SECURITY NOW!

Transcript of Episode #554

Listener Feedback #231

Description: Leo and I discuss a quiet week's few security events, sharing some thoughts about Internet of Things (IoT) security, Bruce Schneier on Apple and the FBI, and some miscellany. Then we open the Security Now! mailbag to hear from our listeners their experiences and thoughts, and answer their questions.

High quality (64 kbps) mp3 audio file URL: http://media.GRC.com/sn/SN-554.mp3
Quarter size (16 kbps) mp3 audio file URL: http://media.GRC.com/sn/sn-554-lq.mp3

SHOW TEASE: It's time for Security Now!. Steve is here. Not a lot of news. Oh, he'll mention the WhatsApp thing, of course, and various other security news. But really we're going to focus on questions, 10 great ones from you, our listeners. Steve's answers coming up next on Security Now!.

Leo Laporte: This is Security Now! with Steve Gibson, Episode 554, recorded Tuesday, April 5th, 2016: Your questions, Steve's answers, #231.

It's time for Security Now!, the show where we protect you and your loved ones. We? Not me, you. Steve Gibson protects our loved ones and our privacy online and helps us understand what's going on in the world. Hello, Steven. He's waving.

Steve Gibson: Yo, Leo.

Leo: Waving doesn't work on the radio.

Steve: No, it doesn't.

Leo: And now Steve is waving his hand back and forth to signal hello.

Steve: I was just looking at the Security Now! episode number, and I love numbers that are fun. And next week is 555. Somebody actually sent me a tweet a few weeks ago saying, you know, you should do a special episode about timers, or time. Because of course it used to be a Signetics part. So it was the NE555. That was - for some reason Signetics began their part numbers with NE. The 555 timer was like this amazing simple little toolkit part from, like, you know, boy, it was in the '70s because I remember I was using them with a friend to make - to drive a relay for police for emergency vehicles. He had a company producing, like outfitting cars with all of the flashing lights and sirens and things. And so, and we didn't like the fact that you sometimes saw lights flashing with a non-50 percent duty cycle. Scott was a perfectionist about these things, as I am.

Leo: Oh, dear. Oh, dear.

Steve: And so we were able to set this thing so that the lights flashed exactly back and forth with a 50 percent duty cycle.

Leo: Oh, my goodness. This is nerdy.

Steve: Anyway, this is Episode - yes, at the tender age of, what, 17 or 18.

Leo: Yeah.

Steve: So this is a Q&A. And remember that last week's episode was "Too Much News" was the title of it, which would have nominally been a Q&A, but there was just too much to talk about. This week compensates for that. Nothing happened.

Leo: Too little news.

Steve: I don't know, maybe it's spring break. Everyone's, you know, all the hackers are, like, touring Europe or who knows where they are. But they don't seem to be up to any mischief. So we have a few things to talk about, some miscellany, but then I found 10 great comments, questions, thoughts, feedback from our listeners. Oh, and everybody went crazy over the picture for this week's front page of the show notes. It was tweeted multiple times. I got it through email, every possible communications avenue. And this is the...

Leo: Oh, I love this one.

Steve: Yes, how the FBI actually cracked Farook's iPhone. And it's just three frames. The first frame is the government guy, the GI with his FBI suit saying, "I have an idea." And then in the second frame there's like your typical mischievous little kid sitting on a couch. And he says, "Here, play with this iPhone; but do not, I repeat, do not unlock it." And the kid's like, eh. And then in the third frame the FBI guy back on the phone, "Bingo." So.

Leo: If only it were that easy. Maybe that kid is actually John McAfee on the couch.

Steve: Just give it to a kid. So this is just sort of a, as I said, not lots of security news. But I picked up on this. Business Insider had two pieces that Rob Price was reporting for them about people disgruntled over something that happened with a purchase they had made years ago with something called a "Revolv," which is a residential IoT hub that does something, who knows.

Leo: Oh, it's a hub. It's, like, no, this makes me so mad.

Steve: Yes, yes.

Leo: So it's like the SmartThings hub. It's a hub that you connect to, and you can connect your locks and your lights and whatever, and it automates your home.

Steve: Correct.

Leo: However, not for much longer.

Steve: And not cheap.

Leo: And not cheap.

Steve: $299.

Leo: Yeah.

Steve: The thing was purchased for essentially $300 with a, quote, "lifetime subscription." And I looked up, just to have it exactly right, what was the exact meaning of "caveat emptor." And the little definition came up, says it's a "Latin term that means 'let the buyer beware.' Similar to the phrase 'sold as is,' this term means that the buyer assumes the risk that a product may fail to meet expectations or have defects."

Leo: I really hate it when they say the "lifetime," and what they mean is the lifetime of this offer, or the lifetime of this device.

Steve: Right.

Leo: Not your lifetime.

Steve: And I was put in mind of anything we do where we're dependent upon the party we're paying to survive. For example, you could be paying into health insurance, and the insurance company goes bankrupt. Oh, wait, you know, what about all of that payment that I made? Oh, sorry, you know, we're gone now.

Leo: But, by the way, this company's not going bankrupt because the company that owns this is Google.

Steve: Yes.

Leo: This is infuriating.

Steve: And in fact these guys had to know. So these guys had to know this was going to happen. So in October of 2014, so, what, about a year and a half ago, Nest purchased Revolv, R-E-V-O-L-V. And then nine months later Nest of course was purchased by Google. So now the two Revolv founders - oh, and this was considered a - what's the term where you're actually buying the company to get the people, an acqui-hire?

Leo: Acqui-hire, yeah, A-C-Q-U-I-H-I-R-E.

Steve: Yeah. So the idea was, you know, they didn't really want this Revolv. They wanted the guys. And so the two founders, Tim Enwall and Mike Soucie, they posted on their website, I think about a week ago, and this is what the Revolv site now says: "We're pouring all our energy into Works with Nest and are incredibly excited about what we're making. Unfortunately, that means we can't allocate resources to Revolv anymore, and we have to shut down the service." So the site no longer, of course, talks about a lifetime subscription. Instead it tells its users that their products are no longer under warranty because, quote, "its one-year warranty against defects in materials or workmanship has expired for all Revolv products."

Leo: Oh, I'm so angry, yeah.

Steve: And then they have a Q&A where they ask themselves, "What happens to my Revolv device?" And their answer: "As of May 15th, 2016," so five weeks from now, "Revolv service will no longer be available. The Revolv app won't open, and the hub won't work."

Now, I was put in mind of the Zeo because of course the EEG sleep monitoring device was created by the Zeo company, founded in '03, whose doors closed in 2013, and the same thing happened. When I was coming up to speed on Zeo I looked and dug into the archives and looked at everything that was online, and there was a similar scramble when Zeo was announcing their pending demise because, just as with Revolv, the Zeo had a whole online thing. I mean, there was, you know, your nightly EEG was being uploaded to the web, where it was being indexed and cataloged, and there was a whole web-facing service that was part of it.

Now, of course, when they died, the Apple app got pulled from the App Store. Thankfully, the Android app continues to live on. So the hardware works, and of course more than 2,500 listeners of the podcast now own these Zeo devices and have been playing with them with their Android. But here's another example of - the good news is it didn't completely die. But it certainly had its functionality capped by the death of the company. So anyway, I just sort of wanted to bring this up as sort of a cautionary note, in general, about the whole Internet of Things deal because...

Leo: Because they do, in many cases, need a server to be running at the company. I don't know if, let's see, does SmartThings need a server running? I guess it would because that's what they phone home to, and then they connect to your phone through that.

Steve: Right. So, for example, our TiVos are network-enabled. So I'm able, you can go to the TiVo site and access your TiVo through a web page. You have TiVo apps. However, all of that is dependent upon the TiVo service. So if they were ever to die, a chunk of value - actually, in the case of TiVo it's even more significant because the guide which the TiVos use to drive them comes from the TiVo service. So I've been counting my lucky stars all these years, since I love this machine so much, that TiVo service has continued.

But we are entering an era where, exactly as you note, Leo, many of the things we do are tied to the 'Net. And what's annoying is in some cases it's only because their revenue model drives that. They want the connectivity. That is, the supplier of the device wants a relationship with us, and on whatever basis. But what that does reciprocally is create a dependence upon their continued survival; you know? And here was a company, in the case of Revolv, that was apparently doing well enough to get acquired. Unfortunately, not that long afterwards, they just decided to blow off all of the customers that made their rise and success possible, having promised them a, quote, "lifetime service," whatever that means. And a lot of people are justifiably unhappy.

Leo: I wonder how many they sold. I'm upset, not because I have one, although if I did I would be very upset, but because - and I think they may back down on this, and I'll tell you why. The damage to Google is huge because that means, I mean, Google already has a problem with discontinuing services people like. Google Reader.

Steve: Right.

Leo: Wave. I mean, I can go on and on. And this is - that is not the reputation Google wants. This is really bad. This is just bad behavior.

Steve: Yeah. So from our standpoint, I just sort of wanted to raise it as, again, as sort of a cautionary note. Consider when you're buying things. If they're $10 light bulbs, then, eh, okay. Not a huge investment. But I would be careful about making a major infrastructure investment in, like, equipping your whole home with technology tied to an external supplier where everything depends upon that supplier. What would be nice would be if there is a fallback provision so that, if the service is not available, you may not get all the features, but you're not left out, I mean, you're not left with your entire infrastructure investment scrapped because it no longer works.

Leo: Well, and this is the argument for open source and, you know, if they had the software as open source...

Steve: And standards.

Leo: ...and standards, somebody else could run the server. I could run my own server.

Steve: Right. We're in the early days. There's no question, I mean, we don't have any security standards. I mentioned that we do have that device provisioning protocol that the Wi-Fi Alliance are working for. And so that would be a good thing. The only reason we're here today - you and I are talking, able to talk to each other. There are people able to listen to us. It's all based on standards. It's the IP standard, Ethernet packet standards, TCP standards. All of this is built on standards which are decades old.

And so I think it's easy in this fast-paced mode that we're in today to sort of forget that we're jumping into something like IoT using the standards that exist. But this technology needs more than what we have now. We need another layer of standards that just hasn't caught up yet. So we're using the lower level plumbing standards which are, frankly, insufficient. So what's happening is the IoT vendors are all rolling their own. They're just making stuff up so that it works right now. All of this needs to be considered sort of transient because it's probably going to turn out to be exactly that.

Leo: Terrible news.

Steve: Yeah. Bruce Schneier weighed in, in the aftermath of the FBI saying sorry about all the news we made, Apple. We no longer need you. And I just - there was one paragraph from his longer blog post. The full link to his blog post is there for anyone who wants to read it, or just go find Bruce Schneier's blog, www.schneier.com/blog. So Bruce wrote: "Whatever method the FBI used to get into the San Bernardino shooter's iPhone is a vulnerability. The FBI did the right thing by using an existing vulnerability, rather than forcing Apple to create a new one. But it should be disclosed to Apple and patched immediately." And then he added...

Leo: But of course the FBI is saying, wait a minute, we want to keep using this.

Steve: Right. And then he added: "To be fair, the FBI probably doesn't know what the vulnerability is. And I wonder how easy it would be for Apple to figure it out. Given that the FBI has to exhaust all avenues of access before demanding help from Apple, we can learn which models are vulnerable by watching which legal suits are abandoned..."

Leo: Ah.

Steve: "...now that the FBI knows about this method." So there's been a huge amount of speculation in the wake of this. Many people weighing in on what should happen now. Is it right for the government to keep a secret that they have which is now creating a concern for all iPhone users that their phone could be cracked open at the government's whim? We don't know. And of course the argument is, if the government was able to get in without Apple's permission, who else can get in without Apple's permission? If there's some third-party company that sold the FBI the information, who else are they going to make it available to? So the idea that there's been essentially an official confirmation of a vulnerability which the FBI is going to keep to themselves and Apple will not be able to patch, this creates a huge concern.

So of course Apple's going to do what it should do, which is to continue moving forward, closing what few remaining vulnerabilities exist, immediately counteracting what it was that this court order was asking them to do, doing what they can to foreclose all of that because they've made their position very clear. And at the same time, Apple has been trying not to have their phones crackable for generations now. But one way or another it seems that the bad guys are always able to get in and root the iPhone, though it's getting increasingly hard as Apple continues to put more tough barriers up.

Leo: Here's a hypothetical. What if the FBI had a vulnerability, and they could keep it quiet? Like no one would ever get access to it? What would be wrong with that?

Steve: Well, okay. So the argument is that's impossible. That is, so someone found that vulnerability. And the argument is anyone else could. I would say it's better for Apple to have a mechanism under their control where they exclusively decide what phones are unlocked. That's superior than something we know nothing about. The FBI says we had a way, we have a way which allowed us to bypass the access code, the unlocking code, so that the phone would not be bricked. So we have to assume now that someone discovered a way to do that, which means anyone else could. And lots of people have been arguing that anything Apple would do is a backdoor.

I, of course, controversially argued that, no, Apple could design something that only they control. I would argue a completely secure platform where they have a key that they selectively administer is better to what we have now, which is we don't know who else has access. We don't know where the FBI got this. I mean, it's much more feasible to think they purchased it. We know they have a relationship with Cellebrite. There were some invoices that floated around the 'Net in the last couple weeks of payments, some big hundred thousand dollar plus payments to Cellebrite. And of course there's no information, there's no way to tie that to any case or any particular instance. But money is flowing from the FBI to a company that specializes in cracking cell phones. So we're in this situation where a third party presumably has a means that they are offering for sale to anyone who wants to come and get it.

Leo: By the way, this means probably Pete Williams was wrong when he asserted that he'd been told by the FBI, well, we got the data, but we haven't unencrypted it. Probably whatever this was, obviously, must have decrypted it. And now, by the way, the FBI's going around to law...

Steve: Remember, we still don't have any proof...

Leo: We don't know...

Steve: ...that they have access.

Leo: Right.

Steve: They've said - all we're going on is them saying they have access.

Leo: Well, they're not going to lie about that.

Steve: Okay.

Leo: Not to the court. Well, maybe they would. I don't know.

Steve: All they said was we no longer need Apple's help. We have extracted the data from the phone.

Leo: Right.

Steve: Nowhere did they say that they had decrypted it.

Leo: Ah, okay.

Steve: So one could argue, I mean, I'm...

Leo: They're going around and offering to do this for other law enforcement now. There's an Arkansas case they're going to help. So, I mean, they've actually sent out a memo to law enforcement saying...

Steve: I did see that.

Leo: ...we have a way to do this. So anybody want help? We can help you over here. Which means, by the way...

Steve: I agree. I think that pretty much puts it to bed.

Leo: It also probably means that they do know what the vulnerability is, unless they plan to pay $100,000 to Cellebrite for every case. They probably bought the vulnerability. But I don't know.

Steve: Yeah. We don't know.

Leo: We don't. Which is...

Steve: And that's one of the frustrating things. I mean, unfortunately, I don't think we're going to know. The argument has been made that, of course, in the security community, that the FBI should disclose to Apple what they're doing, and then Apple will fix it. And of course that's - but here we're also seeing Apple refusing to help the FBI. So the FBI is sort of certainly going to be unlikely to tell Apple how to fix a problem which, as you say, Leo, they're, like, happily able to produce evidence for other cases.

Leo: Right. And you have to think there's a little bit of "screw you, Apple" in this, too.

Steve: It's a mess. I did mention a couple weeks ago that I have a tickler set up on a Google probe, watching for any news of the Burr-Feinstein bill, and it went off yesterday. I immediately checked to see if there was news, and it's still more of this, you know, hold your breath, any moment now. The updated report from The Hill says that the pending legislation has been receiving input from the executive branch, from other parties. It's getting very ready to release. But not yet. And I'm expecting not to be impressed. I'm expecting to be unimpressed by the clarity of this legislation. I'm afraid it's just going to kind of be murky, which says, you know, we should be able to see into encrypted things. End. Period.

Okay. That's all the news that there was. WhatsApp was - Moxie posted, and the news of the day is that WhatsApp is now fully a part of or the full end-to-end encryption has been brought up in WhatsApp. There is a technical paper which looks distressingly short. It's only 15 pages. Rene was kind enough to send me a note in the middle of the MacBreak Weekly podcast, so somehow he was able to sneak that out with a link to the PDF. And I've been receiving tweets about it all morning. I had no chance before the podcast to look at it. My guess is it's sort of an overview. But what we know is that the deeper technology is all available and in the public domain.

So I will probably make that the entire subject, a detailed breakdown and examination of that technology for next week's podcast because I know it's a messaging application that's going to affect hundreds of millions of people, and that no doubt law enforcement's not happy about it. The question is authentication because that's always the bugaboo. How do you authenticate the other party? That's the one part that is the big inconvenience of Threema because they got it so right. And anytime you attempt to make that easy, if you weaken authentication, you can no longer state that this solution is secure. So we need to take a look at how is authentication being handled? The rest is easy. Authentication is what's annoying to people, and so it's the thing that normally gets shortchanged in a truly secure technology. So we'll absolutely be taking a look at that next week.

Leo: Yeah, I mean, the way WhatsApp works is you verify by phone number. So your phone number's leaked. And so...

Steve: Yeah, I don't know what that means. So, I mean, I need to...

Leo: Well, they know your phone number because that's how you match to other people is via phone number. So they know - they don't know your message. They've got Moxie Marlinspike, for crying out loud. And they're using the Signal Protocol, the Whisper Systems protocol.

Steve: Okay. None of that means anything.

Leo: You know Signal.

Steve: Yeah. But if there's somebody in between...

Leo: Well, they say there isn't.

Steve: ...and you have a great secure connection to the man in the middle, that's really nice.

Leo: They say there isn't, but I'm sure after you read the whitepaper you will let us know.

Steve: I mean, that's the only question.

Leo: No, no, you're right, you've got to look at it.

Steve: Yeah. Okay. So a couple bits of miscellany, and we'll get into our Q&A. For the first time in my life, I am migrating my domains away from Network Solutions. They finally - I've been upset with them for years. But it's just, it's so easy to just say, okay, renew, renew, renew. But I just - I hate that they trick you in every way possible.

Leo: I think I did that 10 years ago, Steve.

Steve: I know. I know.

Leo: So who are you moving to?

Steve: Not sure yet. I'm looking around. But I wanted to give people a warning. I wanted to warn our listeners that in the best case it is not a quick procedure. Initially I was thinking they're just making this as bad, as hard as possible. So I'm doing a test transfer to GoDaddy, although I don't think I'm going to end up using them.

Leo: Oh, I wouldn't, yeah.

Steve: Yeah. But so here's the protocol. You tell the registrar you want to migrate to, you go over there, and you say, hey, I want to transfer a domain from somewhere bad to you who are all wonderful and good. So you start that process. They send you a link. You verify that. Then you go to the registrar you're migrating from and ask them for a permission-to-move key. They have up to and take all of three days. Now, on one hand...

Leo: They slow it down as much as they can, of course.

Steve: They make it as painful as they possibly could.

Leo: Yeah.

Steve: So I go over to Network Solutions. Hi there. Oh, and there was nowhere on the website that I could find how to get this link. I had to call them. And so my blood pressure is going up.

Leo: Okay, that's a strike against them. A good registrar will give you a link, at least, to how to do this; right?

Steve: Right. And so I had to get on the phone, and this nice person navigates me through, oh, you have to set this checkmark here, click this, go into manage, scroll down to the bottom. See where it says that? Well, okay, that's not obvious. But click that, then move over here, then, you know. Finally I get this. And then they say, "We'll send it to you in three days." They could send it now.

Leo: Now, instantly.

Steve: But no.

Leo: Part of this they justify because you don't want to make it easy for somebody to steal your domain.

Steve: Exactly. So after I sort of like realized what the process was. So the first thing I do is wait three days. And sure enough, on the, what is it, 72 hours? On the strike of the 72nd hour, finally this little crypto thing comes to me from Network Solutions after three days. So I go back over to GoDaddy, and I put that in, which then GoDaddy uses to make the request of Network Solutions for the transfer. Now, four days. Again, could have been immediate. No. So then I get email from Network Solutions saying, hi, we've received the link we sent you back from the people you're sending, you're transferring - oh, oh, oh, oh, I forgot to tell you. Part of the process of getting the transfer-away link from Network Solutions was "We don't want to lose you. How about a $10 discount?"

Leo: Oh, yeah.

Steve: For, like, one year for $10? And it's like, you know, as everyone knows, I don't like to use explicit language. But it was like, you suckers, you know...

Leo: Oh, I know, I know.

Steve: You're charging me $39 a year.

Leo: Oh, that's outrageous. That's three times what it costs anywhere else.

Steve: Oh, it's $37, sorry, $37 a year. And that's, oh, and I had to go through a questionnaire, why are you leaving? And I said price.

Leo: And I hate you.

Steve: Well, and the other thing is every - I hate you, exactly, I hate you. Every time I update, they, like, try to turn on auto renewal. They try to sell me, for $9 a year, they want to sell me a private listing in the WHOIS directory.

Leo: Oh, no, no, no, no.

Steve: For something that costs them nothing. I mean, it's just I'm being nickel-and-dimed. And they want, you know, email addresses and hosting and, you know, would I like to have a free car wash. I mean, it's like, no no no no no no no no no no no.

Leo: That sounds like GoDaddy.

Steve: Just to get - just to renew my - oh. And finally I said, okay, I just cannot support this any longer.

Leo: So let me give you a couple of things.

Steve: Good.

Leo: Our old sponsor - and by the way, they haven't been a sponsor for years. They don't send us any money, and so this is not an ad. But Hover does have a domain transfer service. You'll have to give them your Network Solutions credentials, but they do it all for you. And now you know why; right?

Steve: Wow.

Leo: And I wish, I mean, I've done it by hand, and I wish I'd done it automatic - because I moved everything to Hover when they were a sponsor.

Steve: Seven days, just for anyone, I mean, this is not something you can do at the last minute. Seven days.

Leo: I've got a story for you. So somebody - LeoLaporte.com went up for sale about a month ago. Somebody was selling it. It was only a hundred bucks. I said, great, I'm going to finally get my name. I didn't have my name. Still would love to get Leo, but I got LeoLaporte.com. So the process, you buy it. It's over at GoDaddy, I guess. And you buy it, and then it takes forever, but eventually, like a week or two later, you get the domain name. And then they say, but now to confirm it you have to have us send an email to the administrative contact.

Steve: Right.

Leo: Which of course is the guy who owned it. So I change the address record in there, and I have them send me an email. Then I say, okay, now I want to get off GoDaddy as fast as I can, move it to Hover. They said, oh, sorry, because you've changed the address, you have to stay here for three months. That'll be $10, please. So they've set it up with the domain purchase system to guarantee that you can't leave them for three months.

Steve: Wow. Yeah.

Leo: I mean, these, I mean, it's a terrible business. It's a terrible business.

Steve: It is.

Leo: You know, we should...

Steve: I really want the security.

Leo: ICANN should fix this. This is ICANN's fault.

Steve: I'm sorry?

Leo: This is ICANN's fault. All these people are registered through ICANN. ICANN could enforce some rules.

Steve: Yeah.

Leo: I know, security's important, I know, I know.

Steve: Yeah. You absolutely want to make sure that...

Leo: You don't want your domain stolen, yeah.

Steve: ...you do not lose control of your domain name. That's crucial. And we've heard, there have been horror stories about people who haven't, for whatever reason, maintained theirs. But so Hover, you think? I guess this is an open call.

Leo: Yeah. Love Google Domains, Google Domains. It's Google. Right? So Google would be good. Google does domain names now. I used Dotster for years before Hover. They seemed okay. But, yeah. How should people let you know?

Steve: Oh, believe me. Twitter will just - I'm sure Twitter is already going crazy with everyone's favorite.

Leo: Who is the best domain registrar that won't jerk you around?

Steve: Yeah. I would just like a clean, simple, domain - I don't even care if it's more expensive. I've been putting up with $39 or $37 a year from Network Solutions and all of their nonsense. So, oh, my god, this whole we'll give you a reduced price if you don't leave. It's like, hey, you had your chance for the last two decades. Ugh.

Leo: Yeah. I've been very happy with Hover. You know, just because they were an advertiser, all our domains are there. The DNS is very easy. You know, yeah, that's part of it, too, is how the dashboard is and setting all that stuff up.

Steve: Yup.

Leo: And there's no upsell. And WHOIS privacy is included in the $10 a year fee. So I think you'd be a lot happier. At least...

Steve: At that sense I would because WHOIS is a nightmare. I have to shut down my email on WHOIS or just it's a torrent.

Leo: No, I know. No, you get automatic WHOIS privacy. Some domains are more expensive, you know, like .tv and stuff. But the basic domains I think are 10 or 11 bucks a year. Google's even cheaper, I think. But it's not about price, is it.

Steve: It's really, no, it's really not. It's about harassment and security.

Leo: No more upselling, please.

Steve: God. Okay. Now, unfortunately we can't run this video on the podcast because it's too long, and it's too wonderful.

Leo: What do you mean? People will go away from the show?

Steve: Well, I mean, I cannot explain to you - the problem is most people are listening rather than watching.

Leo: By the way, this podcast is awesome, Smarter Every Day?

Steve: Yes.

Leo: You're talking about the printable magnets?

Steve: Yes.

Leo: This guy does the best video podcast ever. I love him. He doesn't do very many of them because he's like, well, you can see why.

Steve: Yeah, I mean, they're really good. He went on location to the factory where they're printing magnets. Okay, so this week's bit.ly link, bit.ly/sn-554. Everyone within range of my voice, you have to watch this video. I tweeted it this morning. People were flabbergasted. It is just - it is beyond cool. So I will explain it so that people will go look at this video. I'm sure you can google "printable magnets" at this point, and you'll find it, too. Or bit.ly/sn-554.

The concept is our typical magnet has a north pole and a south pole on opposite ends. And, you know, back in elementary school it was a big red bar magnet with silver ends. And remember you'd stick it under, and you'd sprinkle iron filings on top of the paper, and then shake the paper, and they would array themselves in the magnetic field lines of force. And then lately we've been getting fancier magnets, little disk magnets. And there one side of the disk is north and the other side is south. So they stick together really tightly and so forth. These guys - and I'm sure they're just patented till next Tuesday because it's a fabulous concept. They came up with a machine, very much like a 3D printer, that prints magnets. But what it's able to do is, at the resolution of a printer, to put down a north or south. So they're able to generate, on a flat surface, a pattern of north and south.

Leo: This is so cool.

Steve: Oh, Leo.

Leo: So you can have a custom magnet printed with your logo. It's not visible on the magnet, it's the magnetic field is your logo.

Steve: And what, now, here's the brilliance, is if you have two facing disks with a complex pattern, you can do amazing stuff, like you can have them attract each other as you're pulling them apart until they get to a certain distance. Then they become a spring. But then, if you rotate them, they suddenly snap together.

Leo: What?

Steve: Because, oh, it's unbelievable.

Leo: They're like a lock and key almost; right? They have to be...

Steve: Yes. You're able to create a high-resolution pattern where the two faces interact in a sophisticated fashion so that it has a highly nonlinear effect. And you are playing the video, so we're able to see now...

Leo: Only those watching, but the rest can, you know...

Steve: For only those watching. There's a machine they show where it slowly pulls the magnets apart and measures the attraction or repulsion. And so you're able to create, like, rotary indents. Now, here we're seeing one of the coolest things. Those will not go together. They're springy. But if you pull them apart, then they want to snap back together until you rotate them.

Leo: And then they - boom - fall apart.

Steve: And then they snap apart. Oh, my god. It is so, so cool.

Leo: This is going to be revolutionary.

Steve: Yes.

Leo: Can I buy them now?

Steve: I want to - I just want a pair of those little handles as a toy. I'm sure they will start appearing. So just printable magnets. You've got to take a look at it: bit.ly/sn-554. That just jumps you to the video two minutes and one second in because there's about a two-minute introduction that's not about this, that sort of talks about magnets as we were growing up. But, wow. Very cool.

Leo: This guy does a great job on his podcasts. He's supported on Patreon, and it's Smarter Every Day. And every time I see one of these I go, wow.

Steve: And he said they're owned by Audible.

Leo: No, no no no no no. They have a sponsor called Audible. You might have heard of them once or twice on other shows. I don't think they're owned by - and by the way, the company is Polymagnet, P-O-L-Y-M-A-G-N-E-T, dot com.

Steve: These guys, they're not printing magnets, they're printing money, Leo.

Leo: Yeah, yeah.

Steve: This thing - oh. It's just beyond...

Leo: Smart Magnets. Printable is not the right thing. It's kind of more than that. It's hard to describe. I think he did a good job, though.

Steve: Yeah, I mean, well, because the idea of a high-resolution complex magnetic field - one of the things this does, if you have north and south very close to each other, is the magnetic lines of force don't extend out uselessly into space. They stay very close. So it allows a whole 'nother dimension of magnetic strength because it is the flex lines that pull the steel to the magnet. And so if you keep them high-density and close to the magnet, you end up with a much stronger magnet.

Leo: Three bucks and 60 cents. I'm sending them to you.

Steve: Oh, you can buy them?

Leo: You can buy them.

Steve: Oh, no.

Leo: They have a catalog online. All kinds of magnets. This is the latch magnet that you mentioned. They have the spring magnet.

Steve: I am so glad.

Leo: They have an online catalog, Steve.

Steve: I didn't look.

Leo: Yeah, yeah, yeah, all kinds. All kinds. Wow. You can actually create your own. They have specifications. So you could say how much attach force, how much distance range. Obviously, it's a big industrial business; right?

Steve: Wow. Wow. Wow.

Leo: But they're cheap. They're a couple of bucks each.

Steve: So I did want to mention I've been getting a ton of mail, and I found this actually over in the Security Now! mailbag, and I just relocated it, from someone named Ben Pulido, who's in Greenville, South Carolina, and the subject was "Sleep Formula." He said: "Steve, I've been a listener to Security Now! for several years. Thank you and great show, blah blah blah." And he said: "But sincerely, thank you.

"My wife has had lifelong issues with sleep apnea, nightmares, shallow sleeping, always being tired regardless of how long she slept or how many naps she takes, et cetera. 8:30 bedtime was her standard schedule, just so she could make it through the next day without crashing. Sleeping 14-plus hours and then napping later, only to be exhausted, was a way of life in our household. When I heard you talk about your sleep formula, I had a little spark of hope that maybe, just maybe, things could change.

"Steve, after two nights of taking the formula, my wife is a different person. She's not completely sure what to do with all of the energy she has now. To be honest, I'm a little puzzled, too, because for the first time in years my wife was awake and had energy at midnight. From the bottom of my heart, thank you for taking the time to research this sleep formula. It has made a drastic difference in our health and lifestyle. Best regards, Ben."

And I just wanted to mention I've been so busy with SQRL and with Never10 that email has been coming in, it's just been piling up in the Healthy Sleep Formula feedback bin. There is a feedback link at the bottom of that page for anybody who hasn't sent anything back. And it's - I don't have a sense yet. It doesn't work for everyone, but we're well over 50-50, is sort of just my rough sense. I'm seeing lots of people who, I mean, this has just changed their lives; and other people are disappointed because nothing happened.

So someday I will find some time to get back to that and spend some more time putting up some more documentation. I have ideas from all the research about things that probably, you know, things like Vitamin D and magnesium status are other things that are important, that are not part of the formula because I regard those as sort of foundational things that someone should be doing just for general good health. But they are things that could keep this from working. So anyway, it's been a huge success overall, so I wanted to thank everybody for their support and for the feedback that they've provided.

And also most of the mailbag was feedback about the last PC I will ever build. We have a huge audience, not surprisingly, of PC builders, and tons of input about all aspects of the PC. There's no way I can respond and thank everybody, so I just wanted to do this as an all-in-one thank you for all the feedback I got. Super, super interesting and useful.

Oh, and this tied into Leo's Tech Guy show. I found this also in the mailbag, Alan Farough in Kanata, Ontario, Canada. He said: "I was listening to Leo's 'The Tech Guy' show. He was talking about SSDs and spinning hard drives and the fusion drive compromise. I was curious, how does SpinRite see a fusion drive? Leo mentioned the drive has special firmware to marry the two technologies. I can see that would have to be the case. Does SpinRite bypass the firmware? If so, how could it do this? Would you only want to run a fusion drive on SpinRite Level 1 or 2 so as not to burn out the SSD cells?" He says: "I do not have one of these drives, just curious about it."

And so I just wanted to address that briefly. A fusion drive is a sort of a - it's a variation on the caching drive that we've had for a long time. What we've always had for years is a volatile RAM cache in the hard drive that buffers what the drive is doing. So, for example, when you read a sector, we know that the chances are great that you're going to be asking for successive sectors downstream of the one you just read. So the drives will typically go ahead and read. Or, if you ask for a given sector, sometimes the heads come to the track at the wrong location. Drives now all start reading whatever they encounter, even if they haven't gotten to your sector yet, because those may actually be upstream of the sector you've asked for. So they will have already read it during the previous revolution.

So the point is that as much cleverness as we humans could bring to bear have been brought to bear in improving hard drive performance by putting a volatile cache between the user and their computer and their magnetic media, which is a lot slower. So a fusion drive, of course, marries a, well, there's still RAM. But there's also SSD, the idea being that you sort of have a - it's like in the same way that a CPU has a level one cache, a level two cache, and a level three cache, now hard drives are getting multiple tiers of caching. In this case, the SSD is a buffer in front of the hard drive which is able to provide less speed than the RAM buffer, but more speed than the hard drive. So again, we're bringing the latest technology to bear, creating a hierarchy.

What v6.1 of SpinRite will do is deliberately disable the SSD portion, or allow SpinRite to test it explicitly. That is, there is an API in these drives, an extension to the venerable AT8 spec, which manages both the large sectoring that drives now have, and also the idea of putting nonvolatile SSD in front of the hard drive. All of the hooks are available, and I'll be taking advantage of them with the next generation. Now, I would say running Level 2 is the right thing to do. You are reading from what the SSD may have. And as soon as it doesn't have it, then you're pulling from the hard drive behind it.

So it's not clear what pattern testing, what doing full data inversion, what effect that would have with a fusion drive. It would be a function of exactly how they've implemented their algorithm. And there isn't really a standard for that. That's every manufacturer inventing their own - and they consider it proprietary - protocol. Like, you know, how many bands they divide the SSD into, what their caching algorithm is, is it an n-way cache, is it a simple MRU cache. There are any number of ways that this could be designed. So it's up to the manufacturer to decide how they want to. But the AT8 spec does give us the hooks for getting underneath all that. And that'll be another one of the features of 6.1. I've already got it in my notes for where 6.1 will be going. So we will be taking care of that.

Leo: Nice. Excellent. Let's take a break, come back with questions. I've got questions; Steve's got answers. We wouldn't do it the other way because it would be very boring. Steve has prepared some questions. And by the way, I asked him before the show, you going to talk about WhatsApp? And he said a little bit. But he said, "But I like to have some time to - I don't want to answer off the cuff." And I really admire that about you, Steve. Everybody else, including me, we're pundits. We just make shit up as we go along. You, you, my friend, you try to actually be correct. It's novel. It's kind of cute.

Steve: Just me.

Leo: No, I'm just teasing you. But I really do really appreciate that about you. And so here we go, questions asked at his website, GRC.com/feedback or through the Twitter, starting with an anonymous listener who used Shodan to look around at SMB: Steve, you mention in 553 that, since SMB, CIFS, and Samba should not be exposed to the Internet by any right-minded person, that the risk of Badlock was being somewhat overstated, given that your listeners were all likely right-minded people and therefore wouldn't open up SMB ports inbound on their routers, would they.

But go to Shodan and, of course, enter "SMB port:445," that's the default port, and Shodan shows plenty of systems exposing their IPC, inter-process control share; 250 of them were in the U.S. That's actually not many. That's pretty good. But more than 10,000 in Russia. One of the ones listed on the first five pages that I can view with a free Shodan account, almost all of them - oh, of the ones on the first five pages, almost all of them said they were a Dune [D-U-N-E] SMB server, which is apparently some sort of media player [dune-hd.com] which is headquartered in Taiwan, with engineering teams in Moscow, Russia, and Kiev. The company also has offices in the U.S. and a distribution network that covers more 60 countries. They seem to be very proud of all the awards they've won. Unsurprisingly, none of them appears to be for outstanding network security. Dune servers, wow.

Steve: Oh, boy. So just a reminder that next Tuesday is Patch Tuesday for Microsoft, and the big reveal for the Badlock bug, whatever that is. People have been tweeting me, saying, "Do you have any news? Do you know what it is? Could we get a early heads-up?" And it's like, no. I've found nothing about it. If it's as bad as they say, and they're not just overhyping this, then I understand the need to keep a lid on this. It literally would be a need-to-know basis where "need to know" means you are in charge of fixing any implementation of SMB, and nobody outside of your immediate circle. Because to hear these guys talk about it, the Badlock people, the moment this becomes public knowledge, there's going to be a land rush for finding and leveraging this vulnerability.

So again, I'm glad that our anonymous listener did a little Shodan look. I'm just scratching my head. How could port 445 be exposed, whatever this Dune SMB server is. I guess it's, well, this Dune HD, it's a media player that has opened SMB to the Internet. It's like, oh, lord, okay, well. So I don't think you even need...

Leo: Is it going to stop working on Tuesday?

Steve: I don't think you need an exploit right now. You just, you know...

Leo: What movies you got?

Steve: It's got 445 open. That's an invitation. You don't need Badlock. You can just use good lock.

Leo: Yeah, just use built-in Windows networking.

Steve: Lord.

Leo: Ben Schneider in Cincinnati, Ohio recently encountered Locky, that new ransomware.

Steve: Ooh.

Leo: He was listening to 553: I'm a lead network and system engineer for a managed services provider and, as such, the person ultimately responsible for security for all of our clients' networks. That's about 20 to 30 different small-to-medium-sized networks. Last Friday, one of our larger client's systems got infected with Locky. When the user found he couldn't get to his documents, well, he just logged out and went home early.

Steve: Oh, darn. Nothing's working.

Leo: When everyone came in on Monday, we started getting calls that their users couldn't access their files on network drives. It took all day; but since we do incremental backups every hour on our servers, we were able to restore everyone's files. Good man.

Steve: Yup.

Leo: He's earned his keep. I have been seeing a distinct uptick in the amount of Locky infections in the last week.

Steve: Yeah. I think Locky is the next really bad one. And I just got a kick out of this, you know, that all of the files are encrypted, it's Friday afternoon...

Leo: I'm going to go home.

Steve: Oh, what the heck.

Leo: You know what happened. He thought he did it.

Steve: And that leaves this thing a week, or a weekend...

Leo: Get all those files.

Steve: ...to crawl throughout the network, go find all of the shared server directories and encrypt them all.

Leo: You know what happened. He thought he did it.

Steve: Probably.

Leo: And he just said [crosstalk]. I'm going to go home.

Steve: Well, anyway, Ben, sounds like Ben is on the ball.

Leo: Good man, yeah.

Steve: Yeah. Managed service provider, so that probably says that these small companies are using remote network storage for all of their work. And so they've centralized management at his firm, and his firm is doing incremental hourly snapshots, so nothing was lost.

Leo: Nice. Good.

Steve: That's absolutely the way you want to do it, either in-house or outside. But, you know, bravo.

Leo: He earned his fees.

Steve: And again, when I saw this, I wanted just to remind people there's no excuse. I mean, you've got to have a backup.

Leo: Versioning is the key.

Steve: In this day and age.

Leo: Yeah. Patrick Harrington is wondering how to make the best case for HTTPS: Steve and Leo, thank you for all your work and your expertise. I'm in a class which required me to register with a library on campus so I'd be able to enter the library. It has some sensitive material in its collection. Ironically, the registration page is not encrypted with HTTPS. Part of this registration involved submitting some personal information that I try to avoid sharing online unless necessary. I bit the bullet and submitted it. I didn't really have much choice. But I would really like to bring this up with someone in the library because it's only getting easier, and this is a case where the encryption would be particularly important, given the information being transmitted. Any suggestion how to bring this up?

While Security Now! has been very helpful in expanding my understanding of cybersecurity, I have a lot to learn; and I suspect those in this library do, too. It's gotten easier to move to HTTPS. Institutions have priorities. And I'd like to make a convincing case for making this change despite being a neophyte with respect to cybersecurity. I guess I can thank my English degree for being able to use the word "neophyte," but my network security skills could use some work, and I'd like to sound as savvy as possible. Thanks for the show. Patrick.

I bet you they're WiFi, too; right? So insecure and WiFi.

Steve: Okay. So, yeah. There are several interesting aspects to this. First is I wanted to make sure that Patrick knew, just for the record, that the form you're on does not have to be secure for the form's transmission to be secure. So because the way in this kludge solution that we have that sort of outgrew HTTP, where the original technology was for browsers to be receive-only, somebody said, hey, how can we send stuff back? And so the way you send something back is by asking a question, that is, essentially doing a query of a special type, typically called a post, where in the body of the post you're actually making a query, but you're sending information with the query, as the query. My point is that the submission could be secure. I'm not saying it is, but there's a chance.

Now, the fact that you're filling out a form which is not secure is not a good sign. So even though it's possible that when you hit the Submit button, that's an HTTPS connection and transmission, and back in the day that's the way this was done. For example, when websites were almost completely HTTP, only the username and password when you're logging in, that button would be secure and would briefly protect your username and password in flight as it got to the server, and then you'd be back at, I mean, actually the page you would return to would be secure because it would be a response to a secure query, which was your username and password being sent securely. But then the site was always in a hurry to put you back down to HTTP. And so when you click the link, you return to nonsecure.

So the fact that you are looking at an unsecured form is worrisome. But remember, this is just sort of the tip of the iceberg of the worry because, if security is either a second thought or not a thought at all in this situation, for example, if it turns out that the form submission is not, even that is not secure, then that means that, as you are right to worry, Patrick, the data in flight could be eavesdropped on. But that also sort of lends a concern to the security of the data once it arrives. That is, this really feels like a situation where there's just no concern for security at all.

And, I mean, nothing that we're seeing would lead anyone to believe that, oh, yeah, they didn't really worry about giving you a secure form to fill out, or maybe not a secure transmission of the form data. Why would we think they're doing anything secure with the database where it goes? So, and then the last thing I wanted to make, or the last point, was that, boy, he asks what case do I make? Well, so there's a lot of things to think about, what I've just said. But the other thing we've seen is how difficult it is to get any motion. Everybody's busy. Everybody has, like, stuff to do. And the good news is security is in the air. The one thing the Apple and FBI fight did was really up the profile of this whole issue.

Leo: Yeah.

Steve: So that, you know, people who were not at all interested in, weren't even, hadn't even given it a thought before, are suddenly, wow, what's going to happen? And suddenly they're like, it's an issue now. So the good news is this is probably the right time to mention it. And then the trick is how do you get someone to allocate resources that are probably scarce, which are probably committed elsewhere right now, to do something that is, like, preemptive. And that's the problem. This is preempting a problem. But it's certainly a really great point you raise that, you know, here you're filling out sensitive information, and you have no sense for where it goes afterwards. But in this case Patrick had no choice.

Leo: Well, I'm glad Patrick listens to the show and realized the risks. That's a good thing. And he's no neophyte, as far as I'm concerned. He might be a troglodyte, but he's no neophyte. Peter Haines - I don't know, that's actually not quite right either. Peter Haines in San Bernardino wonders about encryption of data at rest: Steve, long-time listener, first time emailer, yada yada. My employer is wanting to improve our security to come closer to PCI standards. That's the credit card security standards; right?

Steve: Yes.

Leo: One aspect is encryption of data at rest. Microsoft's database solution is SQL Server Transparent Data Encryption or TDE, but Wikipedia has a reference to a recent article on breaking their encryption from Simon McAuliffe that suggests TDE doesn't offer any practical security at all.

If I understand it correctly, it says there's a whole class of problems like SQL injection that it doesn't help with at all. And those that it potentially might help with, it doesn't really because it stores the encryption key with the data at rest anyway. Oh, great.

Steve: Yeah.

Leo: Probably unencrypted, just sitting there in a file called "encryption key." What's really going on here? Do we live in a world where a company like Microsoft can sell a security product that adds no practical security? Or is there some practical benefit? Love the show. Thanks, and keep on spinning right.

Steve: So this is a great question because we like talking about technology and the way stuff works. So this is a database that has to be online, presumably. It's a SQL Server, I mean, we don't know exactly what the usage mode is. But in your typical corporate infrastructure, your servers are on 24/7, even when your employees go home. A lot of them are working from the home and working in the evenings and on weekends. So they probably still need access to corporate resources.

Compare this to a TrueCrypted hard drive because they're very similar. The idea is that, with a TrueCrypted hard drive, it, too, while it's in use, the encryption key is in RAM. And in fact we've talked in years past about various exploits, for example, you know, spraying Freon on the RAM and pulling the RAM out of the laptop and running over and putting it somewhere else and then powering it up and sucking out the data from the RAM. Or sticking a Firewire interface, because a Firewire has DMA access, into main RAM, and taking a snapshot of main RAM, and then finding the TrueCrypt key that's sitting there in RAM. It has to be there because it's in use in order to dynamically encrypt data coming in, being stored on the hard drive, and decrypt data that's coming from encrypted hard drive back into use.

So the reason we bother with TrueCrypt of BitLocker or VeraCrypt or any of these is when it's not in use. That is, when we shut the system down, we log off, we power down, that key is lost. If then the laptop is - we lose control of it, it's stolen or lost or whatever, what's there is of no use to anyone. The problem with - so that's the model. Now we move that model into a real-time online database. And I have to agree with anyone saying, eh, how is this useful? It's useful if bad guys came to raid the company and unplugged all of the servers and took them out, like the FBI has been known to do. They just come in, and they yank everything out, and they take it away. Well, in that case, the SQL key will fall out of RAM, and they'll have an encrypted database.

But frankly, this model, it's encrypted all the time. So access to the physical hard drive would show pseudorandom noise. Queries through the SQL interface would show you plaintext data. So that's the thing to recognize. I kind of agree that it's not clear how valuable this is. It doesn't cost anything to have it in there, so I'm not surprised Microsoft's got it. Maybe it's a bullet point that Oracle offers, and so Microsoft said, okay, yeah, we're going to do that, too, on-the-fly encryption to the physical storage. I can't really think of a great deal of value that it adds. But that's certainly a reason to wonder about what use it is.

Leo: What use is it?

Steve: What use is it? Because there isn't a mode, probably, where, I mean, unless it's on a laptop, for example. But in a corporate setting, it's probably on 24/7, so the key is always there.

Leo: But that is the challenge in many situations. I mean, it's not an unusual challenge. Continuing on. Question #5, Mr. Gibson, comes to us from Ken Drexler in San Rafael, down the road a bit, California. He hit a bump with Never10 and Norton: Hi, just tried to download never10.exe from GRC.com. It downloaded to my desktop. That's where I put my downloads. Then Norton Internet Security went to work and flagged it as a threat. It called the threat WS.Reputation and - pardon me, WS.Reputation.1 - and reported that it had deleted it. Wait a minute. I looked in the Trash; but, no, Norton didn't put it there. How do I get never10.exe installed? I suppose I could disable Norton temporarily. Is there another way? Thanks for

Security Now! every week.

Steve: So I got a kick out of that because I had a lot of reports from people.

Leo: Hey, Norton.

Steve: And this is essentially social networking meets Internet security. It's what we've come to, and sort of as a last resort, is that now our antivirus is using reputation. It's not my reputation, or actually it might be the reputation of the certificate that signs the software.

Leo: Oh, interesting.

Steve: What happened was I was using a certificate that still had about six months on it, and it was an SHA-1 certificate. And so there were some complaints from some Windows systems. Some Authenticode verification was configured not to accept SHA-1 signed software any longer. So it's like, oh, really? Okay. So I contacted DigiCert; and as I mentioned last week, they jumped right up and helped me, and within hours I had an SHA-256 cert. But it was a new certificate. So it, unlike the certificate I had been using for years, had no reputation. And so the, what was it, Microsoft calls theirs the SmartScreen. And so for about 24 hours, anyone who was trying to use the newly signed, or signed with the new certificate, would get a warning from SmartScreen, or in this case from Norton, saying, eh, we don't really know about software signed with this cert, so we're just going to not. You can't have it.

Leo: Wow.

Steve: And so, briefly, deleting it was all you could do. It was obviously perfectly fine. But unfortunately what the world has come to is a better-safe-than-sorry. And, I mean, I can understand. Now I have a reputation with this certificate, and it's a three-year cert, so everything that I do from now on will sail right through. But for a period of time it was, eh, we're going to just say no until we determine one way or another whether this looks like it's good or not. So it's an interesting new little hiccup for anyone with a newly minted certificate who's creating something that a lot of people want to have access to. You actually may not be able to get it to them for a day.

Leo: Wow. That is kind of unexpected.

Steve: Yeah, isn't it? But you can see how it's evolved. It's like the only thing - it's, you know, we've talked about blacklisting and whitelisting, the argument being that only, in the same way like with a firewall, you don't block the bad ports, you open the good ones. And, that is, you blacklist everything, and then you whitelist the ones you know you need. Similarly, there are corporations now that are using a whitelist rather than a blacklist. Only specific applications that have been given the green light by IT are allowed to run within that corporation. Those systems will not run unauthorized software. I mean, they've gotten to that point as, I mean, just as matter of being defensive. They have no choice.

Leo: I suppose it's - you're too diplomatic to say, but I might say it wouldn't be - if you're not going to upgrade, you're doing Never10, what are you running Norton for? Get rid of Norton. You don't need an antivirus. I guess if you're on XP you might. But you don't need an antivirus. That's dumb. Not Norton, anyway. Lisa in Oregon worries about - Steve didn't say that. Leo did. Just so you know. Lisa in Oregon worries about unintended consequences of Let's Encrypt's success: While it's great that millions of additional websites now have HTTPS certificates, thanks to the free and efficient Let's Encrypt CA, I'm concerned that our corporate IPS and firewall will be unable to inspect the increasing percentage of incoming encrypted traffic and therefore be less able to block malicious packets. Is that something I should worry about?

Steve: Well, I liked this because I just wanted to remind people that I've reversed myself on this issue recently.

Leo: What?

Steve: Recognizing that a corporation has a right to control their bandwidth, and as the world becomes encrypted, as malware will thus be traveling over encrypted connections, a corporation, I think, is right to require their systems to have a certificate authority which they sign and their perimeter able to mint those certificates and crack open encrypted communications and inspect it. I think that individuals working for the corporation, using the corporation's infrastructure, need to understand that, yes, their communication is private outside the corporation, but within the corporation's network it's company property. It's company bandwidth. And so they don't have an absolute right to privacy when they are using corporate infrastructure. And so this is inline with that.

So if Lisa's company doesn't have the ability to have an appliance that is proxying connections, then she's absolutely right. It will be virtually useless in the future. And so there is unfortunately, well, or necessarily a huge market for appliances which will be doing this. It's just going to be standard operating procedure in the future. And so I have gone from thinking that it's just a horrible thing, the idea that an encrypted connection is not actually point-to-point. But recognizing that within a corporate infrastructure - different from an ISP. That's a different story. I'm worried about the day that happens. But within a corporate infrastructure on their property, their network, their bandwidth, and you're working as an employee, I think they have the right to inspect the traffic to protect themselves from something that might come crawling down some query your browser makes.

Leo: Mike in Chicago bring news of an uber-cool facility or service.

Steve: Oh, Leo. Go.

Leo: I like the name. Hi, Steve and Leo. I'd like to ask your opinion on a file transfer service called File Pizza (file.pizza). It was developed by two graduates from Steve's alma mater, UC Berkeley. It's great. Files aren't stored anywhere. They're sent from your computer via web browser using WebRTC. But this is my main concern. You mentioned on previous podcasts issues with man-in-the-middle attacks on WebRTC. So my question is, is this service secure? Thanks. Mike.

Steve: Okay, now, first of all, who knew that pizza was a top-level domain?

Leo: I knew that. I knew that. See, if you were on Hover, you would know there's all sorts of fun top-level domains. Hover has them all.

Steve: Oh, I'm getting old, Leo. Pizza?

Leo: Oh, man. There's all...

Steve: In my day, we had .com, .org, and .net. And maybe .gov.

Leo: Oh, no no no no no. Lisa's new website is LisaLaporte.ceo.

Steve: Nice.

Leo: How do you like that; huh?

Steve: So everybody, this is uber cool: file.pizza. It is a point-to-point service. It is on GitHub. It's all open source. It's open protocol. It leverages the webtorrent protocol and WebRTC. It solves the problem of you wanting to send a file to someone else in real time.

Leo: That's the problem I have with it. Unfortunately, they have to be waiting for it; right? You have to be doing it at the same time.

Steve: Correct. Exactly. So you go to file.pizza. You click on the button to choose the file you want to send. And it finds it, and then it gives you a tag for you to provide the other person. I tried it this morning, and my tag was file.pizza/lobster-zucchini-duck-pecans. So they're using word salad passwords: lobster-zucchini-duckpecans. And what's nice is you could say that over the phone.

Leo: Right.

Steve: You can email it to someone. And it's still, given that their vocabulary is large enough, and these are going to be transient, and they could easily, since that's going to go to them, they could do all kinds of brute-force protection and so forth. So essentially their service, the file.pizza server, is in sort of the same way that a NAT server allows NAT penetration, that is, allows endpoints behind NAT to talk to each other, they're just providing the IP address to each other in order to connect your two browsers. And then the WebRTC protocol performs the file transfer.

So it solves the problem of needing to stage a big file, or any file, somewhere else first, like uploading it to some file upload site, or to Dropbox, and then giving someone a public Dropbox link, you know, that kind of solution. Here you say, I'm ready. The file's ready to go. You send them this link that you get from your web browser, which comes from the file.pizza server. They put that in at their end, or just click the link, and their browser downloads the file directly from you.

Now, the WebRTC protocol provides encryption. But once again, we have an authentication problem because there is sort of no authentication in this. It's easy to use because no one's worried about authentication. So what I would say this offers is security - I'm sorry. What it offers is convenience, not security. If you really care about security, encrypt it first. I heard you mentioning on one of the shows recently, Leo, that just PKZIP has good security, and it does. What's the other one that I like? Is it AEScrypt? AxCrypt. I think it's AxCrypt, A-X-C-R-Y-P-T, is also a very nice, super clean, local file encryptor.

So if you're worried, encrypt the file first, then send it, and let the person know what the decryption key is for your encryption. Or if it's just something you don't care about, don't worry about it. I just wanted to say it cannot be secure because there is no authentication. So there is a man-in-the-middle problem. But what this is, is convenient. And so many times for things that don't absolutely have to be secure, this is just a cool point-to-point solution. Oh, and I will mention many people, I tweeted this this morning, lots of people's browsers did not like the file.pizza server because it has a Comodo certificate.

Leo: Oh. What do you expect from a .pizza site anyway? That's funny.

Steve: So it would be nice if these guys would fix their certificates.

Leo: At least it's got a certificate. I mean...

Steve: For what it's worth, anybody else could bring up this server. They're making it available. It's publicly available. You could grab a domain, bring up a server, and then knit these connections together for people. Anyway, very cool. If somebody has this problem, this'll fix it.

Leo: I use - because I'm not, you know, I'm transferring files to, like, radio stations. They're not waiting for me. So I use a service called Transfer.sh, which is also open source. You could fork it on GitHub. One of these days I'll get around to setting up my own server. But you can use Transfer.sh right now. And it's a command line. I mean, this is geeky. It's a command-line server. So you use cURL to upload to it, or you can write a little script. And then you get a unique ID. And of course this is completely insecure. But if you encrypt before sending, it's completely secure. And then you can give somebody that link. I do it to transfer stuff from home to work, and to do radio show stuff.

Steve: Didn't the TWiT network have a sponsor who was providing...

Leo: Yeah, ShareFile, which is, of course, a better corporate solution. And I still use that. And Dropbox will do it and stuff, too. But this is nice for quick-and-dirty command-line sharing. And what's nice is it's free up to 10GB. But they don't store it for more than two weeks. So it just, boom, it's just there. But I use command-line all the time, so it's nice just to say, you know...

Steve: Yeah.

Leo: And I wrote a little script that says transfer. So I say transfer file, and it returns the URL to me.

Steve: We've been following your abandonment of the GUI, Leo.

Leo: I am loving, well, I'm not abandoning the GUI. I'm abandoning Windows is what I'm doing. Although now I'm trying this new OS called NixOS that is highly secure, Linux-based, but you build it yourself. It's like Arch, but you build it yourself. And that way you know exactly what's going on and stuff. Linux is getting better and better. I'm telling you, Steve, stop the insanity.

Steve: I'm only here because I'm able to produce something like Never10. Remember, there's more Windows OSes than anything else, still. Still the majority platform. And the people on Windows are the ones who need me, so...

Leo: Yeah. Somebody in the chatroom, and I don't remember who, but I'm going to give it a little plug, mentioned a Spanish distribution based on Arch. The idea of Arch Linux is you install just what you need and no more. But the problem with it is, I mean, literally that could just be command-line and nothing else. So somebody in the chatroom recommended something called Antergos, A-N-T-E-R-G-O-S, dot com. It's a free Linux distro that gets you started with Arch. So this is kind of a cheat. But you get a whole GUI based on GNOME, ready to go. And then but you still have Arch under the hood, so you can continue to - it's a rolling update and all that stuff. So I've been playing with it. And I have - see all these keys? This one's Apricity. This is Antergos. This is - I don't use Ubuntu. That's Debian. You know, just I'm trying them all out. It's fun.

Steve: Right.

Leo: Yeah, I get nothing done. But who needs to get anything done? I have to say it's better than running the alternative, which is running Windows. And I'm a little worried about Mac. I really feel like Apple doesn't, you know, they make so much money on iPhone, they don't really care about operating systems anymore.

Steve: I wanted to mention that you guys were talking about Safari on MacBreak weekly.

Leo: Yes.

Steve: We don't talk about it often because it's not in the news all the time in the same way that Chrome and Firefox and IE are for various reasons. But I saw a recent very nice write-up from a serious hardcore web designer who prefers Safari over all the other browsers in terms of its support for standards and its speed. Again, it's modest in that it's not making a big bunch of noise. But it's there, and it's solid, and it's working. And Apple's clearly giving it some time.

Leo: Yeah. And it's based on WebKit, so it's open standards. I mean, you know. Chrome used to be WebKit. They forked it. Moving along. We've got more to do. Norm Aylward, currently in Thailand, has second thoughts about SNMP - Simple Network Management Protocol. You mentioned it last episode. But I had understood from many articles on the web that it had security issues, giving out too much information. I think this was especially true on the older version. I have SMP shut off at the moment in Smart Switch and routers. I have two ISPs - they go down a lot here - and two routers feeding into a Peplink 30 balancing router, then to a D-Link DSG 1210 switch. The Peplink does have a SNMP trap in Version 3. So maybe it is safe to turn this on in the network. What do you think?

Steve: So I'm so glad Norm asked this because it was really, I considered it a faux pas of mine last week not to talk about security. I was mentioning that many routers offer the ability for external software like the NetWorx, N-E-T-W-O-R-X, utility, to monitor the bytes flowing in and out of the WAN interface as a really cool way of managing and keeping an eye on the overall bandwidth usage within a network. And I should have mentioned SNMP security because it is a huge issue. So Norm is exactly right. I'm assuming that nobody would turn it on externally, that is, on the WAN side.

What happened historically is that it was a very nice tool which network engineers used for managing bandwidth. And so big iron routers had SNMP on, like making it available. And it also has a default well-known username and password of "public." Which is meant to warn people that, by the way, the password is public. And so don't leave it set to public if you don't want it to be public because otherwise it's public. But everybody, of course, did. And we went through a whole phase in the industry of SNMP nightmares. Because it's not just only a passive read-only protocol. There is a read-only, and there is a read-write password. And if someone gets write access, they can actually reconfigure your router through the Simple Network Management Protocol. It's what it sounds like.

So I'm so glad this came up because, if you were to turn on SNMP, you want to make sure you're doing so for your LAN side, that is, so that it's LAN-facing and not WAN facing. Otherwise you would be potentially opening yourself to a problem. And by all means, change the password in any event so that only you and the utility that you want to have monitoring your router is able to do so. So thank you, Norm. I was talking only about the technology and not the security of it last week. And that was wrong.

Leo: Andrew in Virginia wonders about VLANs for security: You talked about VLANs briefly when discussing IoT security. You didn't really explain how they work, or don't, as a security measure. Perhaps I missed a previous discussion of that. I don't think we've talked about VLANs before.

Steve: No, we haven't.

Leo: If not, could you go over how VLANs work for security, and maybe in the context of Internet of Things? Thanks.

Steve: I'm going to do so very quickly now, with a promise that we will talk about it in depth in the future. Many people misuse VLANs, so Virtual LANs, as a security solution. And it is more of a management convenience solution than a security solution. In theory, you could use it for security. In some contexts, if you absolutely knew how every piece of hardware worked, but there is a history of - it's called "VLAN hopping," in fact, it's been named, where the VLAN encapsulation is broken out of. In the IoT world, I'll just say I don't see much application because the VLAN security, such as it is, which is to say not very good, only applies usefully to wired situations, and only where you've got switches that, again, the switches - excuse me, I've got a little something tickling me.

The switch is the negotiator of Ethernet frames passing through. And many of the exploits have been exploits against, like, small - they're called CAM, Content Addressable Memory tables that switches have to manage which ports of the switch are associated with which virtual local area networks. And so you're depending upon the switch to route the traffic out particular ports. Well, IoT is more often than not wireless. So in a wireless mode, your Ethernet is in the air. So VLANs make no sense because everybody can see everybody else's Ethernet frames. So even if you use them, they'd be providing you no security.

So for now we'll just say I don't really see an application. They can be used, if you really understand the way your hardware works, and if it's hardened against VLAN abuse. But it's much easier to use it in a wired setting than in a wireless setting, where, again, because it's about physical Ethernet connection security. And in a wireless mode the packets are in the air. Anyone can pretend to be on any VLAN and see packets, if they were part of an airborne VLAN.

Leo: Yikes.

Steve: Yes.

Leo: A paranoid listener in Cairo, Egypt wonders about built-in drive encryption: Nowadays, a lot of laptops are sold with self-encrypting drives. We have SSDs that claim to be FIPS 140-2 OPAL compliant, meaning the data on the SSD or hard drive is always encrypted at rest. In all these cases, the drive has a key that is encrypted by the user's ATA password when set in BIOS; or via special interface to BitLocker, that's of course in Windows; or a default one, if the user hasn't specified one or enabled BitLocker.

The question is, in light of the recent Apple/FBI debacle and the clear privacy implications, are such drives really safe? And if I were in Cairo I'd be asking this question. I think this is reasonable. Or is the FIPS 140-2 standard also backdoored in some way by the government? I wasn't able to find useful information when researching this, so I was hoping you could shed some light on this in some future episode of Security Now!. Thank you. I wish you guys the best.

Steve: My sense is, and just sort of being a skeptic, is we never hear of law enforcement having any problem with such drives.

Leo: Mm-hmm.

Steve: You know? It's a big problem when the phones are encrypted. But, you know, people's hard drives, when they were using TrueCrypt, oh, yeah. Brazil couldn't crack someone's hard drive that was with TrueCrypt, and so they sent it up to the states here for the FBI to take a crack at it, literally. Somehow, encrypted hard drives that are encrypted natively, never really hear about them being a problem.

Leo: Well, that's telling, isn't it. That's very...

Steve: Uh-huh. That's sort of a little inverse, an inverse warrant canary...

Leo: Yeah.

Steve: ...is the fact that, eh, that doesn't seem to be a problem for anybody. And I have heard anecdotally, and I mentioned this a week or two ago, that I'd heard stories of drive manufacturers being able to open them, under warrant from a court order through law enforcement. So I think, again, great security for somebody stealing your laptop. I would not trust it for, I mean, if you absolutely, in Cairo, if you absolutely had something that you knew nobody could get into. I mean, for example, even if it were brute-forcing, we don't know anything about the key derivation function. Is it deliberately set, I mean, even if there is like a deliberate slowdown for password guessing on a drive, I mean, there's never been any talk about 10 strikes and you're out, where you keep guessing the ATA key. All it does is it just gives you different gibberish. And so my sense is it's at best very weak encryption.

Now, nice from a standpoint of decommissioning drives, where you use the password from day one. The drive is always encrypted. And so when you destroy the password, then presumably no one can get to the drive. That is, it's just gibberish. You don't have to do the whole cryptographic wipe of the drive because it was always just gibberish, if you can really affirmatively destroy the password. Again, I don't trust it. In this day and age, if you absolutely had to have good security, I'd use TrueCrypt. BitLocker, I'm using it on my Windows 7 laptop because that's all the encryption I need. We've got to get away from this idea that security is absolute. It just isn't.

And we'll be talking about this more with WhatsApp next week. It's something you've heard me say before. When you were using Telegram, Leo, it's like, yeah, that's good enough. Is it perfect? No. They made up their own encryption algorithm. It scrambles up people's brains when they look at it. It's like, who came up with this? But the bits are scrambled. That's fine for sending notes to your partner about when you're going to be home for dinner. But is it incredibly great encryption? No. But it doesn't have to be. So we need to start thinking in terms of this stuff being relative because that's the reality.

Leo: Encryption comes in layers. Or security comes in layers. And the idea is to...

Steve: Yeah, and we don't absolutely always need perfect security because it's a hassle. You know, you're going to sacrifice some convenience.

Leo: And you said TrueCrypt. You meant TrueCrypt, not VeraCrypt or some other?

Steve: I meant TrueCrypt.

Leo: Okay. Continue to use TrueCrypt.

Steve: Yeah, I'm hearing some weird things about VeraCrypt, like every time you log in, it takes a minute because they've, like, gone totally overboard with their password-based hashing, believing that asking someone to wait a minute is reasonable to log into their hard drive. It's like, what?

Leo: You only do it once; right?

Steve: Well, every time you turn your computer on.

Leo: Right. Another nice thing about many Linuxes. You can use their logical volume manager and encrypt the entire hard drive as part of the boot process.

Steve: Yup.

Leo: You enter a password, and it's good, strong encryption.

Steve: Yup.

Leo: And it's all open source, which means, if you have the abilities, you can verify it, or somebody else...

Steve: And we know how to do it.

Leo: It's not hard, yeah.

Steve: It's not black magic. It's just implementation.

Leo: Yeah.

Steve: We know how to solve this problem now.

Leo: Yeah. What version of Linux did Snowden use? I can't remember. I think they mentioned, but I don't...

Steve: Don't know. I'm a FreeBSD person, myself.

Leo: That's the one to go with, obviously.

Steve: And I do build my kernel from scratch.

Leo: Ohhh. You might have seen the...

Steve: It's amazing, you see lines and lines of C-compiled going by.

Leo: [Crosstalk], isn't it?

Steve: It's like, this can never possibly work.

Leo: It's not going to write it.

Steve: There's no way every single one of these things works.

Leo: I know.

Steve: And it, like, half an hour later, look, you have a new kernel. It's like, you're kidding me.

Leo: My kernel, it built my kernel. If you don't build your own kernel, you're nothing.

Steve: Nah.

Leo: By the way, Tails, of course, that's what Snowden used. And that's what this - that's this key. Tails is a hardened Linux for use on a boot drive, a boot key. And, yeah, that's it. So, yeah. In fact, I don't - there was recently a little conversation about is any 386 platform really - can it be considered impervious? And because of the stuff that Intel insists on and AMD insists on...

Steve: Oh, there's stuff that Intel is doing [crosstalk].

Leo: It's built in the microcode. You can't not - so I think there's some progress along the lines of maybe making an ARM processor that doesn't have this stuff built into it. So you could run FreeBSD on top of - but if you're running it on an Intel or AMD processor, you could still perhaps be compromised. I mean, let's not go crazy, but...

Steve: Yeah, Intel has this thing, the Intel Management Engine, that's low-level access at the motherboard level, underneath your OS. And it's like, I immediately go and turn that off. It's like, I don't want that. And it's on by default, of course, like everything else. Okay, my friend. We're done.

Leo: We done. We be done. I hope you enjoyed this show. We do it every Wednesday, I'm sorry, Tuesday afternoon, 1:30 Pacific, 4:30 Eastern, 20:30 UTC on TWiT.tv. You can watch life, as with all of our shows, but we also make on-demand audio and video available. Steve has copies of the audio on his website, GRC.com. He also has written transcripts, if you like to read along. And that makes it easy to search, too. You can google "site:grc.com" in the topic, and you'll find transcripts and then can listen to the show, if you wish. He also has lots of other great stuff. His Sleep Formula is there. So are SpinRite, the world's best hard drive maintenance and recovery utility, his Perfect Paper Passwords, his Password Generator, lots of stuff.

Steve: Never10, which is still going like wildfire.

Leo: Never10. How many downloads so far?

Steve: 107,000.

Leo: Nice. That's awesome. Never10, which you run once, turns off the 10 upgrade, and you just delete the program because you won't need it again.

Steve: Done.

Leo: Thank you, Steve. We will see you next time on Security Now!.

Steve: Thanks, Leo.