SECURITY NOW!

Transcript of Episode #482

Listener Feedback #201

Description: Leo and I discuss the week's major security events and discuss questions and comments from listeners of previous episodes. We tie up loose ends, explore a wide range of topics that are too small to fill their own episode, clarify any confusion from previous installments, and present real world 'application notes' for any of the security technologies and issues we have previously discussed.

High quality (64 kbps) mp3 audio file URL: http://media.GRC.com/sn/SN-482.mp3
Quarter size (16 kbps) mp3 audio file URL: http://media.GRC.com/sn/sn-482-lq.mp3

SHOW TEASE: It's time for Security Now!. Steve Gibson is here. A Microsoft Update to last week's Microsoft Update in the news. We'll also talk about dirtboxes. They're flying over you all the time. And Steve will answer some questions, as well, including a stone DVD. Security Now! is next.

Leo Laporte: This is Security Now! with Steve Gibson, Episode 482, recorded November 18th, 2014: Your questions, Steve's answers, #201.

It's time for Security Now!, the show that covers your security and privacy online. And there could be no better host for this show than Mr. G, Steven Gibson. He is the man in charge at the Gibson Research Corporation, GRC.com, and also creator of many useful tools, including SpinRite, the world's finest hard drive maintenance and recovery utility. Hello, Steve.

Steve Gibson: Yo, Leo. Great to be with you. This is a Q&A, after last week's coverage of certificate transparency. And in fact we may be doing another show on certificate stuff next week as a consequence of some just-breaking news this morning, an announcement from the EFF, Mozilla, and the University of Michigan about the EFF's initiative to make certificates, web browsing certificates, web server certificates, the kind we're always talking about, both free and safe to issue and somehow kind of auto-renewing. So there's a new protocol called ACME we'll be talking about. And then also, boy, Microsoft is having a rough go of it for their patches in November. One dropped today that they'd held back...

Leo: Uh-oh, an out-of-band update? Wow.

Steve: Yeah. They revised one that was - the most important one last week turned out to have problems with it. There's an update to Firefox with a cool new feature that I like. I wanted to chat with you about the so-called "flying dirtboxes," which...

Leo: I'm an expert in flying dirtboxes.

Steve: Yes, I heard you on TWiT. I think that's great.

Leo: That one's peeved me off a little bit, so I'm...

Steve: Yeah, well, rightly so, and many other people. You could imagine the ACLU is having, you know, is on oxygen.

Leo: Yeah.

Steve: Also an update on the cellular provider supercookies, some information, some news about WhatsApp, BitTorrent Sync, and this is a Q&A. So since we had so much stuff to talk - oh, also some interesting miscellanea stuff. I wanted to chat with you a little bit about "Interstellar."

Leo: Oh, you saw it, eh? Ah.

Steve: Yeah. Also an upcoming movie or production about Alan Turing's life...

Leo: Yes.

Steve: ...that looks really good.

Leo: I talked to somebody who saw who says its amazing.

Steve: Oh, I'm so glad. And then we have a Q&A. But so much going on there that I just found five questions, so a half-size, a half-pint Q&A, which I think will pretty much round out a good podcast.

Leo: Excellent.

Steve: So lots and lots of stuff today.

Leo: Holy camoly. Oh, my. I was hearing radio sound in all of this, and I realized what it is. It's my Pono Player has decided all of a sudden to start playing "Psycho killer qu'est-ce que c'est?"

Steve: Hey, so we briefly mentioned it last week. What do you think? You've had it for a week now.

Leo: You know, I'm going to review it on Before You Buy. That's why I have it in the studio.

Steve: Oh, okay, cool. I'll keep watching out for it.

Leo: Well, I'll give you the thumbnail.

Steve: Cool.

Leo: I want to support it because I really want to support the notion that you can buy high-res, studio-quality recordings of your favorite music. And in order for that to work we have to develop an ecosystem of people who are willing to buy it. And it's always going to be a specialty. Although I have to say the PonoMusic Store, HDtracks, Bowers & Wilkins has a Society of Sound. They all sell albums for, you know, 17, 18 bucks.

Steve: And so these are the same albums we're familiar with. They've gone back to...

Leo: They're effectively studio masters.

Steve: Right, so they've gone back to them and said, okay, wait a minute, we want the raw original data before you ran it through your compression. And that's what they offer.

Leo: Yeah. And most - sometimes you'll get stuff with compression on it, both digital and audio. But what you're getting with this stuff is FLAC files, so they're losslessly compressed. And then the bitrate is, you know, it varies. I mean, sometimes it's just, on some of these systems, it's just CD quality, 44.1 by 16 bits. But most of them are at least 24 bits. And then some of them are 96K. Some of the are 192K.

Steve: Ooh.

Leo: Yeah.

Steve: In big files, big files.

Leo: They're big files. I doubt you'd hear the difference. But nowadays with big hard drives, you know, who cares?

Steve: Yeah, it's a very good point. The whole MP3 thing was designed back in a really lean storage and bandwidth environment.

Leo: And bandwidth, right. So, and I think we'll look back on this MP3 era as kind of just an unfortunate blip. I'm hoping. Anyway, but in order for this all to work, besides being able to buy the tracks, you need something that will play back the tracks. For instance, Apple and iTunes will not play back high-res tracks. They'll play back a certain degree, but not all the way. But you can get DACs, Digital-to-Analog Converters, that will do 192/24. And you can get very high-quality digital-to-analog converts, DACs. And so my Onkyo AV receiver, for instance, has an excellent TI DAC that does 192/24. So I can listen on my stereo. I can actually put it on a USB key, put these high-res tracks, pop it in the stereo, and listen on good headphones or on my good speakers. And you know what, maybe it's psychology, I don't know. I think it sounds a lot better. The soundstage is larger. There's more detail. There's more open...

Steve: I would think it would be brighter, yeah, cleaner.

Leo: Yeah, the high ends are brighter. But you know what, even the bass is more precise. It's not as [making sound]. You feel like it's more detailed. And then so but the question is, you know, an iPod won't play it back. So Neil Young created this Pono, P-O-N-O, Player, PonoMusic.com, to basically be a high-res iPod. But as you can see, I mean, it's kind of bigger than an iPod. It's a triangle. In fact, you could put it on your desk, and you could have your name on it, and it would be just about the size...

Steve: Your name tag.

Leo: Yeah, your desk name tag. This one is - I got it on Kickstarter because it was a Kickstarter project. So I have the Neil Young Limited Edition. This is 251 out of 500. But that was just because I was an early adopter. You still can - you can buy them now, they're in their second manufacturing phase and won't be available for a month or two. Four hundred bucks. But they've put, they say, and I think they're right, a very good DAC in here, plus a good headphone amp because you...

Steve: I'm sure they did because when they first announced it I immediately went to the specs for the DAC that they were using because I had not looked at ultra high resolution DACs for a long time. And there's a lot there. I mean, there's a black art to merging the digital world and the analog world at that level of resolution because, when you're talking 24 bits, I mean, those bits are really small down at the least significant end. And there's a lot of digital noise flying around. So, I mean, it's an amazing piece of just technology to be able to convert 24 bits of digital data to analog with that kind of precision. It's really interesting from an engineering standpoint.

Leo: Well, and you know a lot about this because I know you worked in audio as a student. And this is one of the things that you are kind of pretty up on.

Steve: Yeah.

Leo: The problem is, and a lot of people say this, oh, it's all psychological. You know, it's the "golden ears" thing. And you think you're getting better quality, so it's going to be better, sound better to you and all of that. And I can't vouch for that one way or the other, you know.

Steve: There's always been an audiophile component to people who enjoy music, people who really want good speakers and good amplifiers, I mean, just who enjoy that. And so I think it's cool that there's an audiophile offering within otherwise this white-ear-buds-on-the-bus crowd.

Leo: Well, but that's the other thing. Because now, okay, so now you're got this. Now you've got to either get - this has both headphone and line out. So now you either have to get really, really good headphones - 400 bucks for this, that's nothing compared to the headphones - or you have to hook it up to your stereo via analog in. And if your stereo doesn't have a good DAC, well, this could be in a way giving you a good DAC. But you want great speakers or great headphones to even approximate it. But assuming you have that, and you have the ears, I might be - we might be too old to really - I think I hear a difference. I think I really hear a difference, but...

Steve: I would just call it the last half a percent. And if you're someone who wants to push to get the last half a percent, then, hey, great.

Leo: Right. It was interesting...

Steve: So it's not going to be a huge market, but it'll be a market.

Leo: Scott Wilkinson, our home theater geek, on his AVS Forum, you know he's the editor over there, did a blind A/B comparison. He posted four files, two of them CD quality, two of them high-res, all FLAC. And he invited his readers to download them and tell him which were which. And he found an interesting result, which actually to me confirms that you can tell the difference if you're paying attention. And he also asked, what equipment are you listening? The readers that didn't have equipment that could handle higher resolution music, 50/50.

Steve: Ah, wow.

Leo: It was almost exactly 50/50. The statistical odds for guessing.

Steve: Expectation, yes, right.

Leo: Yeah, for guessing. Of the listeners who had equipment that could play back higher res files, 80 percent accurately picked the high-res files. That to me said that's more than statistical. That to me says that you can tell the difference. Maybe somebody 57 years old can't tell the difference, but somebody can tell the difference. And it makes me happy just to buy the music in as high a quality as I can. And it's not that much more expensive.

Steve: And you've got storage space now.

Leo: And I can store it. The whole collection, which is about 20 albums, because I'm not buying everything, just my favorite things, 60GB. That's not a lot. Now, that was a lot when a hard drive was 2GB.

Steve: Oh, my god. Yeah. Well, I remember, you know, the MFM drives were 20MB. So, yeah. We were doing everything we could so squeeze music down at that level.

Leo: But 60GB now, you know what, I put it on my Dropbox, I put it on my Transporter because I want - you don't want to lose these. It's you can't, you know, that's it, they're gone. And I put it on my Pono. This has 128GB storage capacity. So, easy. And classical music sounds beautiful. But really my favorite albums are mostly rock. And, boy, they sound great. When they're not digitally recorded, a lot of albums - when did they start using PCM recording? In the '80s? '90s? So those you're going to get the original, you know, quality is going to be the same as the original PCM recordings, whatever they recorded at. If it's tapes...

Steve: Oh.

Leo: So, for instance, I have Bob Dylan, "Blood on the Tracks," one of my favorite albums, recorded on tape. But the tape is analog. So they master them still at a high bitrate, hoping to capture all the analog data. So it still, it seems to me, would be better, coming off the analog master in a high-quality system to a high-bitrate recording, is going to be better than what you've got on a CD. Right?

Steve: Yeah. Yeah, although, I mean, analog systems are notoriously troublesome, too. You've got, I mean, remember wow and flutter, which are, like, real things. And then hiss, you do have a signal-to-noise ratio on tape that people like Dolby made their fortunes trying to overcome. So what's really the best would be high-end, high-bandwidth, digital sampling that's then stored digitally and then returned to you digitally. And then let's hope that you can transduce that back into analog that your ears are able to accept. We don't yet have our digital interface.

Leo: Or if they did it on tape, but the tape was going around really, really, really, really, really fast.

Steve: Well, and in fact that was one of the things they did was they ran the tape at a high speed in order to increase the signal-to-noise ratio.

Leo: I have a recording which was clearly analog of "The Rite of Spring." It's Leonard Bernstein and I guess the New York Philharmonic. And the dynamic range on Stravinsky's "Rite of Spring" is huge. It starts with one oboe or some woodwind. And you can hear the air in the room. You can hear the musicians rustling. You can hear...

Steve: Like the flutter in the...

Leo: You're in the room. And as far as hiss goes, you hear no hiss on any of these recordings. There's maybe a little hiss on the Stravinsky. But not much. But mostly what you're hearing is the susurration of human respiration.

Steve: Yeah, like being there.

Leo: Anyway, this is a security podcast, but I know you're really interested in this stuff, so I think it's worth - and yes, the Pono does a great job. But really it's such a big topic. It's not just the Pono, it's about getting good speakers, good headphones and all the rest.

Steve: Right, right. Okay. So to our topic, Microsoft having a really rough November. We spoke last week, which was Patch Tuesday, the Second Tuesday of November, about them dropping basically a mega, and I called it a "red-alert" set of updates because what was the third one in order, the MS14-066. That one, I mean, it's funny because my impression I've since been reading echoed elsewhere was that they just sort of slipped it out there and didn't say much. And when I'm just scanning the details of it for the podcast, it's like, what? And as I said, I had to make a trip over to my physical servers at Level 3 in order to make sure they got through the process of updating this because this one was not one I could allow stand.

I should mention that I don't patch my servers every month because I have a very specifically designed constrained environment. And a lot of stuff that affects other people, like SQL Server, I don't have any SQL Server. I don't have any .NET stuff exposed or Active Server Pages or any of that. So I'm able to, and I do, rather than just doing a wholesale update, I actually look at each individual aspect of the patches, and I go, eh, I don't really - and in fact I had not, there wasn't one since May of this year. When I looked, the last time I had installed patches, Windows Update told me and my logs told me, was early in May of 2014. That was when there was one that I thought, okay, this one I can't ignore. This one needs to get put in. And so at that point I caught the system up, I caught my machines up, brought them current. But they've been idling since. I never reboot them. They never crash. They just run. But this one, the 066 was a biggie.

However, a couple days after Tuesday came the news that they had botched something about those four new cipher suites that they added. Our listeners will remember that last week it's not only are they fixing some sort of a horrible vulnerability that we didn't know that much about, except they just said this is a horrible vulnerability, there's no workarounds, there's no mitigations. Fix this. And both for servers and for workstations. And remember that they also added, which is unusual for Microsoft, to add functionality during a patch. Normally they do that later in the month. But for whatever reason, this included four new TLS cipher suites. And they were good ones. They were Diffie-Hellman, which we like. They were SHA-256 and 384, so good, strong SHA-2 family hashes. I mean, you want those.

Turns out they were broken. And so people who did follow Microsoft's advice and implorings about getting their systems updated then began having random lockups and server crashes and huge problems because there were problems with the cipher suites. So the advice then came out a couple days later, uh, delete these from your registry and reboot your machine. The good news was, since I had already overridden the choice of cipher suites to deliberately select the ones that I wanted, that overrode any automatic inclusion of the new bad ones, so I didn't have any trouble.

But today, in addition to dropping another update which they had held back last week, so we have another, another completely different. And everyone is running around again saying this is really important, update today. That is, they did an out-of-band, essentially, although technically they just held it back because they apparently weren't ready for it, or it wasn't tested enough or whatever last week. So in addition to that, and this is important, because this is not automatic, so everybody who's got IIS servers, I'm talking to you. You need to manually now redownload the MS14-066 patch and reapply it. They have reissued it as Version 2, and so I will be doing that at the end, after this podcast is through, since all of this just came in this morning as I was prepping things, and I haven't had a chance to. Besides, I'm always twitchy about rebooting the server, and so I wouldn't want to do it here before the podcast.

So, but anyway, anyone who applied 066 last week, and everyone should have, it's not so crucial for workstations. But you might want to do it anyway. You need to go to the page MS14-066 and get it again manually, download it, and then reapply it. And so everybody who's using IIS in a web server role exposed to the Internet should do that. This fixes whatever it was they broke in those four cipher suites.

Leo: I imagine they'll push it at some point, but they're not pushing it now.

Steve: They're not pushing it now. That's a good question, Leo. I wonder, maybe they're just a little gun shy after having messed it up last week.

Leo: Yeah, yeah.

Steve: So, yeah, but you're right, they really ought to repush it, and maybe they will.

Leo: Oh, I'm sure they will. But, yeah.

Steve: Next month.

Leo: Yeah.

Steve: And the bad guys and the good guys, okay, a range of hat colors from white through gray into black are in fact pounding on what it was that Microsoft changed. I have seen disassemblies posted online of the patch. The people of various hat colors have found the change that Microsoft made, and they are managing to get Windows servers to crash by fuzzing the TLS handshake. So this is always the first stage of developing an exploit.

So, and what is a concern is, and I didn't mention this last week, and I should have, this is clearly wormable. So we haven't seen something like this since Code Red and Nimda and Blaster worms, which were huge events on the Internet. This is of that scale because an unwitting server that supports secure connections could, once this evolves from crashing servers to running explicit code on servers, and that'll be next week's news, then it becomes something where the hackers are going to, you know, first they'll probably play around with trying to get into servers, depending on how much they're able to do through this exploit. But then they're going to want to just do a worm. And so that servers that are exploited start searching for other servers. And that was immediately where you went to last week, Leo, and you were correct about that, about the idea of, you know, of compromised servers then compromising others.

Leo: Yeah, we know that's how they work because - yeah.

Steve: Or their users, yeah.

Leo: Yeah, compromising a server doesn't get you as much as compromising everybody.

Steve: Yeah.

Leo: Is it almost always the case that, if you can crash, you can do a buffer overflow and crash an OS or crash a system, that you're going to likely be able to figure out a way to then go to the next step and compromise them?

Steve: I don't know that we could say definitively one way or the other. It's certainly possible for there to be crash-only problems. Microsoft calls those "denial of service" vulnerabilities.

Leo: Right, right.

Steve: And we hear about those all the time. You know, half of the Microsoft patches that we're receiving are not remote code execution, they're so-called denial of service. And we're used to thinking in terms of the bandwidth flood-style DOS attack. But when Microsoft uses the term, what they mean is you crash the service, thus denying the service that you crashed. So a lot of these are just, well, we could make the code path crash, but we weren't, by the nature of it, we weren't able to give it a payload. And so we just don't know. We won't know, you know, maybe this won't go any further. But we know that people are looking at it really hard. I mean, they're tearing this code apart, figuring out exactly what the code path is. And then they're certainly going to see whether, given what they learn, can they arrange to get a buffer executed. That's, of course, that's the keys to the kingdom. So that's where we...

Leo: Amazing. Just amazing.

Steve: Yeah. Really cool stuff.

Leo: Yup.

Steve: Okay, now, I got news a couple days ago from Firefox that it wanted to restart. And so I always check to see what's going on. I got updated to 33.1. And then this morning for the podcast I went to About Help, or, oh, no, Help About. And it said, oh, look, I've got something more. So they found and they did a little patch on 33.1, and now we're at 33.1.1. Here's the cool thing they added that I really think is neat. They call it "Forget." And it's a new means of enhancing privacy. The traditional approaches are the so-called "incognito" browsing, where you open a window or a tab or an instance preemptively, with expectation that you are going to be doing something you don't want the system to record for whatever reason. That's been the traditional approach. Firefox v33.1 adds a Forget button that allows you - and I call it their Regret button.

Leo: Or the Day After button.

Steve: Exactly, that allows you to cause Firefox to expunge from its memory the previous five minutes, two hours, or day, or 24 hours. And so when 33.1 comes back up, it gives you a little song and dance to talk about its new features. And one of the things you can say is, oh, yes, I would like to have the Forget button added to my toolbar. You can still always access it through the menus, of course. But it just - and it's a little sort of a spinny backwards arrow. And I just thought that was really neat.

Leo: That's such a brilliant idea. I don't know why nobody's thought of that. I mean, every browser has an incognito mode now. But the idea that you may not know you want to be incognito till after you visit that site; right?

Steve: Exactly.

Leo: Whoops.

Steve: Which is why I think this is so clever.

Leo: It's very good.

Steve: The reason is, from an implementation standpoint, it is much trickier. Essentially you're having to do some sort of journaling or logging. You have, like, time passing. And so you need to go back and rewind your state to an earlier time. That's, from an implementation standpoint, that's a lot trickier than setting up an environment that you know you're going to flush so that none of its updates are recorded permanently. So this is a little, I mean, from an implementation standpoint, I appreciate that they're able to do this. It's like, oh, that's very cool, but also a very handy feature. So at this point Firefox now has it.

Leo: Have we ever talked about - and this would be a great subject for a later date - just what is forgotten and what is not forgotten in an incognito mode?

Steve: Yeah, good point.

Leo: Because obviously forgetting is only going to - is going to have limitations. For instance, when you enter an incognito mode, I presume - well, maybe not, actually. Your IP address would have to be sent to the site; wouldn't it? Or you wouldn't have a conversation.

Steve: Yeah, so, you're right. So the idea is it would be like cookies that got set or caching of things that your browser received during that time. The things that are normally sticky don't stick. They're kept in RAM. They're never written to disk. And they are, when implemented correctly, proactively overwritten before the memory that they were occupying is released back to the operating system. But incognito is not necessarily anonymous. So the things the browser is sending out in its query headers are probably still identifying. But you're right, it would be worth digging in and seeing whether browsers sanitize their queries in some fashion because that would be another form of anonymizing which is really different than forgetting on your local system.

Leo: Yeah. And frankly, if that's the case, it's not something you could do retroactively. You can't retroactively sanitize a conversation.

Steve: Correct, yes.

Leo: And so somebody's pointing out this is really not so very different from clear browser history or clear cookies. It's just time, you could specify the amount of time that you're clearing.

Steve: Right, right. Yes. And I'm sure we've all been in the position of where we have sometimes, because we needed to, or sometimes inadvertently, we've overcleared our history. Like if you clear your third-party cookies, baby, well, okay, all of the semi-static things that know you, no longer know you. So you're logging in everywhere again because you've said don't, you know, I want to just scrub all of my sessions.

Leo: That's a pain when you do that.

Steve: It is. It is. Exactly. And that's the point is that, if you knew that you only needed to go back, you know, that five minutes would repair whatever you had mistakenly done, then it's like, oh, good, I'll just take five minutes, please, because I'd like to keep everything else I've been doing all morning.

Also they built in, sort of making it very easy to use, the DuckDuckGo anonymous searching website as one of their available offered searches. So that's the second thing that was added is it just - it's built in, making it very easy for someone to choose non-tracking searching as an option on the Internet. So I thought that was nice. Speaking of what's not nice, flying dirtboxes.

Leo: Mm-hmm. Even the name should tell you something. You have a nice little graph or a graphic here.

Steve: Yeah. The front page, I always try to put something relevant to the podcast topics on the first page, the bottom half of the first page of the show notes. And this is from - this is the graphic, I don't know, maybe it was The Wall Street Journal. They broke the story, but they're behind a paywall that I couldn't get past. So everyone was looking at CNN's coverage of this from The Wall Street Journal behind the paywall. But it got picked up and widely covered because it upset people. The story, okay, we've talked in the last few months about the so-called "fake cell towers." And we really need, like, cell - we need a better term than "cell tower" because there's nothing that's a tower about it, as I explained before. The idea is...

Leo: It looks more like a palm tree in some cases.

Steve: Yeah. Well, and what it really is, is a briefcase, so...

Leo: In this case. Well, they've had this, what was it, the Stinger?

Steve: Yes. And that was my point, is we talked about those, for example, maybe in Las Vegas, in casinos, or law enforcement. We know of municipalities all over the country that have these devices which are, essentially, they are fake cell systems.

Leo: Sites. Sites.

Steve: Yeah, sites. Unfortunately, "cell tower" is really the only term we have. But I remember, when you and I were talking about this before, the question was, you know, could you, if you traveled, could one of these handoff to a real cell tower? And it was no, because the way real cell towers hand off among each other is they're able to just switch the conversation from their feed to the other tower's feed and make a seamless transition. In fact, that was really, when we used the term "cellular," that's what cellular means, is this really cool concept of a grid, and it's actually a hexagon ideally, of overlapping cells where each cell only has a short range and only needs to transact conversations within its radius.

And then as someone driving, in the classic example, drives out of that cell's coverage range, they're already in an adjoining cell's coverage. And by looking at the relative signal strength, the outgoing cell can see that it's beginning to lose this guy. And so it's able to query the adjacent cells and say, hey, who sees this guy? And one of the cell towers says, hey, I do. And so the other cell tower says, okay, you take over. And so that switch occurs without the user, without anyone in the conversation ever knowing. That's cellular communications. And so you're able to drive from San Francisco to San Diego, potentially, I mean, it doesn't really work, but...

Leo: You know, I had no idea that that's how it worked. I thought somehow the phone was involved, like, oh, I see a better tower. So it's actually the towers that are communicating with one another and doing the handoff themselves. How interesting.

Steve: And negotiating, yeah, and negotiating that traffic. And so, and they get all kinds of interesting information. For example, I don't know if we talked about it on this show. There was some guy who was driving somewhere in the Midwest on his commute with a very powerful cell jammer, thinking he was doing some civic good by preventing other motorists in his environment from having any conversations on their cell phones because you're not supposed to talk...

Leo: Oh, good.

Steve: I know. And he was...

Leo: That's horrible.

Steve: It was horrible. And it went on for, like, several years. And he was, like, blocking all emergency services and other stuff because it was just some horribly overpowered, just blanket of like the cone of silence driving down the freeway. Well, the cell carriers all noticed this pattern that repeated at commute time, the same location, the same place every time each day during the week, not on the weekends. And they caught him because they were able to associate this moving zone of destruction through the cell system and finally figured out, okay, that guy passes by the same - oh, there it is.

Leo: For two years. This is the definition of dick behavior, ladies and gentlemen. Holy cow. He's going to face a $48,000 fine, by the way.

Steve: Yeah, a ton of fines because it is absolutely illegal to do this. And so the point is that there's monitoring of all this going on, and the cellular system gets feedback. And so what they saw was this weird dead zone traveling down the geographical territory that was the freeway, and finally put two and two together.

Leo: Here's the story from The Verge. I'm loving this. By the way, it wasn't Verizon, AT&T, or Spring that caught him. It was Metro PCS. Apparently Verizon doesn't care. Ah, we see cells drop all the time.

Steve: We've got your money.

Leo: Reception was flat-lining along the same point of I-4 in Florida twice each day. The FCC used, quote, "sophisticated interference detection techniques." I've seen the trucks. There are only - there's five or six of them. There are not a huge number. But they have these great trucks that can go out.

Steve: Really cool Yagi antenna.

Leo: Yeah, yeah, and track the stuff down. And officers, they found this guy in his Toyota Highlander. When officers finally pulled him over, it didn't take long to confirm their suspicions. As they approached his car, officers immediately noticed their radios lost all contact with dispatch. Jammers, the FCC says, are illegal under any circumstances and can result in jail time.

Steve: Yup.

Leo: Wow.

Steve: Yup. So that was - we were talking about fixed or law enforcement-based sort of suitcase things, and how they use these when people don't know they're there. Basically these things are pretending to be cell towers. They get your phone to connect to them for whatever law enforcement purposes they are alleging. What The Wall Street Journal discovered and has been picked up is that there are also small aircraft flying overhead with the same technology. In the articles covering this, it's like, "Fake airborne 'cell towers,'" in quotes, "dragnet and inspect all phones below." So you can imagine the American Civil Liberties Union is unhappy.

Leo: Oh, yeah, Chris Soghoian, who's their CTO, Sal Soghoian's brother, said this is appalling. He says, "I can't imagine even that if a judge approved this, he even understood the incredibly widespread nature." I mean, you're gathering as many as hundreds of thousands of phones in this dragnet.

Steve: Yes. Yes, you are causing every cell phone within its range on the ground to preferentially connect to this fake flying cell tower. And law enforcement says that they're doing it to catch bad guys.

Leo: How is this different from the guy in Florida? Because aren't they breaking the cell phone when they do this?

Steve: That's a very good question, whether "can you hear me now, can you hear me now..."

Leo: No. It's a jammer. I'm sorry, I'm talking to a dirtbox.

Steve: There's a government dirtbox flying overhead, yeah.

Leo: Nineteen airports in the United States. These are Cessnas. They're small planes. And they say it covers - 90% of the U.S. population is covered by these flights.

Steve: Wow.

Leo: They say they gather - what would they get? They would get the unique identifier for each phone.

Steve: Yeah, generically we would say they get the metadata, which is they say they're not messing with the conversation. We know that the boxes we were discussing before are, I mean, they're pretending to be cell towers. The decryption...

Leo: There's no phone calls going through them, though; right? I mean, it's not - as we've said before with these Stingers, you're not going to continue to have operation.

Steve: No, but you are, but they are monitoring the conversations of the cell phones that they're intercepting.

Leo: You will get five bars briefly. Look over your head. If you're getting five bars, and you see a Cessna, you're being gathered.

Steve: Yeah. I have a great connection to the spy tool.

Leo: But as you pointed out, they couldn't do a handoff because they're not really in communication with the cell networks. So your call's going to be interrupted at some point.

Steve: Yeah. The problem is very little is known about this. The government clams up. And the only problem I have is the secrecy. If this weren't secret, then it's like, okay, justify your existence.

Leo: No, no. It's not - it's a fishing expedition. They say, just as the NSA does, no, no, if you're not a suspect, we, quote, "let go" of the information. But you've got to imagine they're harvesting it and storing it in that giant facility in Utah, just as the NSA does. They collect everything. They say, look, we're not going to look at it if you're not a suspect. But if you ever became a suspect, you could be pretty sure...

Steve: Right, retroactively.

Leo: ...they would say, oh, we know where he was.

Steve: Yeah. We would like to know where these people were. CNN contacted the Department of Justice for comment. An official at the DoJ would not confirm or deny the use of flying spoof cell towers. He said, "Any discussion would let criminals and foreign governments, quote, 'determine our capabilities and limitations.'" Which, you know, is just like, well, you can't make us talk, so we're not going to.

Leo: And of course because this is metadata, you don't need a warrant.

Steve: Right.

Leo: And this is where really we've got to get the courts up to date because metadata is valuable. It does have - is identifiable, ultimately, and has a huge privacy implication. It's not just, oh, metadata. And so this is like a pen register search. This is like when they go to the portals run by the phone companies and say, "Hey, where was Leo on Friday?"

Steve: You, I mean, we've discussed this of course in the context of Snowden and all of that. But you can easily make the case that metadata is a far richer source of information for research and plumbing, even than knowing what the person's yammering about, about their dry-cleaning and whether they need to remember to get cat food or what. I mean, who you talk to and when, your six degrees of separation, that's really vital information.

Leo: Well, we know it is because they want it because it helps them track terrorists. So we know it's valuable.

Steve: And we know that people knock on, you know, that the government knocks on people's doors, saying, hey, you're a friend of so-and-so. Tell us about him.

Leo: Yeah. Yeah. That's just what you're going to get. Isn't that nice.

Steve: Yeah, yeah. So, okay. AT&T got some props, I guess is the current jargon.

Leo: That's how the kids say it, yeah.

Steve: Yeah. For ending their supercookie injection "testing," in air quotes. On the other hand, in digging into this story further, I'm less impressed. Essentially, so first of all, this is the Verizon and AT&T supercookie, which their equipment is injecting into their subscribers' query headers. I detected it on both my AT&T and Verizon WiFi accounts. So I saw it myself. As soon as I switched off WiFi, which was preferentially handling my broadband, my Internet connection, then immediately I started to see their injection of these X-UIDH and other query headers, which contain a serial number which is about me. It is associated by my account with the carrier and makes for a tracking supercookie.

We know that it's actually being used, I mean, the concern when we first talked about it was that it was overly strong, that is, unfortunately, the nature of what they're doing to inject something that's unshakeable, can't be deleted, can't be incognito-ized or regret-erized or anything. I mean, this thing is stuck into your query after you've lost control of the traffic. The concern was that third parties could also use it because all they had to care about was that it was static.

Well, now we have evidence of that. In the developer notes for one of Twitter's acquisitions last year, a company called MoPub, which bills itself as the world's largest mobile ad exchange, it explicitly uses Verizon's tag, the supercookie, to track and target cell phone users for ads pursuant to instructions on the software developers' page. I have a link in the show notes where you can scan through. And in fact I tweeted this link because, as I was scanning through it, just to verify that it was there, there is so much else there, it's just chilling. It's like, okay, I mean, the ad, the guys that want to track us get a ton of stuff. And so again, the Verizon and AT&T supercookie are among what they get.

Unfortunately, AT&T's suspension looks to be just temporary. I mean, they desperately want this. So they're calling what they were doing a "test," which they have ended, and sort of rolled out an end to it, although at no point are they saying they learned a lesson. Maybe they ended it as a function of some backlash, although even that's not clear because they're still talking, even now, in discussions of this being over, talking about, well, creating a code that changes every 24 hours. And of course we've talked about the idea of changing the code, how if there's any other information that allows bridging between changes, then changing the code doesn't help.

But if nothing else, there has been a certain - certainly there's been a bunch of backlash, and the companies are going to have to address the privacy concerns. Remember, this has been going on for years, and it just only recently really came to the fore. So at least it's a good thing that it did.

Just this morning, on November 18th, the EFF has announced an interesting initiative called "Let's Encrypt." They have the domain name LetsEncrypt.org. And next summer, so summer of 2015, they will be launching, the EFF, along with Mozilla and the University of Michigan and Cisco is involved and a couple other large companies, but I think primarily the EFF and Mozilla, and I think...

Leo: This is so great.

Steve: Yes. It is a free certificate authority whose mission is, as the title implies, Let's Encrypt, to get the rest of the Internet encrypted by creating a facility which can issue free certificates. Now, more needs to be seen about how this is going to work. And in fact I think very soon I'm going to cover the protocol. There's a protocol that's been developed called ACME, and I guess we would pronounce it ACME.

Leo: Yeah, of course I would, yeah. So they're automating this, so that the cost of an extended cert, for instance, you can't make that free. But this will be automated.

Steve: Correct. So ACME stands for Automated Certificate Management Environment.

Leo: Got it. Cool.

Steve: And what little is known so far, and only little because all this happened this morning, and I'm busy prepping for the podcast and pulling all of this together, GitHub has the protocol. So there's a JSON over HTTPS protocol. JSON of course is JavaScript Object Notation, which is the approach that sort of has won the war of XML and other approaches for sending data back and forth, the idea being that somehow this is a protocol between the server and the certificate authority. So presumably, and I'm just making this up at this point, the certificate authority queries the server in some fashion for something. And there's a conversation that involves this protocol, this ACME protocol, which allows the certificate authority to confirm the identity to some level of certainty, enough so that it's able to issue a certificate.

And another part of this initiative that the EFF makes very clear is they talk about how it can take a couple hours for an IT person to bring up security on a web server that doesn't have it, just because it's never been made easy. You've got to dig into man pages and look through sort of strange cryptic command line stuff and generate a certificate and send it to the CA and get the result and fix it in, you know, set it in. And then you've got to bind it to ports and open ports and blah blah blah.

Anyway, the point is part of this is automating all of that, too, so that, for whatever server families they support with this, their goal is to make this one click. That is, you download something, and you run it. And if you're running Apache, it knows about Apache config files and configuring and the ACME protocol. It downloads what it needs. It installs things. It edits the config files. There's a complete log of everything done, all configuration changes made, so that you're able to back out. Or you're able, first of all, to audit them and then back out of any or change any that you don't like. It then contacts the CA, this whatever certificate authority this will be that will be established by summer of next year. And the CA, through this interaction, gets enough confidence with DNS lookups - oh, and apparently they will be also using Google's Certificate Transparency logging system to further heighten their confidence in their ability to issue certs, which is good, for example. I don't want anybody else to be able to get a GRC.com cert.

So we absolutely need protection against mis-issuance, and enough attention needs to be given to that. But that's, presumably, that's what the protocol will do. So it won't be able to issue, for example, extended validation certificates. And that's good. We want those still to mean something. We want those to mean that extended validation actually did occur. But the goal here is to, over time, make security ubiquitous. And the way to do that is make it free and make it easy. And his initiative, this Let's Encrypt initiative, the goal is to do both. So I'll know everything about it next time we talk about it because it looks like a perfect topic.

Leo: That's great. And you know, kudos to EFF and Mozilla and Akamai and Cisco and IdenTrust because, frankly, Google should be on that list because Google prompted this whole thing with the change in how Chrome deals with certs. Right? I mean, they kind of forced this HTTPS everywhere. Why aren't they doing this or involved in this? Google, put a little money into this, would you? Give us some 20% time here.

Steve: They certainly are doing the right thing with certificate transparency, along with DigiCert, who's the only other commercial or the only other CA right now that is running a certificate transparency facility because DigiCert's been working with Google from the beginning on this. But Google is providing the stuff from their end.

The other thing that is interesting, and this was mentioned along with this announcement, and that is the notion of short-duration certificates. Remember that what we have now, because the issuing system is so burdensome, you know, it's annoying to have to go through all of this every two years or three years. I mean, even that period of time, it's like, oh, my god, I've got to go figure out that again. I mean, it's just long enough that you've completely forgotten how to do it, and then you have to figure it out again. So...

Leo: That's a clever design.

Steve: So the alternative, if you think about it, is, rather than having long-life certificates, which you may need to revoke, like, early in their long life so that you kill them off for the balance of their otherwise valid period, imagine instead issuing short-life certificates. If you were to issue certificates that only were valid for a few days, then you never need to revoke them. You just stop issuing them, and they expire.

So another interesting approach here is the concept of let's not worry about revoking certificates that have years of life left. Let's only issue certificates that never live more than a few days. But if we're going to do that, we have to make the process automated. We need to have exactly something like this, some way for the server, in the same way that right now servers can reach out to the certificate authority and ask for updated OCSP status, if you were to implement an OCSP must-staple rule in order to say, here is a reassertion from the certificate authority that this certificate is still good. I mean, clearly this is a kluge.

So an alternative is that the cert lasts only a few days. And the server daily goes and gets an updated certificate. And it's only allowed to do that, clearly, if the certificate is still valid. So it's a different model. And this looks like a really interesting step in that direction. So I'm with you 100%, Leo, this is a great-looking initiative.

Leo: Yeah, yeah. Excellent.

Steve: And I would argue there's still a position for extended validation certificates, or certificates that are asserting to a much stronger degree the identity. So all this is doing, this is saying this domain is associated, in DNS, this domain name is being served by this server. That's the assertion being made. So that's a distinction we need to make. That is, nowhere is EFF in this system, the Let's Encrypt technology, nowhere are they making an assertion about the entity that owns the domain. They're making an assertion about the domain name to IP and server mapping. That's what they're doing. So, and they're automating that and making it free. So that's valuable. But I want green in my URL, and I want the browser to be able to make the assertion, this is Gibson Research Corporation, and someone, a certificate authority, went to the trouble of, like, phoning us, talking to us, checking our D&B records, verifying addresses and...

Leo: Flying a dirtbox over you.

Steve: There's a real organization here. That's the role now for tomorrow's certificate authority in this new automated free certificate world. But not everyone needs that. Blogs don't need it. Small sites don't need it. But they'd like to have security. They'd like to have encryption. They'd like to have privacy of their communications. And note that none of this Verizon and AT&T supercookie stuff works if you've got HTTPS. That blocks it all.

Leo: Isn't that good news. Yeah, we want to be, we want to do - in fact, DigiCert gave us some certs. We just haven't, because it's a lot of work apparently, installed them. But of course we want to be able to offer HTTPS. But if somebody poses as TWiT, there's not a lot of harm there. I mean, it's not like we do eCommerce. But it'd be nice to have a cert.

Steve: Right, right. And I'm absolutely sure that there will be technology in place. When they refer to using certificate transparency and some other online facilities for verifying, somebody else would have a big difficulty issuing themselves a TWiT.tv cert because you're already - you exist. Your certificate is there. So any - I'm sure this system will be making those, will have those checks and balances to prevent mis-issuance.

Leo: And by the way, I donate monthly, I have a monthly subscription to EFF, and I encourage everybody to do so. They just are so - they're doing God's work. EFF.org to support them. And, yeah, I literally have an automatic monthly donation because it's just such a good - and I want to sustain it, you know.

Steve: Yeah. So WhatsApp was in the news. They were, of course, acquired early this year, early in 2014, I think around February, for $19 billion.

Leo: Twenty-two. When you calculated the inflated Facebook stock, it went up to 22. Twenty-two.

Steve: Yup, by Facebook, in a cash and stock offer. They recently integrated Open Whisper Systems' "TextSecure" into WhatsApp for Android.

Leo: So awesome.

Steve: Which is really cool. And it's enabled by default. So TextSecure we were just speaking about because an independent audit of it, which was possible because it's open source, concluded that, with a little tweak, fixing something that was found, and the TextSecure guys already were on it by the time this news was out, it passed with flying colors, with that one exception, and that's being fixed. So WhatsApp, of course, has 600 million users worldwide. WhatsApp for Android has this. We're not quite sure when iOS integration will be happening. But we presume it will be.

The question I had, of course, is, okay, wait a minute, how does authentication happen? We know that there is a good secure handshake, that that's easy to do. It's easy to - the technology cryptographically for negotiating keys on the fly exists. But if you don't have authentication, then you're subject, you're subjected to potential man-in-the-middle attacks.

So I did a little digging because I hadn't looked at TextSecure closely. As for the security, that is, the privacy of their messaging, they say: "TextSecure automatically detects" - and this is TextSecure in the WhatsApp app for Android - "automatically detects when a message is received from another TextSecure user" - which will mean another WhatsApp user - "and prompts you to initiate a secure session. If you choose to initiate the secure session, a key exchange will ensue, and a lock icon will be displayed in the title bar of the conversation view, as well as on the Send button itself." So there's visuals showing you that this is now secure. "A lock icon will also be displayed next to each encrypted message received, in order to confirm that it was transmitted securely," that is, that your dialogue remains secure. But that still doesn't answer the question of authentication, which is crucial.

Next they said, under Verifying Keys: "It is prudent to verify the identity key of a conversation's recipient, in order to ensure that no man-in-the-middle attack has occurred." Because, again, you can know that the dialogue is encrypted. But unless you can verify the key you are encrypting with or encrypting to is actually owned by the person you believe you're having the conversation with, then somebody else could have imposed themselves in between. Yes, you've got privacy, except you've included a third party, which is certainly not your intention. So "...in order to ensure that no man-in-the-middle attack has occurred. From the menu in a conversation, select Secure Session Options. And then under there is Verify Recipient Identity. This will present you with an option to manually verify the recipient key's fingerprint, or to verify it by QR code scanning. If you're physically located in the same space as the recipient, you can select QR code scanning to quickly verify each other's fingerprints."

Leo: This is like Threema.

Steve: Yes. And this is the only way to do it. This is why, I mean, this is why Threema, what I liked about Threema was they got it. They put authentication right up out in front. That was the focus. And but TextSecure has done it, too. "If you're remotely located, you can manually read the fingerprints to each other over the phone." So that's an out-of-band - both of those are either optical or some other means, acoustic in this case, out-of-band verification. But that you need to do.

It says: "Once you verify that the recipient's identity is correct, this information is saved and used to automatically authenticate future secure sessions with that recipient." So anyway, so they've nailed it. And so this technology is in WhatsApp for Android, probably coming soon to iOS. And it's worth just once either showing each other's WhatsApp's QR code, crossing that identity. And the fingerprint is just a hash of the key. And so you could just read it over the phone or fax it or send it to their email. I mean, you'd like the channel to be as secure as you think is necessary. The idea being, though, that you just once need to make sure that you've actually got the other person's, the true other end's key. Once that's done, it's stored locally, and it verifies all further communications.

So it's very nice, if this is going mainstream. And one that happens, see, my problem with iMessage is they've sacrificed all of this in the name of convenience. But in the process we do not have a system that cannot be eavesdropped on. What TextSecure is giving us in WhatsApp is that, a conversation that we absolutely know cannot be eavesdropped on because we're given the responsibility of managing the keys. It's the fact that Apple does it for us that is the Achilles heel in iMessage. Which is not to say that they're decrypting conversations. They're not, as far as we know. But we know they can.

Leo: And if it were a demand from federal law enforcement, they'd have to, and they might not be able to tell you they are.

Steve: Well, you know that it would be...

Leo: Now, WhatsApp is not open source. So how do we verify that the implementation within WhatsApp is secure?

Steve: Yeah, I think at that point you just increase the thickness of your tinfoil.

Leo: So you can't be sure.

Steve: Yeah.

Leo: You know, the problem with any of these solutions, Threema or TextSecure, is you have to get your friends to use it.

Steve: That's a good point.

Leo: And WhatsApp is so widely used.

Steve: Yes. That is the benefit is 600 million worldwide now. And, yes. And so it's way easier to overcome that with WhatsApp than it is with Threema.

Leo: And I think you're seeing the benefit of the Facebook acquisition here because we know that Facebook and Apple and Microsoft and Google hate it that the NSA is demanding information from them. They all have transparency reports in which they're saying we want to tell you, but we can't, what people are being asked to do. But this way they secure it.

Steve: Yeah, right, these companies are suing the federal government for permission to talk.

Leo: Right. So I think you could probably safely assume that this is why Facebook's doing this, and they're making darn well sure that they don't have access to the keys because, if they did, then they'd have to hand over the data, and they don't want to.

Steve: They don't want to. Exactly. Now, okay. I'm not going to go off half-cocked the way the rest of the industry has because there's some concern over BitTorrent Sync's security and privacy. It's just been raised. Everyone is running around suddenly saying, oh, my god, we shouldn't be trusting BitTorrent Sync. BitTorrent has themselves, in their own marketing side, claimed that Sync is performing eight times faster than Google Drive, 11 times faster than OneDrive, and 16 times faster than Dropbox. So that's their push. Of course, it's super popular among people who are using it because it is, it's using the BitTorrent protocol, which is strong and mature. It allows you basically to keep a whole bunch of different devices all synchronized. And if one goes down, other ones are able to provide the data. The problem is that some hackers, I have to make sure I say this right because the original term, of course, is cogito ergo sum.

Leo: I think, therefore I am.

Steve: Right. And so these guys are Hackito Ergo Sum.

Leo: I hack, therefore I am. Ren Descartes is spinning in his grave, yes.

Steve: So they're claiming that there are...

Leo: Hackito...

Steve: I know. They're claiming that there are, quote, "probable vulnerabilities in the client," only because they've made it crash. So we know that crashing is not the same as takeover, but it's not a good sign - "and that the protocol can leak potentially sensitive hash and client IP data." Well, okay, it's not content. It's hash. And anyway, so we're, again, this just happened. I'll follow this, and I'm sure my Twitter followers will make sure I see any updates to this. I think it's way premature to worry. I know that BitTorrent is working now on a full rebuttal and response, which wasn't available at podcast time. So, and I'm tied into them. I'm on their PR list. All I want is a protocol documentation. Instead I just get PR nonsense from them, from their PR people. But it's like, okay, fine, at least I know what they're saying. And of course BitTorrent adamantly disagrees with that characterization, so.

Leo: Well, just open, just give us the code.

Steve: I know.

Leo: Here's something the chatroom just told me about called Syncthing that is open source, same idea.

Steve: Yup.

Leo: And it has implementations for Windows, Linux, Mac, BSD, and Solaris. And it's open. So, I mean, if it uses - I don't know if it uses BitTorrent protocols. Those are widely known.

Steve: Yes. Yeah, and there was - we talked about it on the podcast. There was a reverse-engineering effort of the BT Sync protocol. And I think that effort panned out. And I think they figured out everything that was being done because it wasn't a huge change from the original BitTorrent protocol. They just had - they added a layer for doing interdevice, peer-to-peer syncing of files.

Leo: And Syncthing works kind of similarly to BitTorrent Sync. You just give people your ID number, and they can sync with you.

Steve: Nice.

Leo: Yeah. I'll have to try that. This looks kind of cool.

Steve: So, miscellaneous gizmos, miscellaneous things.

Leo: All right.

Steve: So, as expected, Sunday's TWiT discussion...

Leo: Oh, thank you for helping us with that, by the way.

Steve: ...of Net Neutrality was great. I thought that was...

Leo: We used your buddy, Brett Glass, who's been on this show.

Steve: Yeah. And I thought - and your Sonic guy, I thought that the dialogue...

Leo: Isn't he great?

Steve: ...between Brett and the Sonic guy, the Sonic.net guy...

Leo: Dane Jasper, the founder of Sonic.net, yeah.

Steve: Really, really useful. And I don't think anyone reached a conclusion. I don't think there is one. But for a sane, level-tempered, just really thought-provoking dialogue, I wanted to make sure that our listeners remembered that the first hour of TWiT last Sunday was that. It was, I thought, really interesting discussion.

Leo: Thank you. I appreciate that. Yeah. We were trying to, you know, shed more light than heat. And unfortunately, this is one of those things that, if you don't kind of come down on the right side, you hear from people. But we thought it was more important that people hear all of the pros and cons on each side so that they can make up their mind. And...

Steve: Yeah, the problem is it's complicated.

Leo: It isn't easy.

Steve: Yeah, it is a complicated - and where we are today is a mess.

Leo: Right.

Steve: And so, like, how do we get from the mess we're in today to something that makes sense? And, I mean, I completely agree with Brett, for example, saying that Title II is the wrong thing. But, unfortunately, it's all we've got. And so the idea of reclassiflying - reflying - reclassifying ISPs as common carriers and thereby subjecting them to all of Title II, you know, ISPs look at Title II and go, wow, this is awful. This was written in the Stone Age compared to where we are. So, and as you guys said, one of the many suggestions, I think this was yours, was, well, can we get legislators to start over, do something correct? And the problem is, wow, what a heavy lift that is.

Leo: Yeah. I don't think that's going to happen, either. You should listen. I think you'll have an opinion. And then this is just the beginning of what we all need to go through, some homework to figure out what's the best solution.

Steve: Yeah, I would argue no conclusion is reachable. Yet being informed, you know, you can't ever have too much information.

Leo: Right.

Steve: And that first hour of TWiT was really this really useful discussion and information.

Leo: TWiT 484. Listen, if you have it.

Steve: And I forgot to mention, last week I was going to mention that TWiT in the previous week, Dvorak said he thought Security Now! was your best podcast.

Leo: Yeah, he loves you.

Steve: It's like, holy - what?

Leo: I was amazed that he listens.

Steve: Yeah. Like, yeah, he seems impatient. But hello, John. Thanks. Okay. Interstellar.

Leo: Yes.

Steve: This is going to seem so, so geeky. I mean, I don't - I'm normally not this geeky, I think. I don't sort of have a self-identity of being this geeky. But here was my problem. If they just - and this doesn't really give it away, so there's no spoilers here because it's no need to go into that. If they just called it a "spatial anomaly," then I would be fine with that. But they call it a "black hole." And then it did not behave in any way like a black hole. And that annoyed me. So...

Leo: Well, what specific - okay.

Steve: So, for example, information can't come out past the event horizon of a black hole.

Leo: Well, that was silly, yeah. But that's why they had to go in. No, that's - but really, no, Conan O'Brien said it best. He said people are worried about scientific inaccuracies in a black hole, but it doesn't bother them that Matthew McConaughey is an astrophysicist?

Steve: Yeah.

Leo: It's a movie. Now, I have to point out that the Nolans wrote this based on the writings of Kip Thorne, who is an astrophysicist and a black hole expert. Kip was there the whole time they made the movie and did the calculations.

Steve: On valium.

Leo: Well, even he says that the visualizations that the filmmakers make gave him some insight into this spinning black hole that he never would have had otherwise. It was very valuable for him. You know, the Bad Astronomer, Phil Plait, said this whole thing of a planet being on the edge of a black hole is impossible.

Steve: That, too. That was annoying.

Leo: Well, then he took it back. He said, "I'm sorry, this is a very special kind of anomaly. It's a spinning black hole."

Steve: It's a Hollywood anomaly.

Leo: Well, no, Kip Thorne, I think, well...

Steve: Oh, a spinning black hole, okay.

Leo: There are such things, apparently. And the physics of it and the calculations...

Steve: That is true.

Leo: ...are very, very complex.

Steve: As matter condenses, you can get a lot of centrifugal spin, yes.

Leo: Rotation, yeah. And Phil said the calculations on spinning black holes are - if you think calculations of black holes are complicated, these are more so. And he said, I apologize, I was wrong, the planet on the edge of the black hole could exist, and the time anomaly they describe in the movie is possible.

Steve: Okay. Because of relativity, which is your catchall. Okay. Here's the problem, though. And that is gravitational gradient. You cannot, in your spacesuit, fall into a black hole.

Leo: I know.

Steve: You are shredded at the molecular level.

Leo: I know.

Steve: Gravitational gradient, as they say, will get you every time. And so in Star Trek I don't have a problem if they make up absolute nonsense and call it, like, possible. My problem is, if they take something that we know a lot about, we know a lot about the behavior of a black hole, and then just ignore it all. So, however, I thought it was a great movie. I mean, it was fun. It was great special effects. And after about half an hour Jenny nudged me, and she said, "This is different than I thought it was going to be." Because of course we spend a lot of time on the farm.

Leo: It's very emotional. There's a lot of good stuff.

Steve: Yeah. Yeah.

Leo: Yeah, I thought it was a great movie. I think it will be considered a classic like "2001." By the way...

Steve: Oh, no.

Leo: Oh, yeah. You watch.

Steve: Really?

Leo: Ten years from now let's talk. And we know we'll be doing the show 10 years from now.

Steve: Betcha. Nothing's going to stop it. We're in a black hole.

Leo: AMC has announced that they are going to offer unlimited tickets for people who would like to see this movie an unlimited number of times. If you've already gone to an AMC...

Steve: One was more than enough.

Leo: You really will feel like seven years have gone by.

Steve: I could not sit through that again.

Leo: They said for 15 bucks, if you're already seen it once, you can by unlimited tickets, see it as many times as you want.

Steve: See, that's a safe bet for them.

Leo: Did you notice it was really loud? This is what Scott Wilkinson was talking about on Saturday.

Steve: I was very aware of the power of the soundtrack and that it really, if you didn't have that, you'd be like, okay, now, what are they doing now?

Leo: Yeah, right. What is he doing?

Steve: Are we supposed to care about this?

Leo: What is that?

Steve: Is this dramatic or not? I really can't tell.

Leo: Hundred dB plus. In fact, I think he measured it. He brought a sound meter. Talk about geeky.

Steve: Of course he did.

Leo: I think he measured, he said, 117 dB at one point.

Steve: Wow.

Leo: Yeah.

Steve: So I don't want to put anybody off of seeing it.

Leo: So did you like it?

Steve: I was really - yeah.

Leo: A little bit. You liked it a little bit. You wouldn't go see it again.

Steve: No, I couldn't. It was two hours and 40 minutes. It was a long movie. And just kind of, I don't know. It was a little too pop for me. I guess that's the way I would put it.

Leo: You wanted more science.

Steve: Nice how everything wrapped up; and it's like, okay, well, we don't have any loose ends. And of course there is the fundamental problem, and I don't want to - I can't really say more because I refuse to do a spoiler. But there's the big causality problem of you can't get that until you already have it, and then - so it's like, okay.

Leo: It's a movie, Steve.

Steve: All right.

Leo: You've got to have some, you know. Did you like "Back to the Future"?

Steve: Loved it.

Leo: He saves his life by going back and making sure his parents meet.

Steve: Yeah, I guess that's a problem.

Leo: Remember, he has a photograph, and he starts to disappear?

Steve: Yeah, but he had a hoverboard.

Leo: Oh, the hoverboard made it all right.

Steve: Actually, I don't think that was that one. That was one of the...

Leo: It was the second one? Yeah.

Steve: "Back to the Future Again." Okay, so...

Leo: There is a movie, though, that you do want to see.

Steve: Yes, yes, yes, yes. Comes out Friday after Thanksgiving, November 28th. And it is - I saw an interview - what was it on? I saw an interview somewhere of several of the actors who star in the movie. It's called "The Imitation Game." And in fact the website is TheImitationGameMovie.com. And I think you can get the preview there. It's very dramatic-looking, with lots of stuff going on on the web page. But anyway, it's the history or the story of Alan Turing and Bletchley Park and their decryption of the German Enigma machine ciphers, and how vital it was and all of that. But apparently strong on being a tribute to Alan Turing. So...

Leo: Yeah, we have a viewer who saw it and said it's great, and a heart wrencher. So how do I get this site started here? I just - do I click? Do I rearrange? Seems like there's stuff going on.

Steve: I had scripting off, of course.

Leo: Oh, then it worked.

Steve: It kind of limped along and did stuff, but clearly a lot of attention was paid to it.

Leo: There's some real scripting going on here. Wow. Wow, all right. Well, I'm not going to be able to see the trailer unless I solve this crypto game, apparently.

Steve: I would imagine, if you just put "The Imitation Game" into Google, IMDB probably has it with a lot less nonsense going on.

Leo: Well, now I've got to solve it. All right, go on. You know what I'll be doing for the next hour.

Steve: One of the main actors, not ones interviewed, but one who was not sympathetic, I was looking at him, it's like, why do I know him so well? I mean, we already know him. But it was like, why do I know him so well? It's like, ooh, he's the evil father king on "Game of Thrones."

Leo: Oh.

Steve: And so they brought him out to play...

Leo: He's a good actor, yeah.

Steve: Yeah, he really is, yeah. So, looks like it's going to be great. And keeping in with our Q&A theme, I thought I would answer a listener's, because I encountered this reading the mailbag, answer a listener's question about full disk encryption with SpinRite and SSDs. That had never come up before. Greg wrote, he said, "My routine before I do" - and I should mention before he's apparently a Linux guy.

"My routine before I do full disk encryption is to SpinRite" - to turn it into a verb - "the hard disk drive at Level 5" - so a full-strength deep SpinRite - "the idea being to expose as much of the disk as possible before overwriting it with pseudorandom data. Following this, I installed the encryption container using Linux's dm-crypt and LUKS, L-U-K-S. Considering the wear that SSDs incur from Level 5 SpinRite, do you think this is a good idea to do on SSDs prior to overwriting them? For security, the SSD has to be overwritten prior to putting the encryption on. So the wear incurred is necessary. But does my use of Level 5 before overwriting make sense to you from a security perspective, or do you recommend I just use Level 2 for SSDs that I want to encrypt?"

My feeling is there Level 5 is what you want. Because once upon a time when error correction was relatively weak on hard drives, and drives were a lot slower, or a lot smaller, SpinRite wrote a whole bunch of patterns. Its strongest pattern testing did a whole bunch of stuff. Now, because error correction, as density has increased, out of necessity the ability to do on-the-fly error correct has been strengthened dramatically. And of course chip density is higher. That supports much stronger algorithms for correct. The chips are much faster, and so they're able to do more on the fly and so forth. So as all of that scaled up, and drive size scaled up, SpinRite's - the surface analysis and testing that SpinRite had to do was scaled down in proportion. So now what SpinRite does is only two writes. It reads what's there, inverts it and writes that, reads that, inverts it again, and writes it back. So that's just two writes.

Leo: And that doesn't make a wrong.

Steve: Actually, because it's inverting, it puts them right back where they came from.

Leo: There you go.

Steve: Exactly. And then you're going to run encryption which is going to go across the entire thing again. So if you did nothing, if you did no writing, then you'd write everything once. If you use Level 5 on SpinRite, you're writing everything three times. And compared to the writing that you're going to be doing for the rest of the drive's life, that's nothing. So, and it does make sense, I think, to run SpinRite over an SSD to help the SSD immediately find any areas that it might want to swap out at that point.

Now, one thing that you didn't mention, and that everyone should think about, is after you put encryption on the drive, you really then want to reset trim because both running SpinRite on an SSD, where SpinRite is doing any writing, or doing whole drive encryption, will saturate the drive's trim bits. Remember the trim is an extra facility in SSDs to dramatically speed it up, where it operates sort of at the physical level of the SSD to inform, to sort of remember whether anything useful has ever been written there. And if you run SpinRite on an empty drive, you've written nothing useful. So you absolutely want to run a trim-clearing utility after that. And without giving away too much, that'll be an option in the future, at my end.

But if you then encrypt the SSD, you're writing all the sectors again. So what you need is you need to be able to run some sort of trim-clearing utility on the post-encrypted SSD to get it to relax, essentially, to tell it that, even though a lot of writing has been going on, we don't care about any of that because then you're going to put the file system on the drive. And that's going to be doing the first actual writing that we care about. So I would look into post-encryption trim-clearing on SSDs. I'll be able to do post-SpinRite trim clearing, but that wouldn't help you in this case because you're then going to write the entire drive one more time.

Leo: All right. And we'll leave it as an exercise to the listener to figure out how to do that.

Steve: There are lots of utilities for doing that.

Leo: Why is this at the end of your show notes?

Steve: That's at the end of the show notes. If you zoom in...

Leo: It's a picture of a CD.

Steve: It's actually a very special DVD called an M-DISC.

Leo: What is an M-DISC?

Steve: And it's made by Hitachi, and with an interesting name, which I was able to pronounce.

Leo: Millenniata.

Steve: Millenniata. I love that. This refers to one of - I'm glad you asked the question, by the way, Leo. This refers to one of our five questions. This is a 1,000-year archival storage.

Leo: Millenniata.

Steve: Literally written - are you sitting down on your ball? I think you are.

Leo: I am.

Steve: Literally written in stone.

Leo: What?

Steve: It doesn't use dies. It uses inorganic material, basically stone. It melts the stone.

Leo: Wow.

Steve: And so it inscribes 4.7GB in stone.

Leo: Yeah, I like it.

Steve: Yeah.

Leo: Wow. We'll talk about that because our question-and-answer section has arrived.

Steve: Yes.,

Leo: Yes. And we will start with Question Numero Uno. Let me just take the Millenniata out of the way. It's kind of a big - I blew it up. It's real big now, and I can't see the questions anymore.

Advait in India wants to know about going to your servers at Level 3. Steve, on the past episode, you said you had to physically go to your servers to update them for the Schannel bug. We now know you're going to have to do that again, apparently. But so why do you need to go to them? Can't you just do it remotely? That's a good question. I thought you could pretty much do all Windows Server maintenance remotely, including power cycling, if you have the right equipment. Don't most co-los allow for remote power cycling? I know ours does. Just curious. Thanks, love the show, happy SpinRite owner, et cetera. Advait.

Steve: So I just wanted - I thought this was an interesting question. I had several other people ask, like, wait a minute, you know, why are you going there? And the only reason is I have been so well trained by Murphy. I am a disciple of Murphy. And this was, as I mentioned, since I'm not rebooting them all the time, like in this case it's been six months since I have had to touch them. And that meant that I'd be doing a lot of patch catch-up. And we've all had the experience of doing a big update and rebooting our system and the screen stays black. And for GRC, I just - I didn't want to be here at home, clicking things, pressing buttons, waiting for the server to come back up, and have it not. So I wanted to be there where I was physically present and could deal with anything that might happen. And as it was I made fresh images and did some other, you know, just sort of cleanup stuff. And, you know, I just never visit.

So it was nice to sort of check in, and it feels good to sort of see that everything's the way I left it. And I think, frankly, this little update, I probably won't bother making a trip, since I just did a week ago. I will make sure that I've got a current set of images and backups. But this one I will do remotely. So, and I have power control equipment, and I can do power cycling and all that stuff remotely. But this seems like, well, you know, it's been six months. It's worth a trip. And as it happens, everything went just swimmingly.

Leo: Oh, good. Oh, good.

Steve: Yeah.

Leo: Charles Victorian, great name, in Houston, Texas, has some SQRL implementation help questions: Steve, long-time listener, blah blah blah, all episodes, blah blah blah, Security Now! University graduate, et cetera. You and Leo rock, blah blah blah. He actually wrote all of that, by the way, folks. I'm not making it up.

In pondering SQRL, I was wondering/hoping for three important things: Would you please help with the creation of tools or libraries or simple instructions, et cetera, so that web developers not on the rocket surgery level can easily implement your strategy for login? Dumbing it down may be important for this to catch on like wildfire. Two, would you please create a sample login page on your website which would allow people using your SQRL reference implementation, or any other really, to have a known working location to experiment with logging in via SQRL? Three, would you please make an iOS SQRL client yourself so that we know it's as TNO as possible? I, and perhaps many others, apply the TNOBS principle - Trust No One But Steve. We'd pay for that app. Not trying to be mean to other iOS devs, but they haven't earned your white hat reputation yet. I'm just sayin'. Thanks for all your hard work to both you and Leo.

Steve: Okay. So a couple things. Other developers - so the first point was what about dumbing it down or making it easy to implement. There will be drop-in packages for all the major web server-side stuff, Drupal and PHP libraries and all of that. That's all in the works and underway. And there are some that are even up and running at this point. So it won't be necessary for anyone to write this stuff from scratch, though once this is all absolutely finalized, I absolutely will have some simple, like, flow charts of what API call you issue when. Other people who are involved in the GRC newsgroups have asked for the same thing.

As for a sample login page, yes, absolutely. There's actually - it's already there, although it doesn't quite do much. I'm using it myself as I'm finishing the final phases of this. GRC.com/sqrl/demo.htm will present you with a valid SQRL QR code right now. And once you have the client, you can either click on it if it's, well, in fact you would in your browser click on it. And Leo, if you refresh that, every time you refresh it you'll see that you get a different - actually yours is blanking the whole page, so it's less easy to see. In Firefox it just changes the QR code, just goes blink.

Leo: It's because everybody has gone to your page.

Steve: Ah. That's right. Sorry about that. Anyway, so that will allow you to log in and create, like, a fake SQRL account and also, like, show you when you last logged in, how many you've logged in, when you've changed your identity, a whole bunch of other metrics and so forth. And then there will be another place, it'll be sqrl/dump.htm, that doesn't yet exist, where all of the crypto stuff will be made visible so that developers will be able to verify that their crypto exchange is the same thing that GRC is seeing. And that stuff is easy to do. I just haven't gotten around to it. But essentially a pseudo login facility where you create SQRL accounts, that's actually the database on the backend is already there. I just haven't brought it out to the web service yet.

And as for iOS, no. I'm going to be going immediately back to SpinRite 6.1 and returning to the development of that. I don't know of an iOS developer except that Ralf had mentioned - Ralf is the person who has done the Android client implementation that is up and running and works for Android, that people can download. If you have an Android device you can get SQRL for it right now, although Ralf has not finalized it until I finalize mine. And there have been some protocol tweaks. And he did reference, months ago, doing one for iOS. I'll be happy to work closely with any iOS developers to, like, have it get my seal of approval. But that's the most I can do.

I just - I can't take any more time away from SpinRite. Maybe after 6.1 is behind me. We'll sort of see where things go. I also don't want to spend any time if it never gets off the ground because that would be sad. But on the other hand, it will help to have an iOS implementation for it to get off the ground. So maybe I could say that the first iOS SQRL implementations won't be the last, and we'll just have to play it by ear.

Leo: Question 3 comes from Edmonton, Alberta, Canada. James is the one who told us about the stone DVDs, the Millenniata, the M-DISC. Now, I've got to point out, good for a thousand years; but, yeah, a thousand years from now, who're you going to sue if it's not? Oh, man, it's not good for a thousand years.

Steve: So it's a couple years old. It's called the M-DISC, just letter "M" and then DIS - now, I write "K," but I don't know if it's "K" or "C." Sometimes discs are with a "C." So, and it was Mitsubishi that had acquired the rights and was going to produce the medium and the drives. I don't think this thing uses standard DVD writers. So I think you need - because, I mean, it is, apparently it is stone, so it's going to use like a stronger laser.

Leo: I would hope it would use standard readers, however.

Steve: I don't know either way, actually.

Leo: That would be an issue because a thousand years from now, if it's a nonstandard reader, chances of getting one are slim.

Steve: Very good point. A thousand, you know, that thousand years is a long time, Leo.

Leo: Even 10 years from now, the chance of getting a reader might be slim.

Steve: So we had been talking about archiving. And several people, one who is a member of the archives, it's like archivist association, talked about the stone DVD and the M-DISC. So this is the real deal. If you're somebody who wants to or has an interest in storing in allotments of 4.7GB, that is, a single-sided DVD - wait, that's Blu-ray density because a DVD - isn't a DVD...

Leo: No, that's DVD. Blu-ray is more. It says, apparently it does say readable on current DVD and Blu-ray drives. So you can read it. You just can't write it without a special writer.

Steve: That absolutely makes sense.

Leo: You need a chisel and a hammer.

Steve: So they've kept the format identical to standard DVD, yeah. Blu-ray, of course, is in the 40 and 50GB range.

Leo: Yeah. And a single-sided is 4.7.

Steve: Right.

Leo: DVD, yeah. They apparently do make them in Blu-ray sizes. Probably want that.

Steve: No kidding, wow.

Leo: Probably would want that. Question No. 4, Dave in Southampton, United Kingdom, wonders about data recovery on destroyed hard disks: Steve, in a team meeting, a sad and unfortunate story of a soldier in Afghanistan being killed by an IED was told. The story was used to highlight a security problem with hard disks because, although the laptop he was carrying and the hard disk it contained were both destroyed, the data on the laptop was important and was retrieved by taking the disk apart and scanning it somehow to get some of the data back. The story was told to us to highlight the issue, even after destroying hard disks simply by drilling a hole through them or taking them apart and cutting them to pieces with tin snips, this may not completely destroy the data that's on them.

Now, I accept it's possible to get the data back from a disk that's been destroyed in this manner, although I will fall off my chair if you can tell me you will do it with SpinRite. How hard is it to actually do, assuming the disk was not encrypted? What type of equipment would be needed to achieve such a task? How long could it possibly take? My current contention, until you tell me otherwise, is that it is extremely difficult to do, requires extremely expensive equipment and people. Let's assume that he is an attacker, he may not get that data that's of any use to him.

If I'm correct on this, then I assume that an attacker who would have the resources to do this would also have the resources to place an insider in an organization because that would be a lot easier. What do you think? Many thanks. Regards, Dave. P.S.: If this answer is heard on Security Now!, please do not give my surname in case someone from my team also listens, and I get accused of rocking the proverbial boat again. But you can give my location.

Steve: So, okay. The only way to get data back at today's recording densities is to somehow put a disk that can still spin and aerodynamically support a flying head over its surface.

Leo: In other words, isn't bent.

Steve: Right. And it hasn't had tin snips, or hasn't had a pattern of holes drilled through it, all of which would destroy the ability of a head to fly. Older generation drives may have had simpler recovery being feasible. You could actually - there used to be a solution you could actually rub on an old-style oxide disk and then look at it through like a microscope and read the bits. You could actually see the bits because this fluid was affected by the magnetic domains stored on the oxide. But today's densities make that absolutely impossible. So if the story is not apocryphal, I mean, if it's actually true that following an IED explosion - and it wasn't clear whether the bad side got the disk, recovered the laptop and disk. It sounded like maybe our guys got it and were somehow able to recover it.

Leo: The whole thing's made up. Come on. You know it's not a true story.

Steve: Anyway, the answer would be you could, I mean, if you really needed to recover it, you would take the drive apart and mount those platters in the same make and model of another drive and then maybe have a chance.

Leo: And we've talked about this before. It would be possible, I mean, this is governmental level decryption. It's not...

Steve: It is really, yeah, it would be - oh, no, it's absolutely NSA...

Leo: It's not the mob.

Steve: No. No, no, no. It's NSA labs absolutely need to recover the data level. But again, our technology - we talk about this all the time. Hard drive storage is so fragile that it barely works on a good day. So subjecting a drive to any kind of physical trauma, forget about it. I mean, especially tin snips and drilling holes with drills. It really does end - end of life.

Leo: It's hypothetical. It's hypothetically possible that you could read somehow the magnetic field on the disk if you had extraordinarily sensitive equipment and then somehow hypothetically assemble it. You know, we've heard people do kind of outrageous things like reassemble shredder pages.

Steve: Well, and, yes. But, see, there you have the benefit of them being physical things.

Leo: Right, you can look at them.

Steve: Because anyone who actually knows what happens between the user submitting 110011 and what's written will really appreciate there's four different stages of translation. And one of them is called "whitening," where its goal is to average the number of transitions per linear extent in order to keep the write amplifier from being too much biased in one direction or the other. So my point is that even the magnetic image on the drive is only distantly related to the data that was originally written. And only all of the electronics and data recovery in the read path of the drive that wrote it even knows how to translate the magnetic information back to the usable user data. So even if you had, if you could even get the actual magnetic patterns, that's not the data that was written on the drive. You have to go back through the read process. And how do you do that?

Leo: Right. Yeah, yeah. Not to worry.

Steve: Not to worry.

Leo: Final question from Charles - is it the final question? Actually I didn't check.

Steve: Yeah.

Leo: Yeah. Charles in the U.K., we're talking about brute-force encryption: There's one thing with brute-force attacks against encryption that always left me perplexed. How do you know you succeeded? How do you know the results of decrypting with a key is a success if you don't know what to expect?

Steve: You know, this is a great question, and it comes up all the time. And normally we run out of space. And I thought, okay, Question No. 5, since we've got so much to talk about at the top of the show, and as it turns out our timing was about right. Okay. Let's look at two examples. The only way to answer the question is to actually drop it into the real world. Two examples: The first is - and two relevant examples. The first is communications. With an SSL/TLS connection, and the issue is brute-force determination of the key, how do we know?

Well, as we mentioned recently, unfortunately, SSL got the order of authentication and encryption wrong. They got it backwards, the original designers, the well-meaning guys at Netscape. They said we're going to take the user's data. We're then going to authenticate the data, that is, run an HMAC over it or whatever, and have the authentication added to the end. Then we're going to encrypt it. That's unfortunately wrong. As we discussed then, the well-understood now in contemporary crypto knowledge is you always encrypt, then you authenticate, because the process of decryption requires a reversal of steps. So that means, if you encrypt, then authenticate, which is correct, but unfortunately not what SSL does, then that means the first thing you do when you need to decrypt is to authenticate, then decrypt.

But with SSL, it's backwards, which means that the last thing done is encryption, which means the first thing we do is decryption. And that means that we can always test to see whether our decryption was correct because we then verify authentication. And so SSL, by doing it in that order, the only good side, I mean, there's other bad aspects to it, but in this case we can do a trial decryption and then a trial authentication. And if the key is wrong, we will have decrypted the wrong data, which will not authenticate. And so in the case of communications flow, we're able to verify by checking the authentication.

The second real-world example, completely different, is stolen passwords, which we're talking about with websites all the time. Somebody gets a database of password information belonging to some website. Now, what do they have? They have the hashes of the users' passwords. So what they're looking at, what they have is the result of the input password being hashed through whatever algorithm the site uses, which they presumably know because that's typically part of what they're able to figure out in order to crack this.

So the goal is - they've stolen the password database. They want to determine the input passphrase that, when run through the hashing algorithm, whatever it is, how many iterations of algorithm it might be, how much strengthening it has and so forth, they want to determine the input passphrase that results in the hash that they have in the database. So they're not actually decrypting that hash in the database. Rather, they're successively - you can think of it as encrypting or enhashing. They're successively hashing guesses to see if they get the same result.

The reason is, if they get the same result, then they have figured out almost certainly the user's password, or at least a password that results in the same hash, which then allows them to go to that website and log in as that user. And if the user has been unwise and reused that same password elsewhere, and they have any other information about the user that they also stole, like their email address, name and so forth, maybe they can use that same information to log into other websites to impersonate the user.

So those are two examples. In one case, the authentication, which is part of the integrity guarantee of communications, it can be used to verify the decryption, that the guess is correct. And in the instance of a password database being stolen, you're not really decrypting what the database had, but rather you're doing the same thing the server did when the user enters their password to get a result. You know the result, so now you just make a lot of guesses until you get the same result. And that's how it works in the real world.

Leo: Excellent. Or you just, as somebody in the chatroom said, you shout, "I have the message," versus "I have the message z4391x!qtty." Right? Usually it's pretty obvious. Hey, Mr. G, that concludes this edition of the fabulous Security Now! podcast. Each and every week we meet and talk about the latest issues in security. Next week, do you know yet? Or is it going to be a surprise?

Steve: Don't know. I'm really curious about this ACME protocol which the EFF is using. And to what degree, you know, how much they're doing, I want to see how much is there. So if there's enough there, we might talk about that. Otherwise, I've got a long list of stuff to get to. So I'll pick something.

Leo: As always. You can find Steve on the Internet at GRC.com. That's his home on the web. He also has a lot of stuff there you might want, including SpinRite, the world's finest hard drive maintenance and recovery utility, and all the freebies like crazy he gives away. You'll also find 16Kb audio of the show, and transcripts written by Elaine Farris, so they're very easy to read and a good thing to read along while you're listening. Steve's on the Twitter @SGgrc. Questions to Steve can be forwarded to him from GRC.com/feedback. We have full quality versions of the show, audio and video, available at our site, TWiT.tv/sn, on YouTube.com/securitynow, and wherever finer podcasts are aggregated for later distribution through the Internet network of networks. Mr. G?

Steve: As Brett reminded us, it's actually not owned by anyone. It's an Internet of networks.

Leo: It's an Internet of networks. A 'Net of networks.

Steve: Okay. Leo, always a pleasure. I will be back with you next week.

Leo: Thank you, sir.

Steve: See you.