thing.

Leo: I was on a jury trial that was thrown out halfway through for entrapment. It was one of those Dateline "To Catch a Predator" cases.

Steve: Right.

Leo: And the prosecution got to do its whole thing. Then the defense moved, "Your Honor, this is entrapment." And the judge said, "Yeah, you're absolutely right," threw the whole thing out, and we all went home. Because, I mean, but this is - that was different. They really did set the whole thing up. The guy was never chatting with a teenager. He was always chatting with an adult.

Steve: Well, didn't they - I thought that they were able to get some convictions.

Leo: I'm sure they did.

Steve: [Crosstalk] different court?

Leo: Yeah, different courts.

Steve: Because it would seem to me that - I remember for a while MSNBC was running that show.

Leo: Oh, horrible, horrible.

Steve: And it was sort of really very creepy.

Leo: Yeah.

Steve: But they were all sort of this - they were variations on exactly the same theme. There were no underage people involved, so they were always getting people acting underage. And so I would think, if it's entrapment for one, it's entrapment for all.

Leo: And that's why you and I aren't judges or lawyers, my friend.

Steve: Yeah. So we got good news, but it's going to take some action on the part of our listeners. And that is that - and this was Woody Leonard who had the best coverage. He's been writing for InfoWorld forever, "Woody on Windows."

Leo: Yeah. Wow.

Steve: He's been in the industry also forever.

Leo: Good, good, good.

Steve: And when I turned my Windows 7 machine on today, one update, it always wants to try to give me Silverlight. And it's like, no. And it's so funny because I keep marking it no and don't ever ask me again, and Microsoft doesn't care.

Leo: You know what's funny, they don't even put Silverlight on Windows 10. They don't like it. They deprecated it years ago. It's crazy. Crazy.

Steve: So the other one, which was optional, is 3161608. That finally fixes the slow update problem. Woody's article, and I have the link to it in the show notes if anyone's interested, but you can probably find it if you just maybe google "Woody on Windows." And I'm sure it's a recent column. He's talking about two different patches. One's actually a knowledge base - yeah, there it is. And it's this 3161608. It was suggested as optional. And it's funny because I'd just put the show notes together, and I thought, wait a minute, 3161608? That sounds really familiar. And that's the one.

So it's a rollup of a bunch of other updates. Woody goes into it in detail, and it will just make your eyes cross because it's all kinds of interactions of five or six different things, and one's going in one direction, and one's going in the other. And so finally you just get down to, okay, 3161608. I want that. And what Woody's been reporting, because of course he's been feeling this, and remember Paul's story about trying to install Windows 7, basically it ate his entire weekend because hours go by.

Leo: Oh, that's right, yeah.

Steve: Even days, with it just sitting there saying, updating Windows, while no network activity, no CPU usage. It's like, what is going on? So it turns out that there was a problem in the win32k.sys DLL, or device driver, that this fixes. The problem is, if you've got Windows set not to install optional updates, you won't get this. So people who are on Windows 7 - and it's an out-of-cycle update. This wasn't last Tuesday or Tuesday before. I don't know what day it is. Anyway, here we are at the end of the month, this is the 28th, so this wasn't a Patch Tuesday. This is something that just - and Woody mentions it's just coming out now. But my machine got it, and I did not have to go ask for it.

And so we'll find out on the second Tuesday of July - July 4th is Monday, so the 5th is the first Tuesday, so I guess it'll be the 12th - whether this has been fixed. So everybody who's got Windows 7 and has been having trouble with these slow updates, the good news is it's fixed, but you've got to go get it at the moment. Maybe Microsoft will move it over into recommended or important or whatever at some point. But at the moment they need it.

Okay. Comodo, who is always in the doghouse, has put themselves there again. Of course they were the Superfish people and so on. And there's just no excuse for what they tried to do. Their CEO finally went into their own forums to try to defend their behavior. They have tried to acquire the Let's Encrypt trademark.

Leo: What?

Steve: Yes.

Leo: Jerky jerks.

Steve: It's unbelievable. So Let's Encrypt posted on their blog: "Some months ago, it came to our attention that Comodo Group, Inc. is attempting to register at least three trademarks for the term 'Let's Encrypt,' for a variety of certificate authority-related services. These trademark applications were filed long after the Internet Security Research Group" - the ISRG that we've talked about often are the group that did Let's Encrypt - "started using the name Let's Encrypt publicly in November of 2014" - which is when we started talking about it because I have always been so excited about this being the solution - "and despite the fact," writes Let's Encrypt, "the fact that Comodo's 'intent to use' trademark filings acknowledge that it has never used 'Let's Encrypt' as a brand."

So they attempted to use their size, essentially, their clout to simply trademark, to get trademarks on Let's Encrypt. The Let's Encrypt folks continue: "We have forged relationships with millions of websites and users under the name Let's Encrypt, furthering our mission to make encryption free, easy, and accessible to everyone. We've also worked hard to build our unique identity within the community and to make that identity a reliable indicator of quality." Which of course, if Comodo had it, would be blown. "We take it very seriously when we see the potential for our users to be confused, or worse, the potential for a third party to damage the trust our users have placed in us by intentionally creating such confusion. By attempting to register trademarks for our name, Comodo is actively attempting to do just that."

And there was such backlash in Comodo's own forums, I read some of these postings by the CEO, who I just couldn't believe it. He was claiming that Let's Encrypt stole their idea. And it's like, what? No, they didn't. And, like, we came up with 90-day certificates, and these people have stolen it, so we're just going to steal their name. And it's like, what? Anyway, the good news is all of this happens now on the Internet, and it's impossible for this to happen without everyone knowing. There was a such an outcry and backlash that they then abandoned this effort. And in an update to Let's Encrypt's posting, they have notified the industry that Comodo has dropped their effort to obtain Let's Encrypt's trademarks. Unbelievable.

So again, Comodo, no. If you want a certificate, get it from DigiCert. If you want a domain name, get it from Hover. DigiCert is, like, it's just everybody that I've recommended has come back and said, wow, these guys are great. And of course I've had the same feedback from Hover. And I thank our listeners. Many of our listeners suggested that when I was deciding to move away from Network Solutions finally, that I switch to Hover. And if I didn't make sure - I want to make sure everyone knows that it is standard in the industry, which I wasn't aware, that any remaining time that you have at a different registrar is honored by the receiving registrar of a domain transfer. So there's just no reason not to move. And Hover continues to be a great experience for me.

The Sophos guys do their Naked Security Blog, and what has come to their attention is that there's a new approach that malware is taking which is quite worrisome. I mean, it's not necessarily more potent, but it's maybe potentially so. And that is that in 2015, where they have a full year's worth of research, it turns out that even now, in 2015, Word macros, that is, Microsoft Word macros are, and were in 2015, the number one vector of infection. So it was phishing mail and email coming to people with attached DOC and DOCX files, which were getting into people's machines.

Now, they note that Word macros are now blessedly disabled by default. But the document can sense that and tell people, oh, look, you've got Word macros disabled. This document requires them to operate. Please turn them on. And so somebody who doesn't know any better will go, oh. Oh, and it's like, if what's below looks scrambled, it's because you need the macros turned on. And so there's a scrambled document being shown. The person then figures out how to turn macros on, and the virus launches in their machine. So these are typically JavaScript macros.

Now, the problem, of course, is that by default, as we know, Windows hides file extensions. And the trick is that you name the file "invoice.txt.js." So Windows strips off only the final .js, showing the unwitting user invoice.txt. The other problem is that the icon for scripting is kind of that yellow scroll-y parchment-looking thing and so kind of looks like a text. So it doesn't, the icon doesn't make it look like it's non-textual and it's going to run code because users don't know what scripting is. And so Microsoft designed an icon for scripting that's ambiguous, at best. So people end up running this, not knowing any better.

What's changed now - now, normally what Sophos has seen is that the JavaScript simply fetches an executable from a remote website and sucks it in and runs it. The problem is, now that perimeter defenses are getting better, script-pulling EXEs from a remote location, or even if the file is renamed, because of course it could be named something else, and once it arrives, the script itself could rename it to an EXE and then invoke it in order to run it. But deep packet inspection that we now have increasingly on proxy connection boundaries in corporate Intranets, they're looking into the file, and they will see that this is a renamed EXE and prevent it.

The problem with JavaScript is it's just text. And so what has happened is we are now

seeing 100% pure JavaScript crypto ransomware. So as we know, the whole cryptoware is the new scourge because it encrypts people's files that they need and then demands ransom in order to decrypt it. There is publicly available, high-level, full RSA public key and AES symmetric key crypto libraries in JavaScript which are beautifully written. The Stanford JS security library is one of my favorites. And so these guys don't even have to do this themselves. They can grab these modules. The crypto is as good as any that you can find anywhere. No external EXE needs to cross the boundary. The entire crypto ransomware is built into the user's machine.

Now, the big problem is that this is executed by the Windows scripting host, WSH. And JavaScript, the .js file extension, the handler is the Windows Scripting House by default. So what I would recommend our listeners do for their own safety, and also especially for their family members and nontechnical friends who don't know any better, how often are you deliberately running scripts by double-clicking on an icon? That's so rare and so dangerous, it should not be the default. But all of Microsoft's defaults are for the first 10 years, until they finally get a clue and decide, oh, look, everyone's getting infected this way, and no one's actually using what we have, so let's turn that off by default, like they finally did with Windows, where they've got Windows macros disabled.

So the first thing I would recommend people do, certainly the first thing I do when I'm sitting down at a new Windows install, is I turn off "hide extensions for known file types." I just can't imagine seeing only the first names of files. It's just part of - I'm sure that that's the case for our listener base, too, is you look at the type of file that you're dealing with.

The second thing is you want to change the association, the file association, so that it opens with Notepad, which is safe, rather than the Windows Scripting Host, which is not. Which is easy to do. You just name a file hello.js and then - create a new text file, rename it .js, then right-click on it and choose "open with," and it'll default to Windows Scripting Host. Instead, go down and find Notepad.exe, or put it in if it's not made available to you, and then choose the "always use this app to open .js files." That will change the association away from the dangerous Windows Scripting Host and just not run JavaScript. If anything tries to execute the JavaScript, it'll just pop up in Notepad rather than execute.

And, boy, in this day and age, that's the way Windows systems should be configured. Especially now that it's no longer necessary to pull executable content in from the outside. What they were finding, what the bad guys were finding was their EXEs weren't getting fetched by the script that was running because the border defenses have gotten smart enough now to keep that from coming through. So now it's just pure text. And that can bite you just as badly as traditional executable content. Okay, Leo.

Leo: Uh-oh.

Steve: Yet another way to exfiltrate data from air-gapped computers. We've talked about all of the crazy emanations that computers generate. Back in the day, you used to put an AM radio on top of an IBM 1401 and play Christmas carols by changing the length of the loops in order so that AM radios set in between stations would pick up different audio frequencies as essentially a heterodyne of the carrier that the big computer was emitting. And of course all of our machines are shielded and insulated from generating radio frequency interference. But we've talked about how even a cable extending out of the machine creates an antenna that allows people to obtain information. And then there's the screen, where you're able, just from, again, from the leakage of RF, radio

frequency information, through a wall, the image on the screen can be recovered without seeing it, just based on the emanations.

So some researchers have gone to the next, well, I don't know if this is the next step. This is really - this is, like, obvious in retrospect, but not something we probably need to worry about too much. And that is they ask themselves what else does software have control over in our computers? Well, software can control the fan speed. And believe it or not...

Leo: Oh, no.

Steve: ...just sort of for some giggles, as they say, these guys thought, let's do the math. Let's make it happen. So they analyzed the spectrum of sound generated by a PC case's fans. And they wanted to find a set of frequencies where it wouldn't be obvious to someone sitting there that the fan speed was changing. So they settled on 4100 rpm and 4500 rpm, just moving the fan back and forth, or fans back and forth between those two speeds. And it turns out that it generates a significantly distinctive frequency that a spectrum analysis of the sound made by a smartphone four feet away is able to differentiate. Now, because of inertia, it takes a while for the fan to get up to speed or down to speed. So the best these guys were able to do is 10 bits, 10 binary bits per minute.

Leo: That's like the Mars Rover.

Steve: Yes, it's slow Morse code. Actually it's slower than Morse code, right, it's very slow. That would be 600 bits per hour, which for a full day would be 14.4 kbits per day. Now, of course, old modems were 14.4 kbits per second. So this is painfully slow. On the other hand, it works. So imagine a scenario where, I mean, you can sort of imagine maybe a system where - well, now, you would have to first infect this machine. You'd have to want data out of it that you can get no other way. So it's infected in some bizarre means where the infection is one way, yet you will never then - it has no network connection, no WiFi or wired connection. It is air-gapped. Yet you have some facility that is hi-fi audio and a means of listening to this thing. And you can export 14.4 kilobits of that machine's private internal data per day.

Leo: Well, if it were just a PGP key or something, that'd be enough.

Steve: Yeah. I mean, that's a good point. There are, for example - that's a very good point, Leo. If, for example, it was protected with TrueCrypt or VeraCrypt or any of the encryption tools, as we know, while those drives are mounted, the key must be in memory in order to drive the symmetric encryption and decryption. So the weakness of those is anything that's able to access the process running the crypto could find the key. And we've seen those exploits. And exactly as you say, those keys are - they're, what, typically 256 bits. So a couple hours.

Leo: Yeah. In a related story, a hacker has modified his floppy disk drives to play the "Game of Thrones" theme [plays media]. Just thought you'd enjoy that musical

interlude.

Steve: Yes, I love those big arrays of floppy drives doing various pieces of music.

Leo: That's more than 14.4 kilobits a day, let me tell you something.

Steve: That's a lot of data. Okay. So today's bit.ly link, bit.ly/sn-566. That will take you to Netgear's knowledge base page, where you can look up the model number of Netgear router. Because here's the problem. This is not a showstopper, but this is a good reason to get on the ball and update your firmware. What's been found is that there is a worrisome vulnerability in Netgear routers such that - and Netgear does not go into detail, and I didn't bother to dig because it would have been academically interesting, but not crucial. If you don't have password recovery enabled, apparently there's a problem. If anyone can get to your router, they can get your password. So that would mean anyone on your LAN that is inside the network can obtain your password through some fault in the firmware.

Or, even if you have a strong username and password, and you're counting on that to protect your web facing, that is, your WAN-facing interface, if you have remote administration enabled by default, that could be cut through right now. Now, it's disabled by default. So someone would have had to deliberately turn on remote WAN admin. But the problem is that, of course, exposes a server, probably a web server, running in the router to the Internet. Now, a user might go, okay, but I put in a really good username and password. What this means is that won't help you. Anybody can obtain that and get into your network from the outside.

The good news is most people are not going to have that enabled. I mean, it's horrifying to think that that would be enabled. But what's weird is that - so Netgear's workaround is, if you enable password recovery, which is sort of strange, I mean, if you forget the password of your router, you just do a factory reset, and then all is forgiven, and then you reconfigure it; right? But there is a password recovery feature. And when you turn it on, it gives you the two fields of previously prepared questions about your mother's maiden name and your first pet's name and the middle name of your oldest sister and so forth.

So the idea would be, some people would not want to do a factory reset if they forgot their username and password. They would rather tell the router the answer to the questions and then have it ask them if they need password recovery. Okay. So the point is, if you turn that on, dumb as that is, that's a short-term workaround until Netgear gets the firmware updated to address the problem. And believe me, the list of routers is extensive. It's all of the WNDR routers, I mean, it's like, it's not just old creaky routers. It looks like the Who's Who of Netgear routers.

Which is why I made a bit.ly link, bit.ly/sn-566. That will allow you to quickly see whether your router's on the list. You can then - there's a link there where you can sign up to receive email from Netgear as soon as they've got a firmware update. It's my.netgear.com/register. So that's sort of generic, register with Netgear as an owner of a Netgear router. I don't know if they do anything specific when you go there. But Netgear wrote: "Netgear is aware of the security issue that can expose web GUI login passwords while the password recovery feature is disabled." And by the way, it's disabled by default.

"This vulnerability occurs when an attacker can access the internal network or when remote management is enabled on the router." So any contact with the router allows someone to defeat your username and password, no matter how clever it is, if password recovery is disabled. So in the short term, if your router is one that is affected, turn on password recovery and just put gibberish in the Q&A fields there, and you'll be okay until Netgear is able to address the vulnerability.

Okay, Leo. Now there's not much conversation in this YouTube video. So it would work wonderfully if we had a high viewership of this podcast, but we know that it's predominantly audio.

Leo: You can narrate it.

Steve: This is the coolest idea of the year.

Leo: A car with a person. And our voices are transmitted through speakers. Problem is, anyone can hear the transmitted sound.

CLIP: Users of smart watches make calls on speaker phone, meaning they have almost no privacy.

Leo: Oh, this is our friend.

CLIP: Your phone rings, shattering the silence.

Leo: That's [C.P. Grey].

CLIP: But it's deep inside your bag, and you frantically fumble through your belongings. Then how can we solve the aforementioned problems? [Buzzing] Finally, we found the answer, taking cues from vibration.

Steve: It's so good.

CLIP: TipTalk is a new UX. It's a product of watch strap type, so you can change the strap with any existing watch. Of course, you can also accessorize with it as a band. The smart strap has Bluetooth and BCU vibrator inside.

Leo: This is like robot voice.

CLIP: When you receive a call, your smart phone ends a signal to TipTalk via Bluetooth. TipTalk vibrates the BCU, and it transmits sounds through fingertips, not through the app. Users touch their ears with their fingertips and hear the sound. TipTalk can read your messages in private.

Steve: Is that too cool? I just love that.

Leo: Well, if it worked, it'd be cool.

Steve: Well, watch the people's expression.

Leo: They're going, "What is he saying?"

Steve: When they try it.

Leo: So he's putting it - now, this is real people. Of course, we're not hearing what they're saying. They could be saying, "What the hell is that? Oh, that tickles. What did he say? Oh, my. Huh? What?"

Steve: Anyway, I just love the idea.

Leo: Well, would you like to contribute to a Kickstarter fund?

Steve: Hey, I got a great coffee mug. I got a great thermal insulating coffee mug, and it only took three years to get it.

Leo: I've got, let's see, I've got a little Kickstarter project starting up here for this watch.

Steve: Anyway, I just love it, the idea. So for our listeners...

Leo: It was at CES. We missed it somehow.

Steve: Just the idea that you just - you touch your ear, and this thing vibrates your finger, and you hear the audio. Oh, there it is.

Leo: This is at CES, and this is XDA developers. So they're smart. They know what...

CLIP: Not that many people have gone into the smart wristband.

Leo: But why, you know, this was announced in January. Why don't we have it yet, if it's so cool and all that?

Steve: I figured you'd be first in line, Leo.

Leo: I have learned. I'm still waiting for my floating bonsai tree.

Steve: Sure.

Leo: And my Temperfect Mug, by the way. Sounds like they made just enough to get Steve Gibson off their ass.

Steve: Okay. So a couple quick bits of miscellany.

Leo: Yes.

Steve: I have an incredibly exciting announcement over on the Healthy Sleep Formula front. I put up the news on the page last week - or, I mean, yesterday - that I would have something. Over the Fourth of July weekend I will bring it up. Essentially I have what I think is a breakthrough, reducing the formula to two tablets which are far more effective than anything we had before. So I just wanted to give our listeners a heads-up. That's all going to go away. And anyway, I've been testing this new approach for a couple days. I've never had reliability this high. What I had didn't work for some people. I think this may be a home run. So I just wanted to point people there. I will have additional information up a week from now.

And Leo, I heard you mention on MacBreak Weekly that you had not seen the last Terminator, the most recent Terminator.

Leo: "Terminator Genisys," right?

Steve: So I just wanted to say it was a ton of fun.

Leo: Ah.

Steve: It was really - it was faithful to all the mythology. It used some really wonderfully tangled time travel-y stuff. And everybody loves time travel paradoxing. And so it had a lot of that. I just - I loved it.

Leo: Good. I'll watch it.

Steve: So I did want to - I wanted to recommend it. I can commend it without reservation. A lot of good computer-generated special effects.

Leo: Yeah, they make Arnold look young.

Steve: It was a great movie.

Leo: Which is quite an effect. And Emilia Clarke's in that. That's why we were talking about it. The Mother of Dragons from "Game of Thrones" is Sarah Connor.

Steve: There will only always be...

Leo: One Sarah Connor.

Steve: ...one Sarah Connor. And not her.

Leo: But that's okay. I can do suspension of disbelief.

Steve: It was a great movie.

Leo: Good, good. I'll watch it.

Steve: And I got a nice note from a Steve Reed, who's in Rockville, Maryland. And he ended actually a much longer note by saying: "Thanks for the time you put into the podcast and all the other great products you make. I have our company halfway to a site license and expect to get the other two very soon."

Leo: What is it, $45? I don't understand. What, he's passing the hat?

Steve: Well, yeah. So he said, every time SpinRite fixes a drive, we buy another license.

Leo: Oh.

Steve: And so, okay. So here's the deal. And this is sort of interesting. I never thought of sort of this incremental approach before. But my license agreement, I mean, all of our listeners know that I'm very understanding about the SpinRite license.

Leo: Generous would be the word.

Steve: From the beginning, I always felt it was ridiculous and unreasonable to think that somebody would buy a license per drive. But what's interesting is, if you look at any other hard drive utility or backup or mass storage or anything, they all say you can only use this on one single machine, or on one hard drive. And I remember getting into a fight with my company when I had 30-some employees because they wanted to go the same route as everyone else. And I said no, come on. If some person buys this, I'm not telling them they can't run it on, like fix all of their drives, or maintain all of them.

Now, admittedly, back then, drives were $10,000 or $5,000. I mean, most people had floppies back when SpinRite was being born in the late '80s. But still, this has always been the case. And we all know that I don't even mind if you fix your friends' or family's computers with it. I mean, it's paying the bills, and it's made a great living for me, and I'm happy to do that.

For corporations, though, it seemed like, you know, if they've got a big building full of computers, maybe $89 is not enough to ask for. But I also kind of wanted to allow people to try it, verify that it works, and then somehow sort of move up to a license, a site license. And so I just thought, okay, four copies of SpinRite. If a company buys four copies, then they have - and it's all on the honor system. There's no DRM. There's no protection. I won't know. That's the right way to do it. They buy four copies. Then, with our blessings, they can use it forever on all of the drives they have in that location.

We do have what we call an enterprise license, which is 10; in which case, like IBM has an enterprise license. Actually, they have an international license, which is 20 copies. So IBM gets to use it in all of their offices everywhere on Earth.

Leo: Wow, on like their 800,000 machines.

Steve: Yeah.

Leo: You're very generous, Steve.

Steve: Very generous, yeah. So I'm making less per, but that's fine. So what Steve has done, he's like, okay. And I think this is like a neat compromise, is SpinRite fixes a drive, they buy a license. So it's recovered two drives so far. And he says, as two more fail, then he will each time buy another one. They'll get up to four. Then they have a site license. And, Steve, stop buying them at that point because then you're fully site-licensed.

So that's sort of a cool - and that's why I did it was to allow people - if I had like a site license, and then a SpinRite license, people would say, well, but we already got SpinRite license. Now we want a site license. It was like, oh. So, like, what, we had the discounted site license for people who already have a single-copy license and all that. And I said, no, no, no. Let's just - I'll make it one copy for end-users, four copies for corporations that want to use it at will across all of their machines. And then, of course, if it's a multisite license, that's 10. And if it's an international multisite, then it's 20. And it allows people to move forward.

And my expectation in the future was that at some point, once the 6.1, 2, 3 series is behind me, I'll be working on 7. That will be a paid upgrade, whereas all the 6-point releases are free, as we know. And then the idea would be that people would simply upgrade how many copies they have on whatever license they have in order to move to 7. So the whole thing sort of fits together. It's weird, but it sort of made sense to me. So that's the plan.

Leo: And that's why Steve is Steve. Leo Laporte with Mr. Steven "Tiberius" Gibson.

And we go to Question #1 from Simon C. in Lymington, United Kingdom, who was paying close attention and thinking about the ENDBRANCH instruction. Was this from last week?

Steve: That was last week, yup.

Leo: Did you talk assembly language when I wasn't here?

Steve: Oh, we did some deep assembly language.

Leo: What'sa matta you? Oh, man. Now, that I would have enjoyed.

Steve: Leo, you know it's been recorded.

Leo: Oh, I could listen to it, couldn't I. What do I do, I google "adobe marketing cloud" and "ENDBRANCH," and then I'll find it. He says: I was listening to SN-565, where you were describing how the ENDBRANCH instruction would limit an attacker to calling complete functions instead of small snippets as the destination of a call has to be an ENDBRANCH. But an attacker would actually have a little more freedom than that because the control flow within a function would cause there to be an ENDBRANCH at the start of any basic block targeted by a jump. An attacker could jump to one of these and execute a subset of the function, presumably some part of an otherwise valid control flow. Of course, this makes it a lot harder to find useful code to execute, as the granularity of jumping is now basic block rather than instruction or even sub-instruction level. Make sense?

Steve: So I deliberately omitted a detail because we already, as everyone who survived last week's episode knows, there was plenty of detail as it was. But I appreciated Simon's note because he had his pencil sharpened, and he was paying attention.

Leo: Paying attention, yeah.

Steve: And it is only indirect calls and jumps that are subject to this ENDBRANCH instruction as their landing pad. So the way Intel instruction set works, conditional branches and jumps within subroutines are fixed offset jumps. They are not indirect jumps, that is, driven by data, but they're compiled in with a jump forward 32 bytes, or jump backwards 526 bytes. So they have assigned offset that advances or retards the instruction pointer.

So anyway, as a consequence, that is exactly what Intel was thinking. That's why they only made it for long indirect calls and jumps so that the innards of subroutines are not peppered with ENDBRANCH instructions, only the sanctioned entry point to the function which is being jumped to through a ring transition or from another process or another module where dynamic linking has created indirect linkages between these pieces of the system. So you're right on the money with your thinking, Simon. But it turns out I didn't

add that complexity because people's heads were exploding as it was. So good on you.

Leo: Wow. I'm listening to this episode. Fascinating.

Steve: It was neat.

Leo: Mike Woolard, Cleveland, Ohio wonders why SQRL even needs a button to log on: I was talking to a coworker about SQRL. He said, well, wait a minute. Does it even need a button to initiate the logon process? Wouldn't a user just go to a site, the site senses if SQRL is installed on the machine and logs you in if possible? No SQRL: Show a marketing page saying, hey, if you had SQRL you'd be home by now. Authorized SQRL user: Proceed immediately to authorized content.

Steve: If you had SQRL, your nuts would be warm.

Leo: Yes. They'd be buried by now.

Steve: Okay. So this is a great point. And I see questions, all kinds of variations on this. So I just sort of wanted to take a second to note the current embodiment of the concept of a per-domain unique public key that the site has and uses to verify a real-time challenge. That's the kernel of the SQRL nut. And so, for example, the SQRL client for Windows that we'll be playing with soon, or even Jeff's client for iOS, these are sort of like they're working implementations, but only one way of applying the technology. As I mentioned before, Stina of Yubico has indicated a strong interest and the ability to move the most critical crypto into one of their little dongles, which would be very cool. But exactly as Mike suggests, or Mike's coworker, I wouldn't be surprised if we see SQRL clients as browser plugins, and maybe eventually built into browsers, where they could absolutely do the same thing on behalf of users.

So I guess the point I wanted to make was that there is sort of a very clear, well-defined technology, and then lots of ways to skin the cat. Lots of ways to wrap that, with or without passwords, using biometrics. Or, if you wanted to, making it completely transparent. Now, understand that all of these have a tradeoff that we're always talking about, the security versus convenience, because if you simply go to a site, logs you in, then how does anything know it's you sitting at your computer?

So the reason, for example, that my Windows client has a SQRL password with that password hint technology is to create a compromise. It doesn't need it at all. But it feels like we've delegated to the SQRL system the ability to represent us to the Internet. So we need a way to prove to it, who is basically our identity proxy, that it's us sitting at the keyboard. It's us using it to log into a site. Otherwise anybody could come along and log in as you, which would not be good. So anyway, it's absolutely the case that the technology, the core technology is independent of its particular use case. And I have no doubt that, if this thing gains traction, we'll see all kinds of it being used in a compatible way in all kinds of other packaging.

Leo: And of course somebody was asking in the chatroom, and where can I get a

SQRL sticker just like that on your mic flag? Are you offering SQRL stickers?

Steve: The logo is in high-resolution downloadable form at GRC. So I've got a big...

Leo: Make your own.

Steve: I've got vector format and bitmap format, very large. So you just grab it, scale it to an inch and a half, and print it on your color printer, and you've got one.

Leo: M'kay. Sean Jones, Houston, Texas says: How hard are firewalls? Oh, look at that. It's giant.

Steve: It was funny, too, because last week I had it back on the bookshelf, and it ended up being right next to this one, and they were exactly the same size.

Leo: Parallax. It's an amazing thing.

Steve: It was strange-looking. And the depth of field on this camera is so great that it was in focus back there. And it's like, wow, that's really weird-looking.

Leo: I saw it yesterday, I mean last week. Sean Jones, Houston, Texas, wants to know: Steve, I've been talking with my cuz about security appliances, and he pretty much hates the company that I use for UTMs, which is Sophos. He complains that their UTMs are just "software firewalls" and that "hardware firewalls" are better. I really don't understand what he means by this because all firewalls/UTMs have to have some form of software on them to work with the hardware. Can you explain to me what he probably means by this? And say this carefully, by the way, because the new studio is all Sophos firewalls. And I understand it's similar to pfSense, which is what you use; right? Which is software.

Steve: Oh, I wanted to - I'm so glad you just said that because I wanted to just tell people. As everyone knows, when I lost my two T1 trunks and switched to cable modem, I did some replumbing. I switched to gig switches. And among the other things I did is I built a cute little - I took a little embedded PC from - I'm blanking on the name. It's an engineering company.

Leo: Oh, I know you love them, too.

Steve: Soekris. Soekris. Although, frankly, the pfSense people have their own hardware. And I think, had I known about that at the time, I would have chosen that over the Soekris engineering. I love this little router. So it's pfSense is free. Install it in a PC. It just needs a few network interfaces. You can use the pfSense hardware. But, boy.

One of the things I will be talking about in the future is an ARP scan of a LAN in order to find all the crap that's sitting there that you've forgotten about. And anything on the LAN, no matter what it is, will show up. And so this morning I thought, I wonder what pfSense has for showing the ARP table? And I did an ARP-A on my Windows machine, and I only got two entries. So I fired up the pfSense page, brought up the ARP table, and here was a listing of every single device on my LAN, and all the MAC addresses, and all the IPs they have, because of course they're all talking to the gateway, and it's recording all of the MAC addresses in order to forward Ethernet packets to them.

So I just wanted to say that pfSense and I are having a ball together. I've done things like - I'm mapping Skype through so that we get a point-to-point connection. I have port shifting, where because my Cox blocks a bunch of ports, some that I want to use to get to Level 3, so I shift the port to a different number at this end and then shift it back at Level 3 so it bypasses Cox's filter, yet the devices at each end here and there both see traffic on the port that Cox is blocking. And pfSense does all that. I mean, it is just - it's incredible.

And in fact one of the things I need to add to the SQRL client is support for proxies because it doesn't handle web proxying right now. And some corporate users who have wanted to beta test it, or alpha really, were saying, hey, I can't get SQRL to work inside our corporate network. It's because they've got a proxy. So I thought, huh. This was a couple days ago. I wonder, I'm sure pfSense must support proxies.

And so it turns out it's got a whole library of downloadable modules. It supports the Squid proxy. I clicked a couple buttons. It downloaded a latest update, installed it, didn't reboot or anything. It just said, okay, now you have a web proxy. So I'll be able to use that, once I add awareness of that in the SQRL client, I'll add that to the client and then test it using my own little router. So I'm just, you know, it's free. And, boy, there's nothing that I have found that it won't do. And for somebody who really enjoys having control of their network, this is what I would put on the border.

Leo: Well, and to the question, I mean, that's a software router; right?

Steve: Yes. Now, okay. Sean's question about his friend, that is a distinction without a difference. When I was first setting things up and didn't know any better, I bought a, quote, "hardware firewall." It was a red box meant to look all like a fire department or like a firewall. And, ah, can't quite remember the name of it. I had it on my mind this morning because I knew I was going to be talking about this. But it had three NICs with a DMZ and a WAN and a LAN. Fire - no, I can't remember. Anyway, but when I plug a console into it and turn it on, it's booting open source software.

So Sean is exactly right. I mean, technically, a hardware firewall, I would define that as pure hardware. That is, like an FPGA, where there's no OS; there's no programmable. Like it's receiving packets and disassembling them and interpreting them and checking them and operating against hardware-based rules. You could do that, but you'd have to have some strange reason. Maybe - no. I was going to say maybe wire speed performance, but we've even got firewalls running at wire speed. So I just...

Leo: And look what you'd be giving up. I mean, you'd be giving up all the...

Steve: There isn't such a thing.

Leo: ...configurability and patches.

Steve: Yeah, there isn't such a thing as a hardware firewall.

Leo: And it'd have to be perfect out of the box because you couldn't update it.

Steve: Yeah, yeah. So technically, yes, you could block traffic in hardware. No one does. They don't - it doesn't exist. Everything is a software firewall. It may just look like it's not booting an OS, but underneath that's what it's doing.

Leo: There's always something in the firmware.

Steve: I mean, light bulbs are booting Linux.

Leo: Nowadays, yeah. If you get a 43-page GNU public license with your light bulb, you know. Actually, Sophos bought Astaro, which was our first sponsor on the network and on our show.

Steve: Ah, nice.

Leo: And I think that they also make Astaro's stuff. And remember the Astaro guys, one of the cool things about it is you could buy Astaro hardware, or you could download for free the Astaro software.

Steve: Yes. Yes.

Leo: And install it on a beige box.

Steve: And that's all they did.

Leo: That's all they did.

Steve: They just did it for you.

Leo: They just integrated with the hardware. And that's pretty much what everybody else is doing. Although I tell you the one exception. We're going to be getting - get ready for this - Symmetric 10Gb fiber into our new studio. And 10Gb.

Steve: From the outside?

Leo: Yeah.

Steve: Wow.

Leo: From Sonic.

Steve: Nice.

Leo: And one of the challenges of this is you do have to have a hardware switch that can handle 10Gb; right?

Steve: Yes.

Leo: I mean, it's not - you can't just plug in your PC and expect throughput of 10Gb. So we are getting a Sophos box that is rated for 10Gb. I think it's a $45,000 box. They're very expensive.

Steve: Yeah.

Leo: But so to that degree, there's a hardware throughput thing, and you would spend money for that hardware. But then on top of it is software; right?

Steve: Right.

Leo: Abe Sloan, Indianapolis, Indiana - hello, Abe - shares his company's anti-phishing training scheme. I wonder if it involves hazing in any way: Hi, Steve and Leo. I wanted to write in about something my company has started doing. I work for a local hospital network - oh, and we know what a big deal that is, right, because those hospitals that got phished and then ransomware? He says: I work in the IT department. The security department of IT has started sending out phishing emails - oh, this is smart.

Steve: Yup.

Leo: With a link in them. And if you click on it, you're sent to a page with mandatory training. Not a long story, but I thought it was something that you two would get a kick out of. Thanks for the great show. Ive been a loyal listener from the beginning.

Steve: I think that's so smart because there's just no way to get through to most users. They've got a job to do. They're harried. They attend some training where IT says, look, do not click on links in email, period. And they're, uh-huh, uh-huh, uh-huh. But, boy, I

tell you, send them some tests.

Leo: Get their attention, yeah.

Steve: That's smart. Because then they're going to get caught out. And quickly it'll be like, oh, who clicked the link? Oh, you know, John over there, and Shirley over there. And it's like...

Leo: Hah-hah. You clicked the link. Heh-heh.

Steve: Yeah. I think that's...

Leo: Good.

Steve: I mean, that's probably the only way to do it, if you really - and exactly as you said, Leo, hospital networks, we've seen them getting infected by ransomware. They just can't afford that. So I think the only way to enforce a policy is, like, there will be a test on this, and we're not going to tell you when.

Leo: I also think they should do training, global training on social engineering. Because this is really the big challenge is you've got customer service reps who are trained to serve the customer, even if the customer is a hacker. And it really - I think they need to be trained in that and phishing, for sure.

Steve: I have two of the most wonderfully skeptical co-workers. And they just - Sue, you just can't pull anything over on her. And it's like, okay, good.

Leo: No, you need that.

Steve: Because, you know, it's just - yeah.

Leo: Terry Richard, writing from Toronto, Ontario, Canada...

Steve: This is one for you, Leo.

Leo: ...wants to delay Windows 10, but still get it someday for free: Steve, you stated in your most recent podcast you never, ever, will ever use anything beyond Windows 7. That's my feeling, as well. However, July 29th the free upgrade runs out. I'm currently running three computers on Windows 7. The thought of upgrading to Windows 10 today makes me nauseous. Yikes. That's kind of strong. But the thought of having to pay for Windows 10 times three, for my three computers, when support

for Windows 7 ends on January 14, 2020, makes me positively ill. I'd be curious to know what thoughts you have on this conundrum Better to upgrade now, or pay to delay what seems almost inevitable?

Steve: And the reason I say this is for you is that I'm very sure, but I need you to corroborate, that we've heard, and I've heard you saying on other of your podcasts, that once a system has been upgraded to Windows 10, it then obtains a key for that, which then it always has; and that you are then able to back out from Windows 10 back to 8.1 or 7 and keep that upgrade right intact for the future.

Leo: Right.

Steve: What do we know about that, exactly?

Leo: Yeah. And this comes from Paul Thurrott.

Steve: Okay.

Leo: No less of a resource than Paul Thurrott, our own expert. And Microsoft calls this an "entitlement." The new licensing scheme in this particular case is called an entitlement. So your machine, not you, your machine gets the license for Windows 10. It gets an entitlement. So, yes, if you have an authorized copy of Windows 7 or 8.1 installed on your Windows machine, and between now and July 29th you take Microsoft upon that offer, and you upgrade to Windows 10, you go through the whole process.

Steve: If only for a day.

Leo: If only for a day. And then, must do this before you go on, check to see if the Windows 10 activation occurred, which it should, instantly. But just make sure. You right-click on My Computer, you get the properties, and it says Windows X, and you see - and look at My Computer, and you'll see that it's licensed. Once that's the case, you can do anything you want. You can wipe out Windows 10. Technically, you have a 31-day rollback period. So you could go to the recovery control panel and say...

Steve: Nauseous, Leo, nauseous.

Leo: Yeah, take me back. Take me back. Calgon take me away. You can go back to Windows 7. That does not always work, so I wouldn't rely on that.

Steve: I was going to say that what I would recommend is make an image of your system first.

Leo: Yes, exactly.

Steve: I want to say DriveImage, but there's one that I like now more. Image for Windows is my favorite.

Leo: Oh, a new one. You keep changing on me.

Steve: Yes. Image for Windows.

Leo: It was DriveSnapshot.de. Then it was DriveImage. And now it's Image for Windows.

Steve: Image for Windows.

Leo: Okay.

Steve: I think it's TeraSoft. And that guy does beautiful work, and I'm very impressed.

Leo: Terabyte Unlimited.

Steve: Terabyte, that's it. Unlimited, yes. Image for Windows. It also has...

Leo: And it's free? No. It's not free.

Steve: [Crosstalk] Linux. Sorry?

Leo: There's 30-day free trial, though. That would be enough to use it, I guess.

Steve: Yeah, yeah. Although, I mean, anybody serious wants this in their toolkit. It's beautifully done. So make a snapshot of Windows 7 on the machine. Then do the upgrade. Make sure that Microsoft acknowledges that it's been activated under 10. Then back out using Microsoft-sanctioned rollback. And if anything happens along the way, that's okay because you've got an image, and you're able to restore the disk from the image, and then you're good for four years until you decide you want to take Microsoft up on it.

Leo: They have a lot of other cool-looking...

Steve: Yes, Image for Linux also.

Leo: Ahh.

Steve: And also for DOS. Yeah, every single one of their things, very reasonably priced, absolutely bulletproof. Multi file system format aware, so FAT32 and NTFS and UFS and EXT and the works.

Leo: Nice. Oh, by the way, and when you buy it for Windows, you get a copy of it for DOS and Linux, as well. So you know what, I'm going to buy this. I mean, it's only 39 bucks. That's a good deal.

Steve: I know. It's great software.

Leo: All right. Good recommendation. Yeah, make a backup. Don't trust the restore.

Steve: No, in fact, we know that the restore has failed.

Leo: Oh, yeah. It's failed on me. It's worked on me, and it's failed on me. And I think the whole process sounds kind of wonky anyway. Just make an image. You'll be glad you have it later. Somebody else is saying you also could just get an external drive or a second drive, do it on that, and then you'd have a copy of Windows 10 installed. Whatever. You know, there's lots of ways to do this.

Steve: I don't think so. I think you want to have - because we're not exactly sure. Microsoft's always been a little coy about how they lock onto the system. And one of the things they do, drives have serial numbers which are readable. And so there was like a - if you only change a few things at once, I mean, back when Microsoft began doing this, it created a huge upheaval. And it was like, wait a minute. What if I want to, if my system dies, and I bought Windows, but I have a new system, or what if I change my hard drive, or if I change the amount of RAM I have and blah blah blah. And so they lock onto it and plant some seeds, but also do things like look at all the serial numbers that they're able to find.

And so I'd upgrade on that machine and roll back, and just be glad if it does. And if not, just restore from the image. And Leo, given Microsoft's push, I could understand them stopping pushing at the end of July, but why wouldn't they always just make it, like, go here, click here to download, to upgrade? The user has to go get it rather than it being, quote, "offered," unquote, to them. I just can't imagine why they would ever say no because they're just in such a burn to get everyone moved over.

Leo: Right. Absolutely. But also don't...

Steve: Charging anything, especially after this, I mean, basically his sentiment makes total sense. I don't want it now. I want it someday. So why would Microsoft say no, you have to have it now, or we're going to charge you someday? That doesn't make any sense.

Leo: We spend a lot of time talking about that on Windows Weekly, what's going to happen July 29th. The consensus seems to be, no, they will start charging on July 30th. They will. So don't - but who knows. You know? It's all baffling to me.

Steve: Leo, I'm not going to do that. I'm going to stay with 7.

Leo: You're not going to bother. All right. You know, by the time, by what was it, 2020?

Steve: Yeah.

Leo: You'll be using BSD by then anyway. I guarantee it. I guarantee it. Actually, Linux looks better and better all the time to me. I just love it. I just love it. And maybe there's a hook here to Linux with Sebastiaan Langenberg, his email from Bergen op Zoom. I want to live in Bergen op Zoom.

Steve: That is a great place to live; isn't it? Oh.

Leo: Yeah. Oh, man, The Netherlands. Dutch sounds that way. It's like all shwoopy doopy language. It's a very shwoopy doopy language. And I mean that in the nicest way. I'm going to be in The Netherlands in September. Steve, during the time you spend with good Leo on the podcast, you talk about your love for assembler a lot. I am a more modern-day programmer, and I'm interested in the workflow of how one goes and creates his or her software. Therefore, I'm curious to learn what IDE and tools you are using on a day-to-day basis to work on projects like SpinRite and SQRL. I hope you find it interesting to share these details, and I hear my question answered on a future podcast. Yours, Sebastiaan.

Steve: So you know the...

Leo: Do you even know what an IDE is?

Steve: Heard of them, yeah.

Leo: This is the wrong guy to ask about IDEs.

Steve: So I think the main thing I wanted to convey is that I don't have any magic. That is - and I'll answer Sebastiaan's question quickly. But the point is my sense is it sort of doesn't matter. What matters is whether it works for you. So, for example, everyone knows I'm writing SpinRite and SQRL in Brief, which is in a DOS box in Windows, that used to run in DOS itself, where SpinRite was originally written. And I have a colorizing add-on that gives me syntax highlighting. So visually it works very well. Then, if I have a problem, and I write a lot of code that worries me because it just works the first time,

and it seems like I'm going to believe it more if I have to struggle to get it to work. But it just sort of works. So I don't really spend much time in debugging.

And also one of my approaches is iterative. So I'll write a little, and I'll test it. I'll write a little, and I'll test it. I'll write a little, and I'll test it, instead of just writing for days and then seeing if everything that I wrote works. I'm very incremental. And I test everything I do and try all the different cases, and then I move forward. So I'm constantly verifying that everything I've got so far works. If I need a debugger, I use the oldest one that makes sense when I'm on Windows, which is Visual Studio 6.

Leo: Oh, you do use an IDE.

Steve: Oh, yeah, yeah. I mean, so, but I don't...

Leo: So you're not writing in - I thought you were using MASM and Brief.

Steve: I am. But I don't code in that environment.

Leo: I get it. But you'll fire up the IDE for debugging, yeah. That makes sense.

Steve: Exactly. And that one will show my source code, and I can...

Leo: That understands MASM and the symbol tables and everything?

Steve: It doesn't, no. So there's...

Leo: That's some debugger you've got there.

Steve: ...the no-syntax highlighting. But it does understand symbols and the old linker.

Leo: Okay. So it will look at a symbol table and at least say it has the variables names and everything.

Steve: Yes. And I'm seeing my source. So because the map file has line numbers and so is able to link that back to my source. And so it's source code debugging. But, for example, in SQRL, in order to use the libsodium library that has the Bernstein elliptic curves, I was unable to compile that - I didn't really try too hard - on the old Visual Studio 6 C compiler. And they had it all set up for 2012 or something. So I needed to use - so I set up a Visual Studio 2012 and then compiled the libsodium library there. That wouldn't work with my old linker. So then I had to use a newer linker with all of my assembler code.

The point of all this, the reason I'm telling you, is that forced me to use a newer IDE. I'm

using Visual Studio 2008 was the oldest one I could use that was still aware of the new linkage. And as is typical, it's incredibly slow. Visual Studio 6 snaps up and runs like crazy. It's just beautiful. I fire up Visual Studio 2008, and I sit around. I can go make coffee. And windows are coming up, and it's populating the UI, and it's just - who knows what it's doing. But this is why I don't use new stuff. It's all slow and piggish, and it's just in the way. So, yes, I write in - oh, and for SpinRite what I do is that's actually still under DOS. And so I set up a DOS network with NetWare. I have NetWare drivers, and IPX and SPX.

Leo: Wait. Novell NetWare?

Steve: Yeah, yeah, because that runs under DOS.

Leo: What, token ring wouldn't work for you?

Steve: I have a DOS machine with a network share, so that I still write everything in my Windows machine and then compile. Then I switch over to the DOS machine and then launch SpinRite from a drive share on my Windows box, where it runs. And then for there I have SoftICE, which is the penultimate debugger for DOS systems.

Leo: I remember that. Oh, I remember SoftICE, yeah.

Steve: ICE stood for In-Circuit Emulator.

Leo: Yeah.

Steve: Once upon a time, you used in-circuit emulators. You'd actually take the processor off the motherboard and put this plug in with cables running out to this standalone box on the side. And that was called an ICE, an in-circuit emulator, which was the way you debugged things back then. And so SoftICE was the software equivalent of that, where you could, again, see the source, single-step through, and figure out what the code is doing. And so that's my debugging methodology under DOS. And Visual Studio 6 is what I use unless, for SQRL, I've been forced into a more recent and very much slower Windows-based IDE, which I'm only in if I'm trying to track down a problem. And the good news is I don't spend much time there.

Leo: Do you use any version control at all?

Steve: No.

Leo: No, of course not. That would be crazy. Why would you want to do that? No, and I like it, Steve. You represent the old school of programmers. You're like a cowboy.

Steve: And I wrote a little code that 1.365 million people have downloaded.

Leo: Nothing wrong with that.

Steve: You know? So the stuff works.

Leo: Do you do anything in C? I mean, is SpinRite...

Steve: Oh, yeah, I forgot to mention that, for example, the libsodium library at the time did not support the GCM AES encryption. GCM is the hybrid AEAD, the associated data encryption that does both - remember we've talked about you can't only encrypt. You must also authenticate. And so I was using Phillip's encryption from UC Davis. I'm blanking on his last name [Rogaway]. But the point was made, actually by Ralph, who was the developer of that early Android client for SQRL, he said: "You know, even if it's intellectually available, GCM AES makes a lot more sense." I agreed with him. I wrote a full implementation myself in C because that component needed to be portable. I needed to be able to offer it to everyone.

So on some of the SQRL pages is a link to my C implementation. I made it very portable. And so that's one of the components in SQRL that I deliberately wrote in C because that part I had to make shareable. Now the libsodium library includes it, so you don't even need that. It's like, okay, I guess I could have waited a little bit longer. But I needed it for myself anyway. So, yeah. I can write. I have a bunch of, like, all of the newsgroup stuff is in Perl. That crypto stuff is in C. I did a lot of customization of the Internet news server, because it's in C, for FreeBSD. And so, yeah, I sort of speak whatever language I need to to get the job done.

Leo: Nice.

Steve: But if I have a choice, assembler.

Leo: And that's "_get the job done," and that's what he does, and you can't knock that, I'll tell you. Someday we'll get you using something. No, why bother? Why even bother?

Steve: Well, I'm thinking that I may do...

Leo: I think you should learn Rust, GitHub, and Emax.

Steve: I'm thinking Python maybe. I'm thinking...

Leo: Python is good. Python's elegant. And, you know, one of the things that's cool, and you're seeing this, is you want to choose a language that has really good library

support.

Steve: Yeah.

Leo: Because those libraries are what makes it possible for you to go from zero to 60 pretty quickly; right? You don't have to reinvent the wheel constantly.

Steve: Yeah. The problem is that's where a lot of bugs come from.

Leo: I know. You'd have to trust these libraries. I know. I know.

Steve: Yeah. I'm thinking that what I want is platform transportability.

Leo: Well, then, Python does give you that, yeah.

Steve: Yes. And so, for example, it would be really - as soon as a little ARM board has a SATA interface on it, for example, then people could make a little SpinRite box that would just be like a Raspberry Pi.

Leo: Oh, yeah.

Steve: That would SpinRite a drive.

Leo: Plug it in and then take - yeah, that's a good idea. Standalone appliance.

Steve: But of course that would be ARM, and so I'm not going to write it in some RISC - that's where I draw the line. I'm not writing RISC code assembler. That's what compilers are for.

Leo: All the kids are doing Rust now. Rust is the new type safe C/C++. That's the one I would look at, if I were you. No, no interest at all; right? I try. I try. Steve Reed, Rockville, Maryland worries about HTTPS, corporate proxies, and personal passwords: One of the job labels assigned to me is to keep the IT equipment running at our office. One of my co-workers asked me a question; and the more I think about it, the more I question my own understanding of how the world works. Well, not really, but at least on this one particular small point.

To wit, his wife is having issues at her workplace - new director on a rampage, firing and hiring people, et cetera, et cetera, et cetera. She feels there are some strange things happening with her personal email and other accounts, Facebook and so on. He, my co-worker, was wondering if it was possible that their IT person was getting

their passwords for their personal accounts from the network. I initially told him, well, no, it's HTTPS traffic. It's all encrypted. I guess, he said, they could be using a keylogger or something like that, but it wouldn't be from the network traffic.

Then the old discussion of HTTPS proxies came to mind. Are the user passwords encrypted in the same method/channel as the regular traffic, or does it get an additional layer of protection? For instance, if I installed my own root cert in our active directory domain, as many large enterprises do, filtered Gmail through the HTTPS proxy, and looked at all of the decrypted traffic as it went through the proxy, would I see the passwords? I realize Gmail may be a bad example with cert pinning; but just as an overall issue, is that possible?

Thanks for the time you put into the podcast and all the other great products you make. I have our company - oh, this is the guy - halfway to a site license.

Steve: Oh, that's the same guy, that's right.

Leo: Expect to get the other two very soon. So that's a good question. Is it a second channel?

Steve: Would I see the passwords? The answer, yes. And this raises a point because - and this has not been raised, and I'm really glad he asked the question. And that is, I've capitulated to reality from my original stance of, oh my god, the idea of filtering and intercepting and decrypting and looking at HTTPS is so wrong, yet this is what corporations are going to do because...

Leo: They have to.

Steve: They have to have visibility into the traffic of their network. And the problem is all of the authentication has traditionally been relying only on HTTPS. Now, that could be done differently. Browsers are smart enough now. We were just talking about JavaScript and crypto libraries. Those exist. So it would be possible, for passwords and usernames, for there to be a challenge/response mechanism with passwords and usernames. The problem is authentication. We always come back to that. If you've got a man in the middle, you do not have authentication. And if you've broken authentication, there is no theoretical way to prevent that man in the middle from intercepting everything. There's no way to know that any handshake of any kind, no Diffie-Hellman anything, that you're not actually shaking hands with the proxy instead of the other end.

So this is really sad because what this means is the corporations that are filtering their users' traffic are in fact exposing what their users, I mean, are exposing everything, including their username and password authentication, which is maybe none of the corporation's business. Now, that said, it's the corporation's Intranet. It's the corporation's bandwidth. So if you don't want to do that, do it on your cell phone that's using cellular connectivity, or on a laptop that is avoiding the corporate bandwidth. I could argue that's their bandwidth to do with as they please.

But I did want to raise this point because I thought this was really good, that yes, these proxies, I mean, this is back to Firesheep, sitting there in Starbucks and seeing all of the

usernames and passwords and authentication tokens and everything. Nothing is hidden from that proxy. And if you have some bad apples in IT, they can see everything that is going in and out of every employee's connection. And that needs to be said.

Leo: And maybe you should stop using Facebook and collecting personal email on company time.

Steve: Right.

Leo: I mean, look. That's fine, and your boss may not even care. But you're sitting there, doing that, and you're giving them an entre. If you didn't do that, they wouldn't be able to see into it.

Steve: Yup.

Leo: Jay Clark, Boise, Idaho seeks the truth about some email security claims: I recently received some emails from financial services companies. There was a note in the body or subject that indicates it was "sent securely," implying that all contents are safe from prying eyes. The actual wording is: "Please be advised that communications with {SECURE MESSAGE} in the subject line have been sent using a secure messaging system."

This seems strange to me, as I know the email content is not inherently encrypted, and I believe when in transit the connection isn't encrypted, either. It doesn't seem like there would be any way for me to verify, from my end, the message was sent with a "secure messaging system," whatever that actually means.

So how can a company assert so boldly that the message was sent securely with a secure messaging system? Is that a red herring? I still don't trust that their email service is secure, and they want me to send financial documents to them using email. Thanks.

Steve: Yeah. Jay, you are 100% right. It's like the email that you get that says "scanned with a virus scanner, so open at will." It's like, what? Yeah, that's completely nonsense. Unless you apply your own end-to-end encryption, unless you encrypt it first, like with PHP or GPG or S/MIME, one of the true client-to-client encryption solutions, there's no telling what's going on. Now, email is a store-and-forward system where your client deposits it on an SMTP server, probably at your ISP. Then it may connect directly to the target's SMTP server, or it could go somewhere else in the meantime and then ultimately go there.

Generally these things can be configured to be opportunistically secure. That is, if they're late model servers, they will say EHLO, E-H-L-O, rather than H-E-L-O, HELO. And that says, oh, I understand more advanced protocols. They'll say, here's all the stuff I know how to do. One of them may be STARTTLS, meaning that you can then bring up a TLS connection between the servers. But the problem is, if somebody wanted to prevent that from happening...

[Yabba dabba do]

Leo: You are now one step closer to a site license.

Steve: If somebody wanted to prevent that from happening, they could prevent that communication and downgrade to standard port 25 nonencrypted, non-TLS connections. So that's why it's just - it's opportunistic. The point is no one - anything that says, "oh, if this says 'secure message' in the subject line," I mean, I would worry, frankly, about any organization...

Leo: That's so stupid.

Steve: ...that was trying to pull that stunt.

Leo: It's like the email from the lawyer that says, "If you are not the intended recipient, you must immediately delete without reading this email message."

Steve: Yes, yes.

Leo: Like, well, that'll work. Yeah, mm-hmm.

Steve: Oh, okay. No juicy details, yeah. So you're completely correct. Again, the only way to know is to encrypt it yourself and then arrange for the other person to decrypt it because still today, even today, it's more than likely going to have a link or two where it's not encrypted.

Leo: It does really underscore the need for some form of secure email. And Google and others have been kind of trying to address this. But I don't think it's an easy thing to do.

Steve: No. We're trying - I've said it so often. We're trying to shoehorn a system that was never designed to have security, and it resists it. It fights it. And it's winning.

Leo: Yeah. I'm learning how to use PGP. It's not that hard.

Steve: Well, good luck having the other morons on the other end of the connection.

Leo: These guys won't, yeah, right. That's the problem.

Steve: Well, it's funny, too, because my bank, whenever I'm like doing any sort of business, they won't do anything through email. It's all fax. And it's like, well, okay.

Leo: [Laughing hysterically]

Steve: Like somehow that's safer.

Leo: Well, it's not like anybody could forge your signature or anything. That...

Steve: No.

Leo: That is gold. It's the gold standard of authentication. Tim in Orange County, California has an idea for isolating Intel ME the cheap and easy way. This is what you were talking about lat week; right?

Steve: Yes. This is one of the topics was the so-called Intel Management Engine, which lives on the motherboard, aside from the regular Intel processor that we stick into a big socket with big fans running on it. This is a little secretive thing that's, like, monitoring your Internet connection.

Leo: There's been a lot of conversation about this on open source boards lately because, of course, if you want to have a, quote, "free," in the sense of libre, operating system, that's great. But your hardware isn't. And this Intel chip could be a spy chip, in effect.

Steve: It's all locked down.

Leo: Yeah, and you can't tell. It's an obfuscated blob. Just like me. Hi, Steve.

Steve: There's very little obfuscated about you, Leo.

Leo: I'm just a blob. I have a relatively easy and cheap way to bypass Intel ME, or at least prevent it from connecting to the rest of your network. Oh, you know, I've read this. I'm really curious what you have to say to this because I've read this in other places, as well.

ME can only communicate through the built-in motherboard NIC; or, if it's a server board, sometimes there's multiple NICS. If you have a desktop PC with vPro, well, simply install a cheap PCIe add-in NIC and cover the built-in network port with a piece of electrical tape. For laptops, use the built-in WiFi, as ME cannot communicate over WiFi. And if you have to plug in at a hotel or other foreign network, use a USB NIC; or, better yet, bring a portable router/firewall to insert between your laptop and the world.

On a sadder note, one of the features of the most recent Microsoft Excel app for Android is that, "Now you can open files that contain ActiveX controls." Oh, joy.

Thanks for a great podcast. Tim in Orange County. I'm actually really glad Tim asked this because I wanted to ask you the same thing.

Steve: Yeah. It is the case that there is no way for the integrated processor on the motherboard to have any idea how to communicate with some random Ethernet chip that's on the card. Generally, the plugin NICs will have a BIOS. But this doesn't operate through the BIOS. It's direct to hardware. So plugging in an additional outboard card will bypass the Internet Management Engine.

Leo: Wow. That's good to know. Very good to know. So really this isn't as big a hazard as it's been made out to be because...

Steve: Well, it's a pain, though. If you've got built-in sockets, that's the ones you want to use. You want to use your motherboard.

Leo: Well, here you go. This is a Linux laptop. I was all thrilled because it has an Ethernet port on it.

Steve: Yeah.

Leo: But I shouldn't use it. I should use the WiFi if I want to eliminate the ME, the man in the middle. And we should say there's no evidence that Intel's using this in any malign way, or that it has been compromised, either; right? I mean, it's, as far as we know, still secure.

Steve: Right. What's concerned is that there's no visibility into it, and they have locked it down completely, and they do not document it. And they're just like, it's on a need-to-know basis, and apparently we don't need to know.

Leo: Intel's documentation says on vPro systems. But there's also evidence that the chip, at least, is on systems that are non-vPro, as well.

Steve: Yeah.

Leo: So not getting a - this was another conversation I read on the boards, the libre boards, is, oh, well, if you just don't get a vPro computer, you're all right. Well, but the ME chip is on other Intel motherboards, as well. Whether it's enabled we don't know.

Patrick in New Jersey had an idea about blocking Intel's IME access. Oh, yeah, you obviously stirred up a hornet's nest: I was listening to last week's podcast SN-565, and I had an idea how to prevent IME from getting on your network. I have a desktop that I'm using as a server in my house. It's a third-generation i7 processor,

one of the older ones, and it came with one of the earlier versions of vPro. It uses the onboard NIC to get into the network, but with a different MAC address than the in-band interface the OS sees, and thus will have a different IP address on the network. On my computer I can statically set the IP, or have it assigned via DHCP.

I would think that the new versions of IME would operate the same way so as to avoid socket collisions between IME and the OS. That is to say, both IME and a service on the OS trying to use the same IP and port at the same time, as an example, port 80, of course, for TCP/IP. Or HTTP, I should say. One should be able to determine this by doing a packet capture and looking at the date in Wireshark. If this is the case, you should be able to block traffic for that MAC address on a switch or router. Thanks for all you do, happy SpinRite owner, et cetera, et cetera. Will that work?

Steve: Okay. So I did some digging because I was curious about this. And it is true that the older generation - and Patrick mentions that he's got an older solution. What I saw was v2.6, although I don't know how to relate that to time. But 2.6 of this management engine release did display separate MAC addresses. So it looked like two adapters to the wired network. But with 3.0 and subsequent, it is a single MAC address. So it is no longer obviously a second MAC. Were it two, and you could figure out which one, well, and you could figure out which one was which. And this is actually, this is what led me to the ARP table on pfSense was I was just curious to see on various systems whether I was seeing one MAC or two for a given system.

Now, it is the case that you can have multiple IPs per MAC. So we know that. It's very possible to assign several IP addresses to a single MAC address. Essentially, all of the traffic going to the gateway is going to different IPs through the gateway's single MAC. And we know that systems can typically be set up multiheaded with a couple IPs. You could not have two MACs with the same IP, though. So unfortunately, I mean, I guess if you find that you have different MAC addresses, then you could certainly filter by MAC address.

What Intel's boards, because there have been some discussions of this, what Intel's boards say is that, if you manually configure the IP, that is, you do not use dynamic host configuration protocol, DHCP, but you manually set the IP of your Windows machine, then you will not have IP sharing between your Windows or your Linux or whatever running on the motherboard, and the IME processor that is sharing that same NIC. That is, it will get a different static IP. And it may be, I didn't pursue this, that the Windows management tools allow you to configure the IP that that sort of shadow NIC operates under. In which case, if you're able to give it an IP, you could then use something like pfSense or any other firewall-enabled router to simply block that address from having any access outside of your network.

So, I mean, the problem is you ought to just be able to turn this off. The idea that you can't go into the BIOS and say "Off," I mean, that's the obscenity of this whole thing, the idea that users, owners of this machine, of this hardware, cannot just disable it. That's wrong.

Leo: Yeah. You know, I wonder if there will be a market at some point for - and a lot of people are not buying Intel chips for this reason. They're buying AMD.

Steve: It ought to be made much easier to say no to this.

Leo: Yeah. I think there's a market. I think AMD should really look at this market and say - clearly, I mean, I understand most, you know, if you're buying a Windows machine for use in business, you're going to...

Steve: Yes, an enterprise environment.

Leo: Yeah. You're going to embrace TPM - and this is kind of like the TPM Tempest over again. And you're going to embrace IME. And that's fine because it gives you some stuff you need and you want.

Steve: So enterprise level, like inventory and asset management control.

Leo: Right.

Steve: You're able to see who has everything and where these machines are.

Leo: Right. And you know if you work for a company, and the company owns the computer, it ain't your computer. But what about those of us who want our own personal devices, and we don't need this?

Steve: Yeah. And it's creepy.

Leo: And it's, well, we just don't want it. We just want to know that our hardware is all our own, and there's no spy stuff on it, potential spy stuff on it. And I think AMD could make - hey, if you're listening, already people doing libre stuff are often choosing AMD processors, yeah.

Steve: If given a choice, yes.

Leo: Hmm. It's very interesting, isn't it. Well, we've come to the end of our time together, Steve.

Steve: Right on time. Right on schedule.

Leo: Love the show. And I missed doing it last week. But I'm glad to be back for this one. We do it every Tuesday, 1:30 Pacific, 4:30 Eastern. That's about 20:30 UTC. If you want to tune in live, please do. Join us in the chatroom, IRC.TWiT.tv. Steve's not in the chatroom, but I am, and I keep an eye on what you're talking about and try to bring it into the conversation.

Steve: Yeah, that would overload me.

Leo: That's too much. You can ask questions of Steve, though. He's very active on Twitter, @SGgrc. And he has an open DM policy, so you can ask infinitely long questions via DM.

Steve: And, boy, that gets a lot of use, too.

Leo: Which is exactly why I don't do it. You can also - and don't email him. That's not - for some reason, that's not how he wants you to contact him. Do use Twitter DM. That's so much better. It's your life. Whatever. You also can contact him, and this is probably the official way to do it, at GRC.com/feedback. And questions will be used in two weeks as we do a Q&A section. But I'm sure Steve also kind of considers and thinks about it. And if you have stuff to feed him, he'd love to get it. You can also, while you're at GRC, take a look at the SpinRite program, the world's best hard drive maintenance and recovery utility; SQRL, if you want to know more about it, how it works, be ready for the launch.

Steve: Coming.

Leo: Any day now. Get that SQRL patch, put it on your laptop. And of course the Healthy Sleep Formula and lots of other stuff there, GRC.com, including this show, audio versions and full transcripts, which is really useful, plus of course Steve's show notes, at GRC.com. We have audio and video at our site, TWiT.tv/sn. And if you want to listen every week, just subscribe. We're on every podcast program there is. There are even TWiT programs for every platform. Find the program of your choosing and subscribe. That way you won't miss a single episode.

Had a great live studio audience, who are freezing to death right now. But thank you for being here. I like to keep it chilly in the studio. If you want to be in-studio live, there is limited seating for the shows I do in this studio. There are about five or six seats. So email tickets@twit.tv, and we'll make sure we get a chair for you. Thanks for being here. We'll see you next time, see you next Wednesday - Tuesday. I'll be here on Tuesday, just in case.

Steve: That's good. I will, too.

Leo: On Security Now!. Bye-bye, Steve.

Steve: Welcome back, Leo.

Leo: Thanks.