Law Meets Internet

Description: This week Leo and I discuss Russia's hacking involvement in the U.S. election; that, incredibly, things get even worse for Yahoo; misguided anti-porn legislation in South Carolina; troubling legislation from Australia; legal confusion from the Florida appellate court; some good news from the U.S. Supreme Court; Linux security stumbling; why Mac OS X got an important fix last week; the steganography malvertising attack that targets home routers; news of a forthcoming inter-vehicle communications mandate; professional cameras being called upon to provide built-in encryption; Let's Encrypt gets a worrisome extension; additional news, errata, miscellany; and how exactly DOES that "I really, really promise I'm not a robot (really)!" non-CAPTCHA checkbox CAPTCHA work?

High quality (64 kbps) mp3 audio file URL: http://media.GRC.com/sn/SN-591.mp3
Quarter size (16 kbps) mp3 audio file URL: http://media.GRC.com/sn/sn-591-lq.mp3

SHOW TEASE: It's time for Security Now!. Steve Gibson is here. We've got a jam-packed show, lots of security news. Yahoo hacked; a billion accounts lost. What did the Russians actually do to hack our elections? And why the Florida court says, no, you've got to turn over your passcode as well as your fingerprint. It's all coming up next, and a lot more, on Security Now!.

Leo Laporte: This is Security Now! with Steve Gibson, Episode 591, recorded Tuesday, December 20th, 2016: Law Meets Internet.

It's time for Security Now!, the show where we cover your security and privacy online, the last episode, last fresh episode of 2016 for this guy, Steve Gibson, the man in charge at GRC.com, our security guru. I think you have disciples, Steve, who follow Steve's way. I know I'm a disciple, in the sense that, when I see a weird security practice or something new, I always ask myself, "What would Steven say?"

Steve Gibson: What would the Gibsonian response be?

Leo: What would Gibson do, WWGD. I also - I think that you've helped me kind of be a better consumer of technology because of that; you know? So thank you.

Steve: Well, I do know, for what it's worth, that I get a lot of really good feedback from

people who, as you said, they'll see some news and send me a note to make sure that it's on my radar and often say, you know, can't wait to hear what you think of this. Because as we've seen...

Leo: Mm-hmm, exactly,

Steve: Yeah. As we've seen, these things are complicated. It's funny, I was just thinking an hour ago about the Windows metafile issue.

Leo: Wow.

Steve: Which was controversial because it was clear to me, and to Mark Russinovich when he looked at it, that all it was, was an escape that would allow native code to be placed in the image. Now, and the problem is that people immediately go to how horrible that is from an exploitation standpoint. But it's important that it be put in context. Metafiles were created before networking. So this was a clever solution to what if the language of the interpretive metafile doesn't do something that we need? Oh, let's just let the metafile contain native code. You know, Google does that with Chrome. That's NACL, their native library that allows them with a great deal of control to do the same thing.

Anyway, so the point is that this stuff is not simple. And you know me, I love figuring things out and living in complexity. I just dig - I just jump in with both feet. And I've had the advantage of being here all along, watching this sort of all happen. So for me it's really - it's a symbiotic relationship. I love being able to look at these things and explain them to our listeners. And I'm just glad that so many people appreciate it.

Leo: We do. We do indeed. I look forward to - I've learned, you know, I have to say, in general TWiT's more educational for me probably than almost anybody because I'm here for all the shows.

Steve: And actually I do hear you using...

Leo: All the time.

Steve: ...some of the stuff that we've talked about here on the other podcasts...

Leo: Absolutely.

Steve: ...where there's overlap. So this week I called - I named this "Law Meets Internet" because the lead stories generally involve the struggle that I think we're just seeing the beginning of, of the Internet becoming, like, really a thing. It's important now. And in fact the first thing we need to talk about at least somewhat is this whole Russia hacking involvement because we can't just ignore it.

Leo: I really want to talk to you about this.

Steve: We can't ignore it completely.

Leo: Yeah, yeah.

Steve: But, so, I mean, it's clear that this is - like the Internet is really something. And so naturally there's confusion about how to handle the intersection of what started off as pure technology with life, and the way civilized societies have figured out to do this. Well, I guess even tribal societies. They have their sets of laws, too. So it's rules and regulations and dos and don'ts. And so we have a number of things there. We've got of course the Russian hacking involvement in the U.S.

We have to also talk about briefly, incredibly, it gets even worse for Yahoo. Who could have thought it could be worse than 500 million accounts breached? Well, yes, it can.

Leo: Yes.

Steve: Then we've got, speaking of legislation, we have some misguided anti-porn legislation that's been made in South Carolina; some troubling legislation in Australia; some legal confusion from the Florida appellate court that you were talking about on MacBreak Weekly; some good news finally on the patent scene from the U.S. Supreme Court. So those are all just sort of law things that happened in the last week. Some interesting problems with Linux desktop security, four new problems that have just happened recently. And again the press has drawn the wrong conclusion. It's like, oh, my god, Linux is no more secure than anything else. It's like, okay, slow down. Then also why last week's Mac OS X update was important. Because of responsible disclosure, we only found out now what got fixed. And it was kind of scary.

Then we have more news about the steganography malvertising attack that we discussed last week, and what the actual mechanism is that it's using. Some news of forthcoming inter-vehicle communications, which the National Transportation Safety, whatever it is, is mandating for, like, right now its proposed rules. Once it happens, it has a two model-year cycle. And then any new vehicles need to be able to talk to each other. So we'll cover that in more detail. And something I thought would really interest you, Leo, that you'll probably want to talk to Scott about, is professional cameras are being called upon to provide native, built-in encryption. Also Let's Encrypt has an almost foreseeable, yet still worrisome, extension that we need to talk about.

Then, after all that, we've got some additional news, some errata, some miscellany. And then I answer the question, well, or I try to, exactly how does that "I really, really promise I'm not a robot" checkbox...

Leo: Oh, good.

Steve: ...non-CAPTCHA CAPTCHA work.

Leo: I've wondered this a long time, so that's good.

Steve: Yeah. So I think a great podcast.

Leo: [Crosstalk] coming up. And the law meets the Internet.

Steve: Our Picture of the Week...

Leo: I love this.

Steve: ...I got a kick out of.

Leo: Yes.

Steve: I don't know where it came from. But I think everybody's probably heard that, like...

Leo: You don't go on Facebook. But Facebook is loaded with this kind of quiz thing.

Steve: Oh, okay.

Leo: And this could easily be a Facebook quiz.

Steve: Well, so remember, what was it, I think...

Leo: Your porn name.

Steve: Yes, that's what I was going for. I think it was your childhood pet and the street you grew up on, something like that.

Leo: Yes.

Steve: Which would make me Terry Overhill.

Leo: That's good - for an older porn star.

Steve: So, yeah. Well, and I would be.

Leo: By the way, don't give out your - some of these could be used in security questions. But I know you're smart enough. You would never use anything factual in a security question.

Steve: Exactly.

Leo: But that could potentially be another reason why they put these quizzes up; right?

Steve: Well, okay. And that of course is the point. So in that vein, this appears to be, for the people who don't have video, this is a form you fill out that says, "What's your Star Wars name?" And so they're apparently doing the same thing. It just says "Enter your Social Security number and mother's maiden name to find out."

Leo: Great.

Steve: And then I put a little caption down at the bottom, "Courtesy of P.T. Barnum."

Leo: Yeah. There's a sucker born every minute; right?

Steve: He was given attribution for that famous phrase. And so, and I looked at this, and I thought, you know, none of our listeners, I mean, they would find this humorous.

Leo: They'd get it immediately.

Steve: I'm absolutely sure that, I mean, remember, Leo, I know that you'll have probably seen ugly cars driving around the streets.

Leo: A few, yeah. Driven some myself.

Steve: Okay. Somebody bought that ugly car.

Leo: Yeah, yeah.

Steve: There was something called the "Thing" once.

Leo: Oh, god.

Steve: And I actually saw them on the road.

Leo: Yeah.

Steve: Someone bought a Thing.

Leo: That was a Volkswagen.

Steve: Yes.

Leo: And they ended up discontinuing it because, if you got hit in the Thing, it was composed of a bunch of panels. All the panels would pop off, and you'd be sitting there naked on the road in your chassis.

Steve: And so this is the lesson - oh, and we've also seen, like, unbelievable colors on cars. It's like, okay, somebody either bought that, or they did that.

Leo: Christmas sweaters. All you have to think is Christmas sweaters.

Steve: Yeah.

Leo: People wear those; right?

Steve: And somebody bought them.

Leo: Yup.

Steve: So it used to be on a hook. Now it's on them. So the point is that somebody is going to see this and go, oh, I can't - what's my Star Wars name?

Leo: My Star Wars name. I want a Star Wars name.

Steve: What was - let me look up my Social Security number and put that in, and my mother's maiden name. And, oh, wow. Anyway, yikes. Just got a kick out of that. Perfect Photo of the Week for us.

Okay, so Russian election hacking. Now, we're in a situation again where neither of us have any facts.

Leo: Right.

Steve: As you know, I'm not on the inside. I have no connections with the NSA or CIA or anything.

Leo: Seventeen intelligence agencies, of which I can only name three.

Steve: Yeah, it's like, 17.

Leo: Seventeen. Who are these people?

Steve: Clapper is still clapping around in there somewhere, too.

Leo: Yeah, yeah, he's in charge of the whole schmiegel.

Steve: But there is something we could say. And that is that, from the coverage that I have seen, it appears that - well, and in fact there is another, the news just broke today, and I may make it the subject for next - oh, wait, week after next podcast, the first podcast of 2017. And that's a massive Russian organized ring was found doing fake advertising, generating it's estimated between 3 and $5 million per day in ad fraud.

Anyway, the point is that sort of standing back from 10,000 feet, in general I get the sense that Russia, for whatever reason, has been expending a great deal of effort on the cyber front, that is, in all things cyber. Attacks, penetrations, scans of all kinds. We're often talking about .ru domains, and things are generally terminating back in Russia. And so, and as I understand it, they don't have a super energetic economy. It's not producing a lot except for, I guess, some natural resource-based stuff. So it's also something that is perfect for that kind of an economy, that is, cyber is, because it doesn't have a high entry cost, it scales well, and so forth.

So I guess I'm listening to all of this coverage surrounding did they, didn't they, what happened, what's this about, and also just all of the chomping at the bit that's going on. Everyone's just all in a big flutter. It's like, okay, we've been saying now for years on this podcast that everything is porous. That is, we have so much complexity has been added, and we're lagging years behind in finding bugs that are often years old, and that creates a moving window of opportunity. All of that says that the more you want to do something, the more you can - the more opportunities you can find.

That is, everywhere we look we see attack surfaces that can be leveraged. And apparently Podesta got his emails hacked by clicking on a phishing link which got something installed in his machine; and then they were able to say, oh, thank you very much, and look around. So stuff we've been talking about for years is happening. But also many of these organizations have subcontractors which may or may not be very good. Some of the stories we heard were that the FBI was, for example, notifying a subcontractor who did IT for the Democratic National Committee for months that their machines had been penetrated, and the IT guy didn't even take it seriously. He was part-time, and he didn't think it was the FBI. He thought it was just a prank call.

So, I mean, so it's a combination of human error, human factors. And of course ultimately software errors are the same thing. They're extensions of human mistakes that well-intended and intending programmers make which doesn't keep the software

from working, but it does create opportunities for bad guys to get in. And so I guess my take is I'm not, without any evidence, without any insider knowledge except if you just swept your arm across the last several years of this podcast, you would have to come to the conclusion that, when somebody wants enough to get into something, with the way things are today as we wrap up 2016, it's possible, from literally hook or crook. You can get in and...

Leo: Especially a targeted attack, a spearphishing attack, where they're going after a specific asset. That's very hard to defend against.

Steve: Correct, correct. I mean, I would say it's beyond hard at this point. I would argue it's probably impossible. With everything we've seen, we keep seeing gifted hackers able to penetrate whatever they want. Pwn2Own, every browser falls in the first hour. And it's like, ouch. And mobile phones do, too.

Leo: And it's presumed that a government attacker has access to all - unlimited resources, or virtually unlimited resources.

Steve: Yes. So if we imagine that Russia has decided by policy, sometime in the past, that the Internet is the best thing that ever happened for enabling them as a society, as a nation-state, to mess around with other countries, then you put - compared to, for example, what the U.S. spends on military, you put relatively tiny cyber resources behind a concerted effort, and all of the evidence would suggest that they can pretty much do anything they want to, if they try hard enough.

Leo: And it goes both ways. We're clear, I mean, why wouldn't we be doing exactly the same thing?

Steve: Right.

Leo: And there's a certain irony in the CIA saying, well, the Russians subverted our election, after that agency specifically has subverted elections with all sorts of covert actions over the last five or six decades.

Steve: Well, we were listening to Angela Merkel's phone.

Leo: Right.

Steve: She was a little annoyed.

Leo: That was the NSA. But the CIA guaranteed that we would win elections all over the world, and has for years. So, you know, you and I both are liberals and probably voted a little differently than the outcome. But I think it's a little overblown to blame

Russia on the outcome of the election, or to say, oh...

Steve: I didn't say that.

Leo: No, I know, I know you didn't say that. That's what's being said. And I kind of want to push back a little bit on that, not because I'm happy about the result of the election particularly, but just because it doesn't seem like, I mean, okay, getting into the DNC's email, I mean, if you said they got into the voting machines and changed the count, okay. But that doesn't seem to be what they're saying.

Steve: No, no. Well, and relative to whether the election was altered, as we know, you can't prove a negative. And so there's no way now, retrospectively...

Leo: Although we just saw today reports that said there doesn't seem to have been any widespread voter fraud at all.

Steve: Right, right. And I was thinking more in terms of biasing the electorate. On the other hand, you would argue that our own FBI was a substantial influence in that with the timing of what Comey understood, I mean, I completely understood the position he was in.

Leo: That you could argue, yeah.

Steve: He couldn't say nothing, or after the fact he would have been blamed for not saying anything and [crosstalk].

Leo: Well, that you could argue about. But I think in a way it's a disservice to point at the election results because what I would far prefer to hear our intelligence services saying is we have widespread evidence of Russian hacking in a variety of activities, and we need to do something about that. When you tie it to the election results, it makes it much more of a partisan issue.

Steve: Right.

Leo: That really doesn't - it's a disservice to the much larger issue. But I don't think they want to talk about how much we're doing. So I think they're reluctant to get into that larger issue, frankly.

Steve: Right. I think that's exactly true.

Leo: Yeah, yeah. I mean, I'm much, and we've talked about this before, worried

about hacking the grid, for instance. It would be fairly trivial for this nation-state to take our grid down. I mean, I'm a lot more worried about that.

Steve: UPS, baby, battery backup.

Leo: Yeah, yeah. Well, I'm not worried about my servers, anyway.

Steve: Yeah. So, okay. So anyway, that's really all I wanted to say. I didn't want to not discuss it ever, and I just wanted to say I'm sure it's no surprise to any of our listeners that this kind of thing is possible. We know, for example, that from Edward Snowden's revelations, how much our own government, our intelligence services are doing. It was an eye-opener. I mean, like all the project names, I mean, we had a field day for a year covering all of the disclosures that came out of that. And we also know that China is very active. We're very active. And we know in many other ways Russia is very active. And it's not hard to be active. And there's also people in their basements are able to hack things, too.

So, yeah, we have, I mean, what's interesting is that we're seeing, again, as the Internet intersects the real world, and I guess if nothing else the outcome and the issue of the U.S. presidential election is about as real as it gets, then suddenly people are saying, "Oh, wait a minute, this is really bad." It's like, yeah, okay, this has been going on for a long time, and it doesn't seem to have gotten on anyone else's radar in as significant a way as it finally has. So I just think this is all for the good because, as we know, security is hard. And you have to work at it if you want it.

And there are some places you really need it. I would argue messaging, eh, some people certainly need secure messaging. I don't have any particular need for it because I'm just, you know, arranging what time to meet friends for a meal. So I guess my point is it's variable. And something like, you know, the more important things are, the more people want them to be secure. The problem is we don't currently have an infrastructure that guarantees that. And as with all the other lessons we see is that, if we ever get there, it's going to take a while. And you and I, Leo, will have been long retired.

Leo: Yeah, yeah. I mean, let's get the intelligence agencies focused on what Russia is doing and maybe come up with ways to defend against that. That I'm all for. I'm all for.

Steve: Yeah. There's no way.

Leo: But I think you make an excellent point. I would hate - should we be fatalistic about that, then?

Steve: I think you could - we could call it that. Or you could call it realistic. I mean, look at the history.

Leo: Yes.

Steve: The history says your light bulb could be attacking someone.

Leo: Right.

Steve: I mean, we're not making this up anymore.

Leo: Truthfully, if I were Russia, I think that there will be much more damaging things you could do than what seems fairly minor, which is breaking into the Democratic National Committee and releasing its email. It seems like they could have done a lot worse had they really been strongly motivated.

Steve: Yeah.

Leo: Am I wrong?

Steve: Well, no. And that's the other problem is there's always a problem with attribution, and we're also not able to read people's minds. So like just recently the question was, or what was in the news was that Putin himself was directly involved. And of course no one will explain how they know that, but that's now what they're saying. Again, that stuff, I just sort of - I listen to it. I think, okay, well, that's interesting. Maybe that's true. But it isn't anything that is actionable. But what we do know is that everything has been built up in complexity on a fundamentally weak foundation. Lots of security intention. But complexity, as we've often said, is the enemy of security.

And what's happened now is that the Internet is becoming really important. And as we will be covering here in the next few stories, now our legislators, which is like the tool that bureaucracies use for trying to set limits and boundaries, that's now getting involved, which is always a little frightening. But first we find out that more than three years ago, in August of 2013, one - more than, actually, it's more than one billion user accounts at Yahoo were hacked. And so this news comes out since our last podcast. And I think, okay. Actually, it was Wednesday of last week.

And I think, okay. First of all, remember the old phrase, "Fool me once, shame on you; fool me twice, shame on me." If I refuse to learn from my mistakes, well, whose fault is that? Which is my way of asking what security-conscious person could possibly still be using Yahoo? That is, they've been sending up mushroom clouds every few months for the last year. And anybody who is concerned about security should be long since gone. And what's interesting is that, even after three months ago, in September, when they confessed to the 500 million accounts being hacked back in 2014, so that was only two years ago, they didn't force password resets and security question changes. Now they're doing so.

What we learned is that this most recent billion-plus account disclosure revealed sensitive user information including names, telephone numbers, dates of birth, hashed passwords, and unencrypted password reset security questions, you know, speaking of Terry

Overhill. So if anyone listening to this is still using a Yahoo email account, you have to ask why. And also, absolutely, the only thing you really have to do is make sure you're not sharing any of that information, your password reset security questions, your hashed password, which is who knows how three-plus years ago it was being hashed. It's hard to imagine it would have been secure. And make sure there's no overlap between that and any accounts you actually do care about. I could understand having a throwaway email account, but just be very careful with the way you use it. And unfortunately we're also seeing ways that someone getting into an email account can then escalate their attack through various other means.

So again, it's hard to believe anyone would still be there. What we hear in the coverage or read in the coverage is that - and here's the takeaway. Yahoo was ignoring the pleas from their security IT people for years. They were deliberately giving - they were like, yeah, yeah, yeah, fine, we don't want to inconvenience our users. So what's one of our other major mantras on this podcast is security and ease of use are always at odds with each other. So what we're seeing is the downstream consequence of a huge company with public-facing accounts, Yahoo Mail, that also has the policy of not doing what their security people are urging them to do for years because they don't want to ruffle any of the feathers of their users. So billions, billions of email accounts.

Leo: I always suspected it because we'd get all these calls on the radio show from people whose Yahoo account has been hacked.

Steve: [Crosstalk].

Leo: Yeah. And it wasn't that they had bad passwords. I mean, it seems like they were very vulnerable. The other thing I would say is - I would like to propose and hear what you think is one probably shouldn't delete one's Yahoo account because you don't want to give up that mail address, especially if you ever use it for password resets or anything, because Yahoo reassigns those addresses. You don't want somebody else to get your email address. You want to keep it, but just keep it dormant and kind of - I would sanitize it of any personal information.

Steve: Yeah. And we covered this problem with Yahoo a couple years ago.

Leo: Yeah.

Steve: That whole issue of abandoning an account. I think what they were doing was, if you had not logged in for some length of time, they would send an email into that account, which of course you wouldn't see because you weren't ever bothering with it. But then they would unilaterally make that re-available. They were recycling these long-dormant email accounts. And they got a big slap on their hand for that because so many people were using their Yahoo Mail as recovery for other things. So that was my point about how unfortunately our experiences online are interlinked.

So you might use, you might have registered once your Yahoo Mail as your backup email for recovery of something that you do care about. And if a bad guy then got a hold of your Yahoo Mail, they could, I mean, exactly as designed, use their control of your backup password recovery mechanism on an unrelated service to gain access to that

service. So, yeah. Anyway, so I remember when we discussed it before the notion was, well, how do we remember to log in every quarter just to say, "Hi, Yahoo. Don't expire this account. Unfortunately, I used it once, and I don't ever want anybody else to have it." You know, wow. And it's very much like the problem of a domain name that gets lost, and then squatters sit on it because there's going to be some traffic that's going to wander in and get exposed to what's there.

Okay. South Carolina legislation has been proposed by Bill Chumley, State Representative Bill Chumley. He has filed a bill to require computer sellers to install digital blocking capabilities on computers and other devices that access the Internet to prevent the viewing of what the bill says "obscene content."

Leo: What?

Steve: Yes. Yes. The proposal - oh, it gets better. The proposal would also prohibit access to any online hub that facilitates prostitution and would require manufacturers or sellers to block any websites that facilitate trafficking. So he's saying, if you're going to sell computers into South Carolina, our citizens must be protected. But, now, here's where it starts to get sort of strange. Both sellers and buyers can get around the limitation, for a fee. The bill would fine manufacturers that sell a device without the blocking system, but they could opt out by paying $20 per device sold.

Leo: What?

Steve: And even - I know. Even more oddly, buyers...

Leo: Who gets the 20 bucks?

Steve: Well, uh-huh, that's exactly the right question. Which is why I think this whole thing seems a little fishy. We're about to get there. Buyers could also verify - buyers could verify their age and pay $20 to remove the filter. It's like, okay, what? The money collected would go toward the Attorney General's Office Human Trafficking Taskforce.

Leo: This is just goofy. Whoever this guy is, it's not - if it passed, then we should talk about it. But this guy is a crackpot.

Steve: Chumley's bill has been referred to the House Judiciary Committee.

Leo: Yeah, where they're going to bury it.

Steve: And so my comment was the weird "adults may pay $20 and have the filter removed" gives the legislation more the character of a fundraising extortion racket for the Human Trafficking Taskforce. Which, I mean, that's a great taskforce, but still it's like, you know, what? Yeah. So anyway, I just - this just popped on the radar, and I thought, okay, this is too crazy.

Leo: This is a mish-mosh.

Steve: And as you said, Leo, until a gavel drops, it's just, you know, we've seen lots of legislation come and go that never got any...

Leo: It's grandstanding. But to be fair, there is, you know, they have filters in the U.K. I think they have filters in Australia that are mandated.

Steve: Well, that's our next story, actually.

Leo: Okay. Go right ahead. I didn't mean to slow you down.

Steve: Meanwhile, an Australian court ruled on Thursday, December 15th, last week, that the Pirate Bay and a collection of other sites must be blocked by Internet service providers. And I have, in the link in the show notes - oh, I should mention show notes are already online for anyone who wants to follow any links there. So I'm now working to have them always on the Security Now! page, GRC.com/sn. The first item there is this week's podcast. And I will continue to try to get them up immediately.

So the measures have not been implemented yet, but this just happened on the 15th. ISPs have 15 days, that is, by year's end, by New Year's, Internet users will be blocked by default by ISPs. Now, what's interesting is that Google's data shows a large surge in searches for the acronym or the abbreviation "VPN," and VPN services have reported a significant increase in interest from Australia. Justice John Nicholas of the Australian Federal Court has ordered Australian ISPs to block The Pirate Bay, Torrentz, TorrentHound, IsoHunt, and SolarMovie, and many proxy and mirror services of those that, as the coverage says, marks the start of a mass Internet censorship Down Under.

And we're not surprised to find that this is the result of a case brought by Roadshow Films, Foxtel, Disney, Paramount, Columbia, and 20th Century Fox. So stakeholders in copyrighted material are saying we want access shut down to these sites that exist purely for the sake of piracy. And more than 50 ISPs are now required to start barring subscriber access to these sites. And there's also, I did not go into the details of the legislation, but it's been in the works for quite a while. And since it began, Torrentz, TorrentHound, and SolarMovie have already shut themselves down. But the judgment continues to name them in case they might return.

And the ISPs are given some latitude about how to actually perform the blocking. They could use DNS, so like intercept and remove those from DNS services available in Australia, so you just can't get the IP addresses; or block the physical IP addresses as they try to exit from Australia; or also filter and block on the URLs; or any alternative means which are approved by this coalition of copyright holders.

And this is a little worrisome now, too, because this legislation essentially puts this group of copyright holders into the loop, like permanently in the loop for, like, can we block them this way? Is this okay? And there's apparently a $50 charge to the copyright holders per blocked site. So it's not free for them to keep adding them, but there is a mechanism also for them to routinely go back and say, okay, and now we want you to block this, this, this, this, this and this and this. So, wow. It just seems like a very

slippery slope.

Leo: And a template for what's going to happen here in the next year, I would predict.

Steve: You really think so.

Leo: Yeah.

Steve: Yeah, wow.

Leo: Well, I'm sure the Motion Picture Association...

Steve: Will be pushing.

Leo: ...will be pushing hard, yeah.

Steve: Yeah, who was it, it was a leading Democrat who went to work for the MPAA. I can't remember his name now. I was disappointed.

Leo: I know what you're talking about, yeah.

Steve: Yeah.

Leo: It's sad, I mean, so we have a mess. It's a revolving door.

Steve: Well, and again, here again is real world meets Internet. And unfortunately this is an attempt to apply controls that the Internet was not designed to provide. We were, in the early days of innocence, we were all celebrating freedom of expression and the openness and how it was uncensorable and all that. And it's like, well, it isn't designed with any of those things in it. But you can do things like filter. And the problem of course is VPNs. If you're allowed to run a VPN tunnel outside of Australia, then that bypasses any border protection. Just, bang. So no wonder searches for VPNs are on the upswing.

Leo: By the way, it's Chris Dodd, the chatroom says.

Steve: Chris Dodd, yup, that's exactly who I was trying to think of.

Leo: You know, the thing that really worries me is you get then this escalating battle

between the people who want a filtered government at the behest of the copyright holders, and users. And it really is bad in general. Then VPNs get blocked; right? And then, I mean, ultimately it ends up being not just bad for people who want to steal movies. It's terrible for everybody.

Steve: Well, and as we've talked about, there are also - there's a real problem with false positives.

Leo: Right.

Steve: I often hang out next to a Verizon that has free open WiFi. And it's great WiFi, so it's there, and I'm not doing anything I'm trying to hide. So I'll use their WiFi. And sometimes, because I'm also researching medical stuff, I'll get a block page. And it's like, what? Because, of course, there's some overlap. Maybe the page mentions testosterone or something, who knows what. But, I mean, for whatever reason, this is like, you know, it's PubMed. It's our government's medical research archive. And Verizon's nanny gate isn't letting me see a page because it's like, oh, no, no, can't get there. It's like, okay.

So unfortunately, as we know, it's an imprecise technology. And even the definition, you know, who was the judge who famously said, "Well, I can't define pornography, but I know it when I see it." It's like, okay, well, how do you write a law around that? Yeah.

Leo: It's not, you know, what it is, what the problem is, is that I'm not for stealing things. And that's one of the things laws and government do is they punish thieves. And that's an appropriate thing. I don't want somebody to come into my house and steal my stuff, either. But the problem is the people who are making these laws are generally, in fact, entirely ignorant of how the technology works, the technological issues, and the long-term consequences of the things they ask for. And that's what worries me more is the ignorance among lawmakers.

Steve: That and we have a history now of the copyright holders being very overreaching. As we've often quoted on this podcast, they tried to prevent home videotape recording under the argument that it was purely for piracy; that the only reason anybody would want that was piracy. Fortunately, that didn't happen, and we're able to record content for our own consumption at home, but against the desires of the media providers. And of course we went through the same thing with the DVD, that was all ridiculously encrypted, which lasted all of a week or two because you can't do that. There's just no way to do that securely. But again, a huge amount of effort was put in. And it just ended up inconveniencing everyone and producing no effective result.

Leo: I said 10 years ago that ultimate freedom fighters would end up being hackers, people who know how to use technology and can protect our freedom online. So everybody needs to start learning this stuff now. If you want to preserve freedom, learn technology.

Steve: Yeah.

Leo: Otherwise it's used against you.

Steve: So we've often spoken of the difference between something you know and something you have relative to recent court decisions. And I heard you talking about this on MacBreak Weekly just now, Leo, and our listeners need to hear it, too, because a recent decision was reversed on appeal in Florida which changes, I mean, depending upon how this goes, like what future this has, this argues that, unlike what had been believed up until now, that a suspect who is blocking the acquisition of evidence by not divulging something they know, their passcode, up until now, as we've discussed often, that was regarded as testimonial. And so the Fifth Amendment to the Constitution, which protects against self-incrimination, was protection. That is, you could not compel testimony against oneself, thanks to this Fifth Amendment.

So last Tuesday a Florida appeals court ruled, in a case of a man suspected of voyeurism, that police may lawfully compel the disclosure of a mobile device's passcode for the purpose of searching it for incriminating evidence. Okay, so a little bit of context here. The guy's a creep. No one's on his side. His name is Aaron Stahl. He was arrested after a woman who was shopping in a store saw him crouch down and extend an illuminated cell phone under her skirt, according to court records. When she confronted him, Stahl told her that he had dropped his phone. He ran out of the store when she yelled for help, but police were able to identify him using his car's license plate number as he made his getaway. He was later arrested for something known as third-degree voyeurism. Sounds like first-degree to me, but...

Leo: What would first-degree be?

Steve: I don't know what the degrees are. Sure does seem premeditated, if nothing else. In a police interview Stahl initially gave verbal consent to a search of his cell phone - so they said, "Will you let us search your cell phone?" He said, "Yeah, okay" - which was an Apple iPhone 5, but subsequently withdrew his consent before telling the police his four-digit passcode. Once police obtained a warrant for the phone, they were unable to access the photos on the phone. Okay, again, no one's on this creep's side. We're on the side of civil liberties and the question of does the Fifth Amendment still apply. So there's what I described in my notes as "tortured logic." There's some rather tortured logic at this.

So at trial the judge denied the state's motion to compel Stahl to give up his passcode, finding that it would be tantamount to forcing him to testify against himself, in violation of the Fifth Amendment. But subsequently the Florida Court of Appeals Second District reversed that decision, actually it was last Wednesday, finding that the passcode is not - this is what's strange, the wording of this, but I was careful to get it: "The passcode is not related to any criminal photos or videos found on the phone." Okay, meaning that so somehow the fact that you need the passcode to divulge them, a three-judge panel disconnected those. So Judge Anthony Black, writing for this three-judge panel, said: "Providing the passcode does not 'betray any knowledge Stahl may have about the circumstances of the offenses.'"

Leo: This is about the right to not self-incriminate. So they're saying giving the passcode is not self-incriminating.

Steve: Right, right, exactly. It's not like divulging the photos themselves.

Leo: Right. That would be incriminating yourself. Give us the photos. You have the right to say no.

Steve: So they have a search warrant they cannot execute because he's blocking it.

Leo: Right.

Steve: So the text goes on to say: "Thus, compelling a subject to make a nonfactual statement that facilitates the production of evidence for which the state has otherwise obtained a warrant based upon evidence independent of the accused's statements linking the accused to the crime does not offend the privilege." In other words, they assert that a passcode is not testimonial; it's surrender. So you're compelling a person to surrender something, not to testify. And of course, as we know, it was the Supreme Court decision back in '88, Doe v. the U.S., where Justice John Paul Stevens wrote something that we've often repeated, and you mentioned in the previous podcast, Leo, that a person may be "forced to surrender a key to a strongbox containing incriminating documents, but cannot be compelled to reveal the combination to his wall safe."

So what we have now is an appellate court essentially reversing that standing U.S. Supreme Court decision. Okay. So I guess maybe this will go back up to a higher court. Maybe they'll just - I don't know what the mechanism is. If the U.S. Supreme Court says we've already ruled on this, does the appellate court decision stand? I don't know. Anyway, we'll keep an eye on it because this has been an interesting point for people to say, oh, no, you don't want to use your thumbprint. You want to use a passcode because that's something you know, not something that you can be compelled to produce. And you can be held down and forced to put your thumb on the phone. No one can make you tell something you know; but you can be held in contempt of court, I'm sure you would be, and then jailed until you surrender the information.

Leo: Yeah, and as we pointed out, I mean, the issue isn't so much the passcode is that you don't want to let the government pluck things from your brain because that kind of encourages torture, or compelled or forced confessions. And so that's - I'm no lawyer, but I would guess that that's the reason for the Fifth Amendment. And that's what the Supreme Court would have to decide. I don't think it's obvious what the right answer is, by the way. I can understand the case on the other side, as well.

Steve: Yeah, yeah.

Leo: Very interesting.

Steve: We do have some good news, thanks to our friends at the EFF. The U.S. Supreme Court has agreed to hear a case that could end the famous Texas grip on patent cases. "In the case TC Heartland v. Kraft Foods, that case effectively asks the court to decide whether patent owners" - which as we know are unfortunately all too often not patent users. They are trolls, patent trolls that collect patents purely for prosecutorial purpose,

in order to squeeze money out of people, much like the podcast patent that was hanging over TWiT's head for a while.

Leo: And would have gone to Tyler, Texas.

Steve: Yup, "to decide whether patent owners can sue in practically any corner of the country." The EFF supported the position and side of TC Heartland, who was the petitioner, at the Court of Appeals for the Federal Circuit and as well in asking the Supreme Court to hear the case. "The petition to the Supreme Court became necessary after the Federal Circuit issued a disappointing decision that maintained the status quo." So it's like, okay, we need to escalate this. And the good news is the Supreme Court has said, we agree. Bring it in front of us.

The current law, as it is now and was unfortunately just recently re-upheld, "allows patent owners to pick and choose between federal courts, often opting for courts that are perceived to have rules and procedures favorable to their position. The result," writes the EFF, "has been astounding. Last year almost half, 45% of all cases were filed in a single Eastern District of Texas, a rural part of the country that has no major technology industry." Just a well-fed judge.

Leo: Well, it's not even that because what you want is a jury that is well disposed to protect the little guy, which is how these non-practicing entities position themselves. I invented something, and now the big company came along and took it from me. So you don't want people who work for big companies.

Steve: Right. And we've talked about this in the past. There's a stadium in that town that Samsung fully supports. They've got, like, Samsung banners and flyers, and they...

Leo: They're trying to win hearts and minds.

Steve: Exactly. They're saying, "We're good. Please think of us in a good light," because they recognize they're vulnerable. They've just been raked over the coals by this ridiculous county.

Leo: So I can now talk about this because during this - we did, of course, get approached by the podcast troll. And that patent was invalidated by the Patent Office, so the whole issue is over, thank goodness. And they did go after, as you know, Carolla, and Adam Carolla fought it. Good for him. We would have fought it, as well. I would not - they asked for $2.5 million, I think, something like that. And we just laughed. And we would have fought it, but they didn't end up coming around to suing us. But what we did do is engage an attorney who practiced in East Texas, who had done patent cases in Tyler, Texas, who knew the judge. Because there is one judge who's the one everybody wants, and this is where it would have been - this is the jurisdiction it would have been tried in.

Steve: He's busy.

Leo: He's very busy. But again, it's kind of - it's positioned as, no, we're trying to defend the rights of the little guy, you know, the guy who invented intermittent windshield wipers, against the big bad corporation stealing his idea. And that's something juries really eat up. So our strategy - we had several meetings with our attorneys. And our strategy would have been to go down there, have a barbecue, have all the churches that have podcasts, have all the little podcasts from that area come and meet people, and bring our viewers down because the idea would be to say, "We're not the big guy. We're like you. And podcasting is how normal people get a voice in the world." And I think that would have been actually a good strategy, but we never got to exercise it.

Steve: Well, and there's so much wrong with the system because, for example, none of the money spent is recoverable.

Leo: Right.

Steve: It's all just gone.

Leo: Right. Well, it goes to the attorneys, but that's the point.

Steve: Yeah, exactly.

Leo: It doesn't go to the - usually the patent, the guy who came up with the patent has sold out long ago to these non-practicing entities.

Steve: Right, right, right. So anyway, the EFF concludes, saying: "We're glad to see that the Supreme Court has agreed to hear this important case that could significantly curtail some of the worst actors in the patent game. EFF will be there to urge the Court to restore balance and fairness in patent litigation." And I say again, for the umpteenth time, yay to the EFF. Thank you. Thank you.

Leo: Well, it was the EFF got that podcast patent overturned. We were thinking of, and we had decided not to do this because it's risky, they had decided to pursue what's called an inter partes appeal to the Patent Office. And the reason it's risky is if the Patent Office rules...

Steve: Affirms.

Leo: Affirms the patent, of course that goes right in front of the jury. Look at this. These guys tried to get the patent overturned, and the Patent Office came back again and said it's a good patent. So we didn't want to do it. But the EFF decided it was a good thing. They had prior art and so forth. And so they felt like they had a good shot at it. I donated money, a lot of people donated money to the EFF to

pursue this, and they won. And that really eliminated the whole thing and got the patent overturned, and they won. But it's why I also continue to donate every month to the EFF, and I think everybody should. It's a great organization.

Steve: And I think I've mentioned before that the last time I ever agreed to serve as an expert witness was in a suit between Princeton Graphics Systems and NEC over the MultiSync, which Princeton Graphics alleged was infringing on their patent, and NEC was fraudulently making claims that were unsubstantiated. And so I thought, this sounds interesting. And so I said yeah. And I agreed with NEC's position, so I let them hire me and explain to the judge, as I do, so clearly, so carefully, so that the fly on the wall understood what was going on here with its two neurons. They had been synchronized. And the decision came down the wrong way. And I thought, okay, you know, I'm not here to earn money.

Leo: Never again.

Steve: I'm here to help the good guys, and it didn't work. So I thought, screw this. I'm not doing this anymore.

Okay. So on the topic of everything is porous, Linux is in the crosshairs. A neat security researcher, Chris Evans, whom we've spoken of before, I'm not sure if he's employed by Google or affiliated. He does on his site refer to his buddy Tavis Ormandy, and of course we know Tavis well, of Google. Chris has been playing recently with the GStreamer - okay. Are you sitting down, Leo?

Leo: Yes.

Steve: Media library. And we know what a problem Android had with its media library. Turns out GStreamer is the de facto media processing pipeline which is open source, multiplatform, present in most Linux distros by default, and makes Stagefright look like a good thing. So this is just the beginning of taking a look at it.

So about a little over a month ago, November 15th, Chris posted - he has a blob. Blob. I did write "blob." [Indiscernible] my own notes. A blog called "Scarybeasts." It's scarybeastsecurity.blogspot.com. And so his posting on middle of November was - and he had the tags "0day" and proof-of-concept, "PoC." He said: "Risky design decisions in Google Chrome and Fedora desktop enable drive-by downloads." Meaning you just visit a web page, and in the background, if you visit it with Chrome, Google's Chrome browser, on Fedora, it downloads files, and I think it runs them.

So his overview says: "A confluence of risky design choices, combined with various implementation issues, makes drive-by downloads possible with Google Chrome on Fedora. First, Chrome will automatically download files to a user's desktop with no confirmation." Oops. "Fedora's tracker software will auto crawl downloaded files to index them, including media files. Three, the GStreamer framework, as used to handle media in the Fedora desktop, has questionable," he writes - and then in the next two blog postings we're going to learn just how much that is true - "implementation quality from a security perspective. Four, the tracker component responsible for parsing media files does not appear to be sandboxed," as in, for example, with security-enhanced Linux,

SELinux.

And, finally: "The Fedora default desktop install includes a range of fairly obscure media decoders that confer risk, but are not necessary for a thorough desktop experience." Which is Chris's polite way of saying there's a bunch of crap in there, installed by default, that few if any people will need, but which expands the attack surface needlessly and dramatically. So basically - and he goes into great detail afterwards. But that's the gist of this. So a drive-by file download vulnerability. And it's not sandboxed. And apparently this tracker indexing then allows you to leverage problems with GStreamer in order to essentially execute content on the system in the security context of GStreamer and/or tracker. Not good.

A week later, Chris is back, on the 21st of November. This one's tagged "0day" and "exploit." And he says: "Advanced exploitation: A scriptless zero-day exploit against Linux desktops." And then in his overview he says: "A powerful heap corruption vulnerability exists in the GStreamer decoder for the FLIC file format. Presented here," he writes, "is a zero-day exploit for this vulnerability. This FLIC decoder is generally present in the default install of modern Linux desktops, including Ubuntu 16.04 and Fedora 24. GStreamer classifies its decoders as good, bad, or ugly. Despite being quite buggy and not being a format at all necessary for a modern desktop, the FLIC decoder is classified as 'good,' almost guaranteeing its presence in default Linux installs. Thanks to solid ASLR/DEP" - that's, as we know, Address Space Layout Randomization and Data Execution Prevention - "protections on some modern 64-bit Linux installs and some other challenges, this vulnerability," he writes, "is a real beast to exploit."

But that doesn't stop Chris. "Most modern exploits defeat protections," he's writing, "such as ASLR and DEP by using some form of scripting to manipulate the environment and make dynamic decisions and calculations to move the exploit forward. In a browser, that script is JavaScript," he says, "or ActionScript" in the case of Flash. "When attacking a kernel from user space, the 'script' is the user space program. When attacking a TCP stack remotely, the 'script' is the program running on the attacker's computer." That is, remotely over TCP. He says: "In my previous full GStreamer exploit" - and this was something I didn't cover because it wasn't quite as on point - "against the NSF decoder, the script was an embedded 6502 machine code program."

Leo: What?

Steve: Well, it's because that was the chip in the Nintendo something or other. And so they were emulating, in order to run Nintendo stuff, they were emulating the 6502 famous processor technology chip. And so he was able to abuse essentially the 6502 interpreter in order to leverage an attack. And he says: "But in order to attack the FLIC decoder, there simply isn't any scripting opportunity. The attacker gets, once, to submit a bunch of scriptless bytes into the decoder, and try to gain code execution without further interaction."

And he writes: "And good luck with that. Welcome to the world of scriptless exploitation in an ASLR environment. Let's give it our best shot." Which is the beginning of a post where he shows how he did it. And, you know, this is somebody you want on your side. This guy has - and he looks like he's about, now, I don't mean this in any negative way, Chris, but it looks like he's about 12. So it's like, if he's not already working for Google, everybody should go try to hire this guy because he's got some serious skills.

So what all that means essentially is that he figured out how to feed the FLIC decoder

interpreter essentially a program which was complex and would do what he needed it to do by writing code in this metafile that the FLIC interpreter is going to interpret in order to get it to do this work. And it's like, okay. Wow. I mean, so here's a classic example of, if you want something bad enough, our current systems are replete with opportunity. Most people can't do that. Somebody good enough, I mean, there's other lower hanging fruit, probably. Chris is bored by low-hanging fruit. He wants to get a trampoline with stilts in order to reach up high enough in order to pluck this thing. But major skills.

And, finally, this brings us to last Tuesday, a week ago, when he posted most recently - apparently he'd been working on this Super Nintendo thing for a while. And all I did is just have his little quick TL;DR, which is a "full reliable 0day drive-by exploit against Fedora 25 and Google Chrome by breaking out of Super Nintendo Entertainment System emulation via cascading side effects from a subtle and interesting emulation error." Then he says, "Very full details follow."

So again, I have links to all of Chris's blog postings. If anyone is interested in the mechanics of this kind of serious, roll-up-your-sleeves reverse-engineering, this is the guy to go read. And you might want to just follow his blog because he posts every, you know, his latest exploits and exploitations at scarybeastsecurity.blogspot.com.

Leo: Now, they said in the chatroom that these exploits were patched before they became public. Or is that not the case?

Steve: You know, I didn't follow up and find out. I would be surprised if not because Chris is nothing if not responsible. So I'm sure that's - but again, these were there until Chris found them.

Leo: Right. But this is why open source works. I just want to point out, you know, people going, oh, this is a terrible thing, well, but this is why it works. If it gets patched right, then this is all good; right?

Steve: Yeah, doesn't do any better than closed source. It's all the same, Leo.

Leo: Well, I guess.

Steve: This is all, you know...

Leo: You can at least look at the code and look for the flaws in the code.

Steve: But it also allows you to look for exploits in the code. So, I mean, it's a double-edged sword.

Leo: Yeah, I guess, yeah.

Steve: Yeah, no, I mean, and that's what people do. They go over the code to find it.

And so, for example, when Microsoft releases updates, people have to reverse-engineer the patch in order to figure it out. There's no reverse-engineering needed here. You just look at the open source. I mean, I think...

Leo: By the way, GStreamer, I think, is not open source, come to think of it. I think it's proprietary. No? Is it non-free?

Steve: I looked on Wikipedia. It's got a full Wikipedia page. I think it's completely open, but I'm not sure.

Leo: Oh, okay.

Steve: I know it's multiplatform. And I did mention four vulnerabilities. Those were three. And this is just quick. There's another researcher, who knows Chris, recently posted his. And I guess Chris is probably a Fedora user, so that's why his exploits tend to be Fedora-based or aiming, not that other distros wouldn't have the same problem. He says his was reliably compromising Ubuntu desktops by attacking the crash reporter.

And he just writes: "In this post I'll describe how I found a remote code execution bug in Ubuntu Desktop which affects all default installations from Quantal on. The bug allows for reliable code injection when a user simply opens a malicious file." Okay, so the user has to take action. "The following video demonstrates the exploit opening the Gnome calculator. The executed payload also replaces the exploit file with a decoy zip to cover its tracks. Full source code for this exploit is available on GitHub."

And he says: "This research was inspired by Chris Evan's great work on exploiting client-side file format parsing bugs in the GStreamer media library on Ubuntu. We will look for other default file handlers on Ubuntu which may be vulnerable to exploitation. I'm not," he writes, "a binary exploitation guru like Chris, so instead we'll try to find bugs which are exploitable without memory corruption."

So again, our systems are complex. We want them to do everything. And there is legacy code that predates - in many cases there's legacy code that predates a manic concern over security. But even since then mistakes get made. So the new cycle here is, as we've been discussing recently, is that problems are found; they're responsibly disclosed; they're fixed in a timely fashion; and, hopefully, as we move forward with a heightened appreciation for security, we'll be making fewer mistakes than we fix. So the count of unknown vulnerabilities drops over time.

And ultimately I think we just need to scrap this entire model. Everything we're doing is like the way firewalls used to be of permit all and block known problems. It was easy to flip the firewall model around. We're basically still using an architecture from the first computers with relays and tubes. Nothing has changed. Our systems are fundamentally exploitable because of the way they're designed.

And I think we're getting to the point now where we have enough excess power, you know, this all comes from the fact that computers have never been able to be as fast as we needed them to be, so we just did the fastest possible solution. Having them operate in a way which is fundamentally secure rather than fundamentally insecure, they will not be nearly as efficient; but they will be potentially bug-free, that is, in terms of this kind of exploit. I know there's research in labs going on now saying we just have to scrap

everything we have. It's gotten us to this point, but it's just really becoming a problem.

So, and we learned also, just to sort of share the wealth, that last week's patch of Apple's Mac OS X closed a problem that we've discussed for years, in this case with the fact that Thunderbolt gave external peripherals that were attached to the Macs DMA access. After the patch was in place last Thursday, the guy who figured this out told us what he had found. He wrote: "MacOS FileVault 2 let attackers with physical access retrieve the password in cleartext by plugging a" - and in this case he quotes a $300, but it certainly didn't, I mean, just because it was available for $300 - "$300 Thunderbolt device into a locked or sleeping Mac. The password may be used to then unlock the Mac to access everything on it. To secure your Mac," he writes, "just update it with the December 2016 patches. Until then anyone, including but not limited to your colleagues, the police, the evil maid, and the thief will have full access to your data as long as they can gain physical access, unless the Mac is completely shut down. If it's sleeping, it is still vulnerable."

So he poses the question in his coverage, how is this possible? He writes: "At the very core of this issue there are two separate issues. The first is that the Mac does not protect itself against Direct Memory Access attacks before macOS is started. The EFI BIOS which is running at this early stage enables Thunderbolt, allowing malicious devices to read and write memory directly. At this stage macOS is not yet started, but it resides on the encrypted disk, which must be unlocked before it can be started. Once macOS is started it will enable DMA protections by default."

So the point is there is a little window of opportunity after the motherboard BIOS boots, which by default enables the Thunderbolt hardware interface. And until the macOS has booted, which is then able to apply the DMA memory restrictions that we've talked about in the past as a means of thwarting this kind of DMA access, there's a little window there, a gap.

"The second issue," he writes, "is that the FileVault password [oops] is stored in cleartext in memory and is not automatically scrubbed from memory once the disk is unlocked. The password is put in multiple memory locations, which all appear to move around between reboots, but remain within a fixed memory range." Obviously making that range searchable. "This makes it easy to just plug in the DMA attack hardware and reboot the Mac." So the reboot doesn't wash away what's in RAM. So the reboot then allows that window to give the plugged-in device access before macOS starts to restrict its access, but while the previous content of RAM is still present.

So he says: "Once the Mac is rebooted, the DMA protections that macOS previously enabled are dropped. The memory contents, including the password, is still there, though. There is a time window of a few seconds before the memory containing the password is overwritten with new content."

So this was responsibly disclosed. Back last summer, at the end of July, the issue was discovered. At DEF CON 24, in early August, on the 5th of August, PCILeech is what the hardware was called, was presented. But although known, the FileVault issue was not mentioned. Ten days later, Apple was notified. The day after that, on August 16th, Apple confirmed the issue and asked to hold off on disclosure. And then on the 13th, one week ago, Apple released macOS 10.12.2, which contains a security update. And he verified that it worked on his MacBook Air.

And then his conclusion gives props to Apple. He wrote: "The solution Apple decided upon and rolled out is a complete one, at least to the extent that I've been able to confirm. It is no longer possible to access memory prior to macOS boot. The Mac is now

one of the most secure platforms with regard to this specific attack vector." So as often, it's secure now. It wasn't before. But it got responsibly handled. Apple got the fix in place for something that could have been done just with a physical hardware device that was able to rapidly scan memory, find the plaintext password that had been left in memory after the machine had been suspended or put to sleep.

We learned also more details about the steganography - there's widespread agreement that I made up the term "steganometry."

Leo: Oh, too bad, I liked it.

Steve: Well, you know, so that means measuring steganography, steganometry.

Leo: Yeah, there you go. See? Yeah.

Steve: How big is your steganography? Well, that would be steganometry. So we learned more about what's going on, and it's chilling, so I wanted to share it. A security firm, Proofpoint, has dug way into this. And in fact there's a graphic on their site, Leo, you might want to bring up just because it's sobering in terms of the sophistication of this attack. And it's sobering because it demonstrates to what length bad guys are willing to go in order to get what they want. And so it just sort of says, you know, this is not the script kiddie anymore. So you're now showing the graphic of the back-and-forth protocol that this attack uses in order to achieve its ends.

Leo: It's so complex I can't fit it all on the screen.

Steve: I know. "Proofpoint researchers have reported frequently this year," they write, "on the decline in exploit kit activity," or they call it "EKs." So they say: "EKs, though, are still vital components of malvertising operations, exposing large numbers of users" - and we're talking millions - "to malware via malicious ads. Since the end of October," they write, "we have seen an improved version of the DNSChanger exploit kit used in ongoing malvertising campaigns. DNSChanger attacks Internet routers via potential victims' web browsers." Okay, so think about that. DNSChanger attacks Internet routers, meaning local, the users' routers, their LAN routers, their own 'Net routers, via potential victims' web browsers.

"The exploit kit does not rely on browser or device vulnerabilities, but rather vulnerabilities in the victims' home or small office routers. Most often, DNSChanger works through the Chrome browser" - and of course we now know that's the majority browser on the Internet, so yeah. If it's not multibrowser, you're going to choose Chrome - "on Windows desktops and Android devices. However, once routers are compromised, all users" - remember, because this is changing the DNS which all devices on the LAN will then use, which is why this is such a devastating attack - "all users connecting to the router, regardless of their operating system or browser, are then vulnerable to attack and further malvertising.

"The router attacks appear to happen in waves that are likely associated with ongoing malvertising campaigns lasting several days. Attack pattern and infection chain similarities led us to conclude," they write, "that the actor behind these campaigns was

also responsible for the CSRF (Cross-Site Request Forgery) SOHO Pharming operations in the first half of 2015." They write: "However, we uncovered several improvements in the implementation of these attacks, including external DNS resolution for internal addresses; steganography to conceal an AES key used to decrypt the list of fingerprints" - these are router fingerprints - "the default credentials and local resolutions; the layout for the commands sent to attack the targeted routers." And get this: "The addition of dozens of recent router exploits. There are now 166 fingerprints [that this thing recognizes], some working for several router models, versus [a year ago] 55 fingerprints."

They say: "For example, some like the exploit targeting the Comtrend ADSL Router were a few weeks old, [dated] September 13 when the attack began around October 28." Meaning that there are resources being put into this. The moment a new local router vulnerability is found - remember, this is not a remote vulnerability. This is inside your network because you've got routers typically with web interfaces that allow you to log onto them. This thing figures out what router you've got with 166 different fingerprints and knows the default credentials for logging onto those routers, and then does so.

It says: "When possible, in 36 cases, the exploit kit modifies the router's network rules to make the admin ports available from external addresses," so it's opening incoming connectivity to your router, they write, "exposing the router to additional attacks like those perpetrated by the Mirai botnets. And the malvertising chain is now accepting Android devices, as well."

So this is what we're up against. So innocent people using Chrome on Windows, or of course the default browser, Chrome on Android, just visit a reputable page, and their browser receives malvertising, which is all that's necessary to commandeer the Chrome browser. And this is using AES encryption with dynamically varying keys. So the content was encrypted under the key, which is steganographically hidden in the ad's image. The script that's part of the ad runs in Chrome, parses the least significant bits out of the image to extract the AES key, uses that to decrypt a blob which looks like random noise so it doesn't get caught by any advertising blocking anywhere.

That then decrypts the code necessary to cause the JavaScript running in the browser to be able to then perform a series of local queries to your router, checking its fingerprints to identify it, and then selecting specific router-specific code to access the router, change DNS, so now instead of your ISP's DNS or Open or Google or wherever, like real DNS, you then, without knowing it, your DNS gets changed so that all the devices in your network are now obtaining DNS services from a bad guy.

And as we know, DNS security is lagging behind. We've spent a lot of time on the podcast talking about how important it is that it be secure. It's, however, lagging behind. And that means that anything you do in your network for as long as this change persists will be looking up IP addresses from illegitimate DNS servers, any of which can then be used to point somewhere else. And of course what that means is your browser actually thinks you are at https://amazon.com, but you're at a site pretending to be that. And unfortunately, that then allows them to grab the cookies that your browser provides with its very first query to that site, thinking that it's actually at Amazon, but it's not Amazon, picking up your cookies and your authentication at that site. And it just goes from there.

So, wow. Again, complexity. We have built an incredibly complex system where, by leveraging individual features, each individually, which seem benign, when you put them together in a clever fashion, you can pretty much do whatever you want to. And unfortunately, these guys have watched millions of people being subjected to this, and some subset of them being vulnerable. So if nothing else, you may think, "I do not need

to change the login for my router because I'm sure I'm blocked from the outside. Only people on the inside can get it." Don't leave it set to admin and password as your username and password, or admin and admin or whatever it is. It's worth making sure that something that might operate in your machine is unable to get to your router. Yikes.

And in something not security related, the National Highway Traffic Safety Administration has published a Notice of Proposed Rulemaking. It is 392 pages, so I'm not going to go through it. The table of contents, though, was interesting to browse through. This proposed rulemaking will be mandating vehicle to vehicle, so it will have a new acronym, V2V, communications systems in all new cars and trucks. Once the rule is finalized, car makers will have two model years to begin including V2V systems, with a bit of slack to allow the synchronization of product cycles. V2V-equipped cars will communicate with each other at short ranges, I think I remember seeing up to 300 meters, to prevent the kinds of accidents where current advanced driver assistance systems, most of which depend upon line of sight, are not effective.

V2V and, well, for example, a car might be broadcasting "I'm stalled, I'm stalled, I'm stalled." And when you get within range of that, that would heighten the awareness or alertness maybe of a human driver or of the auto-driving software which is approaching the stalled vehicle. V2V and related vehicle infrastructure, there's another one, Vehicle to Infrastructure, that's V2I, and they're referring there to things like stoplights - apparently we're going to be talking to stoplights before long, too - relies on what's known as the Dedicated Short-range Radio Communication (DSRC) wireless protocol to communicate between devices at ranges - oh, yeah, here it is - of up to 984 feet, which is an odd-looking number because it's actually 300 meters. Vehicles will send out standardized basic safety messages that trigger driver alerts or even emergency avoidance actions to prevent crashes. And of course, under the topic of what would possibly go wrong, what we're doing is we are escalating the level of technology with the best of intentions. Merry Christmas.

So Ars, in their coverage, writes: "Recognizing the immense implications of an insecure protocol, the notice asks industry and the public for input on the proposed security specifications and proposes that 'vehicles contain firewalls'" - of course they don't know what that means - "'between V2V modules and other vehicle modules connected to the data bus to help isolate V2V modules being used as a potential conduit into other vehicle systems.'" Which of course happens all the time, unfortunately.

"Privacy is also given due attention" - how nice - "and the proposed rule would prevent cars from sending out identifiable data like a vehicle's VIN number or a driver's name or address." Well, again, how thoughtful that your car is not going to be announcing who you are as you're driving along. So, oh, goodness. This podcast will never end, Leo. We will have fodder from now, I mean, we keep doing this. We have IoT. Now we're going to have the IoT of vehicles, called V2V. Notice there's no "S" in V2V, either, just like there's no "S" for security in IoT. Wow.

And then in something that I thought you would find interesting, Leo, the news is photographers and filmmakers are calling for encryption to be built natively into cameras as a standard feature.

Leo: I don't get that. I don't...

Steve: Okay.

Leo: Yeah, explain it.

Steve: Here it comes. Over 150 filmmakers and photojournalists are calling on major camera manufacturers to build encryption into their cameras. Last Wednesday, Trevor Timm, the executive director of the Freedom of the Press Foundation, wrote: "Today, Freedom of the Press Foundation is publishing an open letter to the world's leading camera manufacturers - including Nikon, Sony, Canon, Olympus, and Fuji - urging them to build encryption into their still photo and video cameras to help protect the filmmakers and photojournalists who use them."

The letter has been signed - and by the way, Leo, you ought to pull up the letter. It's the second link here on the page. It's a DocumentCloud.org letter that shows the signatories to this. "The letter is signed by over 150 documentary filmmakers and photojournalists from around the world, including 15 Academy Award nominees and winners [including] Laura Poitras, Alex Gibney, Joshua Oppenheimer, and many more.

"Documentary filmmakers," they write, "and photojournalists work in some of the most dangerous parts of the world, often risking their lives to get footage of newsworthy events to the public. They face a variety of threats from border security guards, local police, intelligence agents, terrorists, and criminals when attempting to safely return their footage so it can be edited and published. These threats are particularly heightened any time a bad actor can seize or steal their camera."

Leo: Like Nicolas Cage? He's a terrible actor. Oh, no. Oh, I know what you mean. Bad guys. Well, no, bad actors might want to see some footage, too. I don't know.

Steve: Bad actors may want to lose the key. "They are left unprotected by the lack of security features that would shield their footage from prying eyes. The magnitude of this problem is hard to overstate: Filmmakers and photojournalists have their cameras and footage seized at a rate that is," they write, "literally too high to count." Although we do have high numbers. Anyway...

Leo: Sounds like it's infinity.

Steve: You know, crypto. Those are big numbers. We recite those routinely. "The Committee to Protect Journalists, a leading organization that documents many such incidents, told us: 'Confiscating the cameras of photojournalists is a blatant attempt to silence and intimidate them, yet such attacks are so common that we could not realistically track all of the incidents. The unfortunate truth is that photojournalists are regularly targeted and threatened as they seek to document and bear witness, but there is little they can do to protect their equipment and their photos.'

"Camera manufacturers are behind the times compared to other technology companies. All iPhones and many Android phones come with encryption built into their devices. Communications services like Apple's iMessage and FaceTime, plus Facebook's WhatsApp, encrypt texts messages and calls by default. And many major operating systems on PCs and Macs give users the ability to encrypt the hard drives on their computers. Yet footage stored on the professional cameras most commonly used today are still left dangerously vulnerable. Finding the right way to provide encryption in their

products will take some research and development from these camera manufacturers, and we welcome having a conversation with Nikon, Sony, Canon and others about how to best move forward on this important initiative. However, we are hopeful they will publicly respond with a commitment to building encryption into their products to protect many of their most vulnerable customers."

And of course we can see where this is going to go. The instant one of them does, they all must follow because this is obviously an important need and issue for that segment of their cameras' purchasers, and they're going to have to have that bullet point; you know? Native military-grade encryption, blah blah blah, which will soon become, I expect we will see, a standard feature in high-end professional cameras. I thought that was really interesting. Hadn't thought about that.

Leo: Yeah, I hadn't either, yeah. Makes sense, though, yeah.

Steve: Yeah, totally.

Leo: Yeah.

Steve: So there is a service that I would argue our listeners, if you need it, it's nice on its face. But unfortunately, to make it maximally available, they have sacrificed some security in some worrisome fashion. What this is, it's a web-based frontend to the Let's Encrypt service. It's called SSLforFree.com. Amazing that domain wasn't used before now, S-S-L-F-O-R-F-R-E-E dot com, SSLforFree. So under their "How It Works" they say: "We generate certificates using [the Let's Encrypt] ACME, A-C-M-E, server by using domain validation. Private keys are generated in your browser" - okay, there's problem number one - "and never transmitted."

Well, it's good they're never transmitted. Browsers are just not a place you want a private key for your web server to ever be resident. But they say: "For modern browsers we generate a private key in your browser using the Web Cryptography API, and the private key is never transmitted. The private key also gets deleted from your browser after the certificate is generated." Well, how nice.

But then they say: "If your browser does not support the Web Cryptography API, then the keys will be generated" - they're talking about the private key for your web server - "will be generated on our server using the latest version" - okay, good - "of OpenSSL and outputted over SSL" - good - "and never stored." Good. But you don't ever want your server's private key to ever go anywhere, really preferably never outside of your server. And that's the way it's being done now. Your private key never leaves the server that generated it. Now they're saying, oh, yeah, maybe your browser can generate the private key. If not, we'll happily do it for you and then send it to you [audio dropout] with the Let's Encrypt ACME server. Yikes.

Then they end, saying: "For the best security you are recommended to use a supported browser for your client. You can also provide your own CSR" - that's the Certificate Signing Request - "when using manual verification, in which case the private key is handled completely on your end." Okay, so, okay. So from all of this, if there's anyone out there who wants a Let's Encrypt key, or a Let's Encrypt cert, then you should definitely generate the private key the old-fashioned way.

If you've got a web server that supports HTTPS connections, it has the ability, I mean, it's probably got OpenSSL, or it's a Windows machine. So it's got the crypto API in it. They all can generate the private key for you. Do it on that server. And what that generates is a certificate signing request. Then you could use that safely because it does not have your private key in it. You are asking them to sign the matching public key. So that you put into SSL for free. They do the interaction with the Let's Encrypt server and give you back a certificate that is the signed public key for your server's private key that never left the boundary of your server.

So, I mean, it's nice that these guys are, like, bending over backward to solve the problem. But they're doing so in a frightening fashion. We know SSL. We wish it were secure. It's not. You're using web HTTP connections to move this stuff back and forth. We just talked about DNS changer, so you don't even know if you're actually connecting to them or to someone spoofing them. So again, there are too many holes in this Swiss cheese that we are currently operating in for that to be secure.

There's an interesting site I have in the show notes that I didn't want to skip. He calls it - it's RobinLinus.com. And the service is Webkay, W-E-B-K-A-Y. So if you go to, and it's a little unnerving, webkay.robinlinus.com, what you are presented with is a page of what your browser knows about you. Now, we've sort of seen these things before, but maybe not for a year, where we were looking at, like, all of the junk in the query headers and stuff.

This shows the geolocation of where you're located, what OS you're using, a bunch of stuff that's familiar. But it's just sort of - it's a convenient reminder of any site you go to, every place you go, just like this one, the amount of data that your browser and our state-of-the-art technologies, I mean, you'll notice there's battery in there. He says that my computer is currently charging its battery. It's like, okay. But the point is there's an API in the browser that tries to report on the state of your battery. And as we know, because of the resolution of some of those parameters, that can be used to track you. So just sort of a worthwhile reminder.

Also, Google Contributor is being terminated. Apparently it's going to get reborn.

Leo: Oh.

Steve: I know.

Leo: I was using that.

Steve: I am. In fact, what's so funny, Leo, is yesterday when I was researching this I was looking at a web page covered with Google Contributor blanked out ads.

Leo: Yeah.

Steve: So I thought, well, isn't that ironic. It was a big banner at the top and a bunch of little squares floating around with sort of those little pastel circle things that...

Leo: I see kitty cats.

Steve: Oh, you do?

Leo: Yeah. Well, that's - you can choose what you want to replace ads.

Steve: Oh, okay. Yeah, I went with, I don't know, fuzzy floodlights.

Leo: Probably smart, yeah.

Steve: So email has been sent. I didn't get mine yet. Maybe it probably went to my Gmail account, which is my slop account. Email being received reads: "Thank you for being part of Google Contributor, a service that helps readers enjoy fewer ads while funding the sites they love. Early next year we are launching a new and improved Contributor."

Leo: Oh, good. Okay.

Steve: Yeah. "Your input throughout the testing has been invaluable." Okay, I didn't give them any input, but I gave them a lot of money.

Leo: Yeah.

Steve: "As we build this new service, we will discontinue the current version of Contributor. What this means to you: Starting in mid-January 2017" - so three weeks from now, four weeks - "you will no longer see Contributor ad replacements as you browse the web." So they're shutting it down. "And you will be unable to access your Contributor account." They're closing that down. "You will no longer be billed" - well, thank you - "for the [nonfunctioning] Contributor service starting mid-January 2017, and we will refund your remaining account balance to your credit or debit card on file." That's all they say.

Now, they don't give a commitment to a start date. They're not telling us when they're going to replace it or when the new service will be coming up. But there is a link to a Google Docs form which you can fill out, and I did, and I've got the link in the show notes, to register yourself for notification when they relaunch the service. And, boy, I tell you, I mean, I see so many of those little Contributor boxes when I'm surfing the 'Net, which I'm happy to pay pennies for instead of either blocking the ads or having to put up with ads or the malvertising that they're replacing, that I'm happy to do that.

Also, I did want to mention the pfSense people have a beautiful little cute two-port pfSense hardware security gateway. It's the SG-1000.

Leo: They named it after you.

Steve: They did. Wasn't that thoughtful? I think it actually stands for Security Gateway, but I'm happy to go with Steve Gibson. It's just, I mean, it's like the size of an Arduino or a Raspberry Pi. It's just a cute little thing. It's got a WAN and a LAN port and power. Just adorable. And I forgot to say it's $150. So it is the low end of their range. Now, what it does not have is more ports. And that's the nice thing about the $49 switch, which also has a lot of security features, that allows for physical network segmentation. However, this is full pfSense. And that's, I mean, that's on top of NanoBSD, with the kitchen sink and five bathtubs in there. I mean, OpenVPN. You can run reverse proxies. You can, I mean, it's got everything.

So what you could do, because it also supports VLAN, is if you put a VLAN-aware switch downstream, that is, connect its VLAN port to a VLAN-aware switch - and those are cheap, those are 20, $30 - then the switch would enforce network segmentation with its multiple ports so that this cute little fire engine red security appliance does not need more than just its two ports. So I wanted to put it on everybody's radar, I mean, because it doesn't get any nicer than pfSense from a standpoint of a massively feature-rich solution.

And we have some errata. I misspoke last week because, in my defense, my brain cannot believe, it cannot process the reality of the truth of this. I said that IPv6 subscribers get 64K IPs. Wow, 64,000. Actually, we know it's 65,536 IPs. It's essentially a, what would that be, a 16-bit network block. Unh-unh. No. They get 64, not "K" IPs, 64 bits of IPs. Now, okay. Remember that 32 bits is the entire Internet currently. The entire IPv4 Internet is 32. And remember, 64 is double the bits; but it's not double the Internet, it's the Internet-squared. It's every single IP on the Internet is an Internet itself.

So I don't even know how to pronounce this number: 18,446,744,073,709,551,616. That's how many IPs we each get. You, Leo, get that many. I get that many. Actually, that's not even true. The recommendation is that, rather than a /64, maybe give them even more, shorten the so-called prefix.

Okay. So here's what's going on. The IP space in IPv6 is 128 bits. And they decided, we've got so many bits now, we're just going to slice it in half.

Leo: Wow.

Steve: There's going to be 64 bits on the left, and 64 bits on the right.

Leo: It's 18 quintillion, 446 quadrillion, 744 trillion, 73 billion, 709 million, 551 thousand, 616.

Steve: Light bulbs. That's how many light bulbs you can have.

Leo: I'm ready. My Internet of Things is going crazy. 18.5 quintillion.

Steve: Ooh, light bulbs.

Leo: Nice.

Steve: Okay. So I'm thinking, okay, this cannot be right. This is nuts. So I go to RFC 3177. Introduction. It's very staid, of course, when you're writing your RFC.

"There have been," they write, "many discussions between the IETF and RIR experts on the topic of IPv6 address allocation policy." Yeah, because, you know, we've got more bits than we could ever - so we've got them to burn. "This memo addresses the issue of the boundary in between the public and private topology of the Internet, that is, how much address space should an ISP allocate to homes, small and large enterprises, mobile networks, and transient customers."

And then they say the background, and we're running out of time, so I'll make this quick: "The technical principles that apply to address allocation seek to balance healthy conservation practices" - okay, I don't see any sign of that - "and wisdom" - or that - "with a certain ease of access. On one hand, when managing a potentially limited resource, one must conserve wisely to prevent exhaustion within an expected lifetime." Yeah, of the universe.

"On the other hand, the IPv6 address space is in no sense as limited a resource as the IPv4 address space, and unwarranted conservatism acts as a" - we don't have to worry about that being applied here - "acts as a disincentive in a marketplace already dampened by other factors. So from a market development perspective, we would like to see it be very easy for a user or an ISP to obtain as many IPv6 addresses as they really need without a prospect of immediate renumbering or of scaling inefficiencies."

Leo: No one could ever need more than 18.5 quintillion IP addresses.

Steve: Oh, Leo. "The IPv6 address, as currently defined, consists of" - get this - "64 bits of 'network number.'" That is, the high left-hand 64 bits identify your network. That is, this is you. That's the public IP, essentially, on the Internet. We used to have 32. Now we have 64. So there's also 18.5 quintillion individual networks. And, they write, "64 bits of 'host number'" - that is, light bulbs. "The technical reasons for this are several." And then they go into it.

So they're just going to split the 128 bits in half. Everybody gets their own Internet-squared in their house. And in fact it's even worse. I won't go into it any further, but that absolutely is the case. They say, they're suggesting in the general case maybe even give them - someone might need multiple Internet-squared 18.5 quintillion networks. So why not? We don't want to be bothered with anybody running out of anything ever again.

And we've run out of time. Not IP addresses, but time. I did want to acknowledge, as I did earlier, that steganometry apparently is a word I've invented for the measurement of steganography, which is the actual word. I did misspeak and talk about - I was talking about a high-number port, and I said 123123. Sharp observers noted that, well, Steve, 123123 is a little bigger than 65535, so you would need an extra bit of port numbering for that. And so of course, yes, I meant 12312. Doesn't sound quite as nice. But anyway, so thank you. And somebody who liked our 1973 map of the Internet had a 1977. I've now given these links permanent residence on the Link Farm.

Leo: Oh, good.

Steve: So you can find them. And it's in the show notes. So this shows, four years later, how the ARPANET has grown. And it's interesting. It's maybe twice the size, though we're not going exponential yet, but it's a nice - oh, and there is a key on this map verifying that the squiggly line is a satellite link out to Hawaii. And there's also something called NORSAR or SAC or something, I don't know [Norwegian Seismic Array]. Anyway, so there is that map. And I had a little SpinRite note about cabling errors, but we'll do that in two weeks.

Leo: And you also have to tell us...

Steve: Oh, about the robot.

Leo: About the robot CAPTCHA.

Steve: We'll start with that two weeks from now. For those who haven't already heard it, next week we're going to play the infamous, or famous, "Portable Dog Killer" episode. I guarantee you, I've never heard Leo laugh so long and hard as I was...

Leo: There's no video of this, though; right? It's just audio.

Steve: Well, we did a video.

Leo: Is there a video? Oh, good. Okay, all right, good.

Steve: Yeah. It is a video. So I think everybody will get a kick out of it. You may have forgotten it. You may...

Leo: It's a great episode.

Steve: If you've got some family around for the holidays. And it had a moral, also, an unintended consequences moral. So I think everybody will enjoy it. And we'll see you next year.

Leo: Tonight is the darkest night since 1648 or something like that.

Steve: What? Why?

Leo: Well, tonight is winter solstice. You know, the longest night of the year tonight.

Steve: Right, the shortest day, right.

Leo: Yeah, shortest day, longest night. And there is a total lunar eclipse. And there hasn't been a total lunar eclipse on winter solstice for almost 500 years.

Steve: Cool.

Leo: So the moon will go out. You can look at the stars.

Steve: Completely blocked by the Earth.

Leo: Completely blocked by the Earth. It will be very cool. It's from 2:41 to 3:53 a.m. Eastern time.

Steve: [Crosstalk].

Leo: Well, no, but, no, at midnight our time, 11:41 p.m. our time.

Steve: Oh. Oh, yeah, yeah, okay.

Leo: So that means midnight should be very interesting in your neck of the woods. And Happy Winter Solstice to you, Steve. And have a wonderful holiday, and we'll see you in the New Year.

Steve: Indeed, my friend. And I'll shoot you an email in a couple days with some update stuff.

Leo: We'll converse.

Steve: Yes.

Leo: Via the Internet. Steve Gibson does this show every Tuesday, 1:30 Pacific, 4:30 Eastern, 21:30 UTC. So do tune in, if you want to watch it live and join us in the chatroom at irc.twit.tv. Or join us in-studio, as Nina and Alexandra did. You just email tickets@twit.tv. We'd love to have you in here. But if you can't do any of that, you can get on-demand versions at GRC.com, Steve's site. He has audio plus transcripts, nice transcripts of every episode, makes it easy to search for something

you're looking for, at GRC.com. He also has other stuff there. While you're there, get SpinRite, the world's best hard drive maintenance and recovery utility, even for SSDs. You can also find Perfect Paper Passwords and SpinRite and all the other great stuff Steve does.

Now, if you want to get video, you need to go to our site, TWiT.tv/sn. We have audio and video. And of course you can always subscribe, and that way you won't miss an episode. You don't want to miss an episode. This is a good show. You learn a lot on this show. It's a lot of fun. Somebody said it's not a full moon tonight, it's a full Earth. I like it. Thank you so much, Steve. Thank you, everybody. Happy New Year. Merry Christmas. Next week, "The Portable Dog Killer." And we'll see you January 3rd, right here, for a brand new Security Now!.

Steve: Cool. Thanks, Leo.