Listener Feedback #143

Description: Steve and Leo discuss the week's major security events and discuss questions and comments from listeners of previous episodes. They 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-352.mp3
Quarter size (16 kbps) mp3 audio file URL: http://media.GRC.com/sn/sn-352-lq.mp3

SHOW TEASE: It's time for Security Now!. Steve Gibson is back with security news and answers to 10 of your questions. It's all next on Security Now!.

Leo Laporte: This is Security Now! with Steve Gibson, Episode 352, recorded May 9th, 2012: Your questions, Steve's answers, #143.

All right. Turn on your modem. Tune in your router. Fire up your browsers, ladies and gentlemen. It's time...

Steve Gibson: And get under your desk.

Leo: ...for Security Now!. Thanks to PWR for writing my intro today in our chatroom. Steve Gibson, the one, the only, the man, the myth, the legend, the Explainer in Chief is here. Hey, Steve.

Steve: We've got it all bundled together in one...

Leo: Yeah, one giant expletive.

Steve: ...giant description, yes.

Leo: Steve is a security wiz at GRC.com. He writes SpinRite, the world's finest hard drive maintenance, and recovery utility. And he is here to answer questions today. This is Q&A #143.

Steve: Well, you dragged me over here to give our listeners some feedback. Or they've given us some feedback, and we're going to...

Leo: Feed it right back to them.

Steve: Exactly.

Leo: Boy, there were some interesting articles this week about security. Wired magazine had a wonderful article in their Threat Level blog, I don't know if you saw it, that said, basically, everybody's been hacked. It's all over. Get over it.

Steve: Actually, I saw that go by, and I just said, uh, okay, fluff.

Leo: Well, it wasn't fluff. It was basically, no, it was quite a good article, and it was basically the premise that, even though no one wants to admit it or say anything, the truth is pretty much anybody who is worth hacking has been hacked. It's kind of some of the stuff we've talked about, which is it's impossible to write perfect code.

Steve: Yup. Security is porous, inherently, I mean, it's really, really, really hard. And the cost goes up exponentially as you increase security linearly.

Leo: And what the point was, and I thought this was actually pretty accurate, was that there is conventional security wisdom. And most people charged with keeping their servers or their business secure just do what's - it's kind of like, if you just do what everybody else does, then you have plausible deniability. You say, well, I'm doing what everybody else does, so it's not my fault. And the problem is that the conventional kind of wisdom is insufficient these days.

Steve: And remember, the old, in the early days of the computer industry, the phrase was "Nobody ever got fired for choosing IBM."

Leo: It's exactly the same mindset. Which is it's a little tough - this is Kim Zetter writing May 4th: "Everyone Has Been Hacked. Now What?" It's a little tough, if you're in the IT business, to be really a security expert. Just do what everybody else does, eh, you're off the hook. Nobody can complain because you've done everything. And so they talk about Oak Ridge Lab being spear-phished. There's a picture of Dan Kaminsky, who said, "There's been a deep conservatism around." This is Kaminsky, the guy who wins that Pwn2Own every time. He says, "'Do whatever everyone else is doing, whether it works or not.' It's not about surviving, it's about claiming you did

due diligence. That's good if you're trying to keep a job," but "it's bad if you're trying to solve a technical problem." That sounds like almost something like you would say.

Steve: Yeah. I agree with him completely.

Leo: Yeah. He says, "No one knows how to make a secure network right now. There's no obvious answer that we're just not doing because we're lazy."

"Simply installing firewalls and intrusion detection systems and keeping antivirus signatures up to date won't cut it anymore since most companies never know they've been hit until someone outside the firm tells them." So I guess the point of the article is, well, okay, so let's acknowledge this. Now what do you do?

Steve: Yeah.

Leo: What is the appropriate response? And basically they say it's a constant battle, as we've said. There's no magic bullet. There's no ultimate security protection. We've just got to constantly manage the risk, assess the risk, and do the best we can.

Steve: Yeah, there are too many potential ways in. And those ways in are all, for the bad guys, are also being used for good purposes. And so it's these ways in, like visiting a site that runs some script that you want to have run because you want the services that the script provides. But in the process, there's a sneaky way, for example, to abuse subtle characteristics. In fact, we're going to talk about the recent iOS update to 5.1.1, which Apple just released, which fixes a couple problems just like this. And we've talked about them all in other contexts, like cross-site scripting, where you can - really clever people can figure out how to get their code to run with your browser believing it's the code of the site you're visiting, which it gives permission to, yet you're running malicious code in this other site's context.

And, I mean, these are difficult things. It's really hard to get all this right. I would say I'm really seeing progress. I mean, I think, for example, gone are the days where most people think they can use their mother's maiden name as their password throughout the entire Internet.

Leo: Yes, yes.

Steve: And that'll be enough.

Leo: And yet, I mean, I don't know, it's not in your notes, so I don't know if you saw it, you probably did and just decided to forget it. We just learned that there has been a flaw for three months in OS X Lion that allows anybody to execute a simple terminal command and change anybody's password, including the sysadmins. That there's another flaw that's been around for a couple of months that logs your password in the clear. There's holes in PHP. We just learned about another hole in

the CGI version of PHP. It's just - it's never-ending. And then somebody just released 55,000 Twitter passwords and login accounts today. It just is never-ending.

Steve: Yup. And again, it's the systems, the way we designed these, they are so complicated that it's why I shudder every time new code comes out. And it's why I'm on, like, I hang back, like really back, because new is not better in security. New is just a whole bunch of new opportunities for problems. And we see it over and over and over is that it takes a long time to sort through and find the problems in new stuff. You're just - you're going to have them. So unless it's something you really need, I'd say, well, if what you've got is working, stay there because at least it's a known quantity.

Leo: That's too bad because geeks like me, we just - we can't wait till the upgrade comes out.

Steve: And Leo, this show is all about helping you pull the arrows out of your back.

Leo: I know what you mean. All right, Steve. We've got questions, I see, but let's dig into the security news before we go any further with that.

Steve: Yeah, well, we are on the other side of the second Tuesday of the month, so everyone knows what that means. It means that it's time to update our machines with Microsoft's latest fixes to things that they have found. This time we have 23 vulnerabilities which are fixed in web browsing, file sharing, and email. Eight of those 23 were rated "critical," which as we know means that no user interaction is required for hacking to result from the exploitation of these vulnerabilities. At least three of them have been circulated publicly before their release, so that's always sort of something to keep an eye on.

And, interestingly, the one that Microsoft considers most critical is their MS12-029, which updates the patch they did last year for the Duqu worm. Remember that was regarded as maybe a relative to Stuxnet, which was the one that famously worked on the process control systems in Iran to mess up their centrifuges. Duqu was believed to be derived from the same code, thus from the same authors. Microsoft fixed the one patch that it was known to be using. But then they realized, ooh, that the same problem exists in a bunch of our other stuff. So this 029 patch fixes Windows Office, .NET, and Silverlight, all that shared the same vulnerability. So anyway, this is one you'd want to update as you normally would all of these things from Microsoft.

Also, Shockwave moves forward, for those people who care. I mean, I understand that sometimes it's necessary. There is gaming that is done with the Shockwave Player on the web. But our standard advice stands, which is get rid of it if you know you don't need it. Or maybe if you even don't know that you need it.

Leo: It's not the same as Flash. It, like, predates Flash.

Steve: Correct. Exactly. Different from that. I'll just quote briefly from Adobe's page. They said, "Adobe released a security update for Adobe Shockwave Player 11.6.4.634

and earlier versions." So if you're earlier than the .634, you want to update. They said this is for Windows and Mac. "This update addresses vulnerabilities that could allow an attacker who successfully exploits these vulnerabilities to run malicious code on the affected system."

So again, it's one of these remote takeover kind of things where you just go to a site, and if you've got Shockwave Player installed, and a site is malicious, this provides an entry that your browser doesn't provide by itself. It's the add-on that creates a door that the bad guys can use. So they recommend updating to .635, if you know that you've got Shockwave Player. Don't install it if you don't have it. But if you need it, you want to keep it up to date, which is standard advice for all of these browser add-ons.

Now, Apple did just release an update to iOS 5 across all their platforms. It took me, over my two T1s with 3GB download, about an hour, a full hour to get this thing. And what's annoying, I'm assuming this is going to be the same, is that you have to do a different one for your WiFi-only iPad versus your 3G or 4G or LTE iPad versus your Touch versus your Phone. So if you're like me, and I'm sure like you, Leo, and we have all of these things, it's like every single one of them needs a whole blob to be downloaded and installed. So you only get to save that if you have a couple of the same thing. So it's a little bit annoying. But it's important.

This fixes three - well, first of all, I got a kick out of it because when I did this on my Mac earlier today, up comes a window and talks about all the happy little things they're fixing. It's, like, seven or eight little improvements. And it sounds like, oh, that's all good. What they don't tell you at all there is that there's an address bar spoofing problem in all iOS prior to 5.1.1, where Site X could redirect you to Site Y but make it look as though you'd gone to Site Z. So essentially it's an address bar spoofing problem, and that's not good because a lot of us who are security aware and just sort of web aware, we're just looking to make sure we went where we intended to and where the indicator thought we were. But there's a way for the address bar to be spoofed which is fixed in 5.1.1.

Another problem that we've discussed extensively in the past is only now fixed. This particular one is a cross-site scripting problem, where you're visiting one site, and as I was mentioning earlier in the show, code sucked in from a different domain could execute in the so-called security context of the site you're visiting. You want the site you're visiting's code to be able to do what it wants, but you don't want some other domain to do that. That is cross-site scripting.

And, for example, that would mean that that other site could access your cookies for the site you're visiting, and things like if you're logged into a site, as we know the way login happens typically, is there's a token that authenticates you which is being sent back with every query. Well, that's what Firesheep famously was able to grab before sites began running exclusively over SSL to lock that behavior down. But this kind of cross-site scripting problem even penetrates that, so that the third-party site would be able to get the tokens that are tied into the first-party site, even if you're running over SSL. So that's not good.

And then, finally, they fixed a remote code execution problem where a maliciously crafted web page could crash your browser in such a way that it ends up running program code that was embedded in the page. Now, that's a less elegant hack of your system. You'd like it to be more transparent and not crash your browser. But still, actually, I'm sure all of us who have been using iPads from the beginning see our apps crash from time to time.

Leo: Sure.

Steve: You're just doing something, and it blip, just kind of goes away.

Leo: Kind of restarts, yeah.

Steve: And it's like, oh, yeah, exactly. It's like, oh, okay, well, yeah. So anyway, time to update iOS. Plan for however your download speed is. How long did it take you with massive pipes, Leo?

Leo: Well, I do it differently than you did it. You obviously used iTunes to do it.

Steve: Yes, I did.

Leo: So you can now do over the air updates, and those are delta updates. So if you do it, you want to plug in your phone or your iPad or your iPod and then just go to the Settings, and in the settings General Updates. It will do it for you. And it's much faster if you do it that way.

Steve: And WiFi and cellular both? Or WiFi only?

Leo: It'll do either, but I would suggest getting on the WiFi, unless you don't have anywhere close to the cap. But I think it's considerably faster. The problem is that for some reason iTunes doesn't do deltas, apparently. It downloads the whole image, which is almost a gigabyte. So, yeah, it would be a lot better just to download the changes. And Apple recently implemented that. But apparently it's only for the over-the-air downloads.

Steve: Yeah. And I have, I did update for the last one, I updated that way. And I just sort of, like, didn't occur to me...

[Talking simultaneously]

Steve: Yeah, so, good, I'm going to do the rest of them that way. That makes a lot of sense. The advent of DNSCrypt from OpenDNS has generated huge enthusiasm from our audience. It was initially only available for the Mac. And I am pleased to announce, thanks to Gavin Groom, who tweeted me from the UK a heads-up, that it is now available for Windows. So OpenDNS users who like the idea of having their DNS encrypted, now can install it for Windows. There was some hack when it was in beta which we talked about briefly. Now it's official.

So the reason this is significant is that DNS is like a little privacy leakage that we really haven't focused on maybe enough because we keep talking about SSL and setting up crypto and having everything you're doing in a tunnel that is encrypted that no one can get into, and that's cool and important. Except DNS, as we all know, is the way our

browsers get the IP address for the domains that we're then going to go into this secret, lower the cone of silence over ourselves with, for those who remember "Get Smart" back in the old days, except the DNS has no encryption.

So even if you were using - if you were in a mode where you were in an open WiFi and smug that all of your services you were using were carefully forced to be over SSL or natively over SSL, one little glitch is that any time your browser looks up any domains, not just the primary domain you're visiting, but remember many domains now bring stuff in from all corners of the globe, all of those domain names are being sent out to the DNS server in the clear. So there's a privacy leakage. They're not seeing the data, but they are knowing that your machine is asking for the IP addresses of this whole set of domains and that, depending upon what you're doing in that setting, that may make you feel uncomfortable.

So to some fanfare, OpenDNS has uniquely, of all DNS services, created their own encrypted tunnel between your machine, when you install the matching client on your side, and their service, so that all queries get encrypted as they go, as they say, over the wire or through the air to grandmother's house, over to DNS and back. So now for Windows.

Just yesterday the Wi-Fi Alliance, who brought us...

Leo: The fabulous WEP.

Steve: Fabulous.

Leo: The fabulous WPS. The fab- what else have they ruined?

Steve: All these wonderfully secure technologies. And I shudder at this. I think, oh, goodness, have they not learned their lesson. Anyway, they're all excited about their next drop of update called Passpoint. We touched on this a few weeks ago. Not much was known. Yesterday they dropped the announcement that they will begin authenticating and verifying equipment to be Passpoint compliant. Also there's something called Hotpoint, I think it's Hotpoint [Hotspot] 2.0 that this is part of, starting in the summer, July of this year, hoping for a 2013 rollout.

Now, what this promises is sort of a way of offloading cellular bandwidth to the ubiquitous hotspots, free hotspots in a way that is less hassle for users. When you typically need to use a free WiFi hotspot, like Starbucks, for example, there's some sort of a login screen, and you've got to go yes, sort of agree to the terms and services there. It's never very seamless. I find that it's always in the way so that I just don't try. I don't use WiFi at Starbucks. I stay on cellular, and that's the point is that that hassle keeps people on cellular. And of course the carriers, the cellular carriers, would much rather you switched over away from their bandwidth, over to landline bandwidth that is less expensive and less burdensome for them.

So the concept is that there'll be some sort of new authentication cycle where your phone's unique ID, be it SIM card or hardwired 25,000-character ID that these things have, would be included in an authentication loop. And the spec for this is $200, so I haven't bit the bullet yet to buy it, as I did the WPS spec, because I had to tell everybody about exactly how it worked and how it was broken. There's a whitepaper also that is

free that I've not gone over yet because this hasn't happened yet.

Also, this is not going to be a small change. It is not going to happen overnight. The problem is we have a huge install base of hardware for hotspots which all need to get upgraded. This is not a simple change. This is probably, at the least, would be a firmware change. And it's not clear whether this is authenticating you and encrypting your connection, that is, is it WPS or WPA2? What is it?

Leo: And they don't say.

Steve: It's just all, I mean, they even say "Mobile authentication could use your SIM card." It's like, well, you know it's going to. Why do you say "could"? No, anyway, it's just - this Wi-Fi Alliance is the biggest example of committee design.

Leo: I think it's also more marketers than technologists.

Steve: Oh, absolutely, yeah.

Leo: Yeah. It's about marketing.

Steve: I mean, why else would I be paying $200 to have a specification, which you want people to have in order to implement something good? It's like, okay. They have - the marketers have salaries that need to get paid. So anyway, I just wanted to say that it's moving forward. We'll certainly keep an eye on it. It's going to be critical. I mean, at some point in the future we'll be talking about it seriously. But it's not going to happen overnight. No rollout until next year sometime. And even then, probably slowly. So we'll, again, see how it goes.

Now, yesterday Firefox fixed another memory leak. This is sort of a constant battle for them. They've moved beyond their own code, though, and they're now looking at the behavior of sloppy add-ons. The sandboxing technology that Firefox uses has the add-ons loading in what's called the "chrome privilege zone." So chrome's own UI interface is written in JavaScript. The add-ons run in there. And then Chrome and the add-ons together sort of reach out into the other domain sandboxes for containment.

And the problem is that, if pages are shut down that had resources allocated to them, they weren't always being freed. So on the Mozilla.org Bugzilla page, there's a bug 695480 which has just got resolved. And from that, in the summary the guy who fixed this said, "Add-on authors probably don't need to bother changing anything." Oh, I should mention they're very bullish about this. They regard what they managed to do, essentially fixing the problem without the add-ons needing to fix, as a real step forward.

They said, "Add-on authors probably don't need to bother changing anything unless they see breakage [from what we've done]. Breakage should be pretty rare, and the huge upside of avoided leaks will be worth it. It's a little early to be sure what effects this will have, but the amount of leaks we see on our test suite dropped by 80 percent. I expect that this change will also fix a majority of the add-on leaks we see without any effort on the part of the add-on authors."

So this is still not in the release chain yet. This is in the development chain. But I'm sure this will get pushed out soon because I have heard, I think it's 15 is now - they're out there somewhere. I think I'm on 13, if I remember right, because we moved - either I went from 11 to 12 or 12 to 13. Anyway, there's sort of a string of these major versions that Mozilla is putting out now. They're moving fast, and they really are updating it, which is a good thing.

I have major news about Buffer Bloat. Jim Getty, who was the individual credited with coining the term and who produced that neat video that we referred to months ago when we did the episode on Buffer Bloat, he blogged about a paper that was just released by two network researchers. There's a Dr. Kathie Nichols who submitted a paper which is going to be published this summer in the journal called the ACM Queue, Association for Computing Machinery. Her co-author was none other than Van Jacobson, who is known as the designer of TCP, so he knows something about these problems.

So Jim Getty said that they published an article which was entitled "Controlling Queue Delay." And essentially they have developed a new algorithm. The name is CoDel, and it's pronounced "coddle," unfortunately. But maybe not, maybe that works because you're "coddling" TCP to help it work. CoDel is - the reason people are excited about it is, I mean, it's like the Holy Grail of queue management. It's a novel no knobs, just works, handles variable bandwidth and roundtrip time, and simple adaptive queue management (AQM) algorithm.

To quote from the article that Kathie - it's Kathie, by the way, Kathie Nichols. I think I said "Katie" before. Kathie Nichols and Van Jacobson wrote, they said, "In the decade since, many researchers have made strides in adaptive queue management, but no one has produced an adaptive queue management algorithm that has the following characteristics," which theirs does: "It is parameterless. It has no knobs for operators, users, or implementers to adjust. It treats good queue and bad queue differently, that is, it keeps the delays low while permitting bursts of traffic. It controls delay, while insensitive to roundtrip delays, link rates [meaning wire speed], and traffic loads. It adapts to dynamically changing link rates with no negative impact on utilization."

And that's significant because one of the problems people have seen at home is with WiFi connections where the link rate over the air is dynamically scaling as a function of your connection quality. So that can confuse things a lot. If you're doing things, and you move around a little bit, even just a few inches, or turn, if your connection quality drops, your link rate changes, and you've already trained up your TCP connections for one link rate. Suddenly it changes. They don't adapt well. So this fixes that.

"It is simple and efficient. It can easily span the spectrum low-end, Linux-based access points and home routers up to high-end commercial router silicon. And CoDel's algorithm is not based on queue size, queue-size averages, queue-size thresholds, rate measurements, link utilization, drop rate or queue occupancy time." So it's not like time in queue, which some recent approaches have had.

So, I mean, this is very exciting. It sounds like they've solved the problem. There is an implementation for the CeroWRT build, which is an offshoot from the OpenWRT. Anyone who's interested can go to BufferBloat.net. That's the site where this work is being focused on. People are working with early releases of code to implement this. I expect we'll see rapid progress. And looking around at people who have had a chance to study this, I have not yet gone into detail. The paper is available, probably linked from BufferBloat.net, but you can get it from the ACM site as a web page or PDF. It's got pretty pictures and shows buffers and things. Once I figure out what it is, I will simplify it and tell everybody. But I'm excited because I'm seeing people saying this could solve the

problem. Like, okay...

Leo: This is a modification to TCP, to the protocol?

Steve: No. And that's what's so beautiful. This is a modification to the buffering, that is...

Leo: The buffering algorithm.

Steve: Yes, because we really can't change TCP. I mean, if we wanted to, oh, boy, talk about rolling out new access points being slow. TCP is in everything already. So the beauty is this would be essentially a change to routers. So firmware updates, or throw away your $49 blue plastic box and get another one, and suddenly your network will run a lot more robustly at home.

Leo: So we'll start looking for CoDel on the box, or AQM, which is really what this is, is queue management.

Steve: Well, but there are other AQMs. Like we talked about random early detection, where as the buffers began to get full, they would start throwing away, randomly, packets. And so the chances were that the hogs would statistically have more of their packets discarded than the non-hogs. And so, I mean, so much time has gone into this. But it really sounds as though it wasn't until it really, like the pressure got cranked up, that some really smart people who really understand the history of all of this problem sat down and said, okay, we've got to figure this out.

And I think CoDel does. From everything that they have written, they've got a solution. So it's going to be very cool. So, yeah, I would say we'll keep an eye on it, see what the marketers call this. I mean, CoDel is, okay, well, I wish we could have had a really cool name. But at least it's not awful.

Leo: This is great.

Steve: Yeah. I did want to mention that I have a new Twit handle, a new Twitter channel that I created in order to move my health stuff off of SGgrc. There have been a lot of people that have been tweeting since you and I did the special on Sunday, some asking where is it, where can they go to get it.

Leo: It's out now.

Steve: Yes.

Leo: Yes. And Part 2 is Sunday.

Steve: Yes. We're going to do a second half. Anyway, my new Twitter handle, for anybody who wants to follow that, is SGvlc. Instead of GRC, it's VLC for Very Low Carb. So SGvlc is where I will tweet stuff of interest to people who are interested in that so that I'm not clogging up my regular SGgrc with such traffic.

Leo: And you haven't tweeted there yet.

Steve: Nope, I haven't.

Leo: But you will. But of course people can go to GRC.com/health. There's a lot of information there. And then the special that we did on Sunday is available in our Specials feed. We didn't put in the Security Now! feed because it's not security. It's a special. So TWiT.tv/specials. It's the most recent one, Episode 124, "The Sugar Hill." And Part 2 - this one's going up the sugar hill, Part 2 going down the sugar hill - is next Sunday, 2:00 to 3:00 p.m. Pacific, 5:00 to 6:00 p.m. Eastern time, 2100 UTC on TWiT.tv, if you want to watch.

Steve: Yes. We squeezed it in between your Tech Guy radio program and your recording of your Sunday TWiT show. And so we'll do that again. As you mentioned, there are now a bunch of health-related pages. Well, I've already had, I've always had that Vitamin D page up, since the summer of '09 when we did the Vitamin D podcast. I'm now adding a bunch more with the references to the books that you and I have read and like and a bunch of other resources. So GRC.com/health, and everyone will be able to find those things there.

Leo: And I've been skimming this "Rosedale Diet" book you recommended, which I ordered as soon as you recommended it. It's fantastic. I'm very excited.

Steve: Yes, it really is. I did read it cover to cover, except I skipped the recipes because, although I looked...

Leo: You eat out. We're going to get you cooking, Steve.

Steve: Actually, I have sent a note to Jenny and said, you know, there may be a day when I'm cooking for us. And she wrote back, "Ha ha ha."

Leo: I think you should. Some of the fun of this, frankly, for me, has always been doing the cooking. And then you know what's in it.

Steve: Yes.

Leo: Then you know what oils are used and everything, and you can really make - yeah.

Steve: Exactly. Exactly.

Leo: They even have a healthy version of granola in here.

Steve: Yeah, and this guy gets it. He's been, I mean, what I like about Ron Rosedale's book is he's been involved in this for decades, and it's, like, applied. It's a practical manual. But I want all the science and all the biochemistry first. And in fact I was waving this book in front of the camera before we began recording, my organic and biological chemistry textbook. It's the only thing I could find that actually showed the actual chemical processes for doing some of the things that I was talking about last Sunday and will be going into more detail in next Sunday.

For me, I want to - I just don't want to take it for granted. The scientific method is verification and multiple sourcing and really, not just third-person hearsay, but how exactly does this stuff work? And I think that's, if I bring any value to these, that's what it is, it's that it's not the same thing you can find anywhere else.

Leo: It's fascinating stuff.

Steve: I do have an interesting SpinRite story that I ran across. We've never discussed this before. And it ought to get everyone's attention because it's something unusual from a Bob England, who wrote on April 23rd. He said, "Hi, Steve. I'm a long-time listener to Security Now! and long-term OP," which he says stands for "Other People's SpinRite User."

"I have to admit I've often borrowed other people's copies of SpinRite to keep my own hard drives working well. When I used to work as an IT support tech, I used to recommend SpinRite to all my clients. Some purchased it; some did not." Well, Bob, I'd say you've helped me out, so I appreciate that. "Over the past couple of weeks I noticed that my PC was beginning to just not feel right. I'm used to how my PC feels, during what part of certain actions I will notice hard drive activity and so on and so forth. Windows started to take longer to boot up. File transfers from one hard drive to another were getting slower, and apps were often not responding.

"Then, last Friday, Windows BSOD'd and refused to boot into anything other than recovery mode and could not find usable partitions. I thought enough was enough, and I decided it was time I actually got around to purchasing my own copy of SpinRite, so I did. I ran SpinRite and saw the estimated completion time was 1,028 hours!" He said, "I know that SpinRite can take a long time, but instead of just leaving it, I looked at the other screens to see if there was something important that I should take note of. Sure enough, in the stats screen I saw large numbers of cabling errors. The SATA cables I was using were the ones that came with my motherboard, so I thought they should be okay. I checked to make sure they were seated correctly, and they were.

"I purchased new SATA cables and reran SpinRite, and it did a complete scan of the drive in less than five hours. I wish I could say that I rebooted and everything was fine, but many of the Windows files were corrupted beyond repair" - and I'll explain why when I finish reading this letter - "mostly due to the BSOD I am getting" - the Blue Screen of Death. And it's actually not, but I'll explain. "However, SpinRite did manage to recover the files and sectors from the data drives, where I store all the work I do for our charity, of the 5TB of files that I have, including accounts and other extremely important

information. But as SpinRite had fixed these drives, it meant I did not have to wait to redownload all my backups or copy over archives from DVDs.

"I'm now setting up a NAS with RAID to further protect the data. I was wondering if you had experience of bad SATA cables, particularly those that come with original motherboards. Also, does the length of the cable matter? I ask this because I have some very short SATA cables and was wondering if the principles of signal propagation apply to these. Also, one other question: 'Locus of control.' Is that a typo, or does SpinRite have a personality? I've only heard the term used in psychology. I've always assumed it was supposed to be 'focus of control,' but now I'm not sure. Anyway, great show, and thanks for giving us SpinRite and all those other utilities. Bob England."

So this is interesting. I show checksum transfer errors and label them, what did I call them, "cabling errors," because that's all they can be in SpinRite. And it's uncommon, but you know me, I'm pretty thorough. And when I was designing this, I thought, well, there is a chance that, because it's a simple checksum algorithm, that errors could be occurring and going undetected. And that's what happened in this case. There is, in the actual hardware between the controller on the motherboard and the drive, there is a checksum-generating process that verifies the transfer of data. This is different than the ECC that the drive appends. It's different than any external checksums that are being applied. It's just essentially the cable. Did the data that left the controller get to the drive correctly?

And so what happens is, after a packet of data arrives at the drive, it verifies the checksum and flags an error if it's wrong. But sadly, nothing else in the world except SpinRite brings that to your attention. So what was happening was the drive was - this was all slowing down because many of these checksum errors were causing a retransmission over bad cables. Unfortunately, he had corruption because, again, it's a simple checksum. It's not like a good hash. That's just too much algorithmic overhead. And this stuff all dates back to 20 years before.

So it is possible to have undetected transmission errors between your motherboard and your drive. This is really bad because the drive thinks everything was fine and writes the wrong data onto your drive. And the motherboard thinks everything is fine because the drive didn't say, whoa, I didn't get that correctly, send it to me again.

So to answer Bob's question about length, I'm always - it is the case that shorter cables are theoretically better. Probably in this day and age of digitization of everything it's not as important. I mean, it would take a huge amount of noise, like lightning strikes, in order to glitch these cables. They're differential in nature where there's a plus and minus that are balanced. They're typically twisted so that any electromagnetic interference is received equally by both sides, so the difference doesn't change. That's what differential is all about.

What happened, doubtless, is just some noise in the actual physical connection. The fingers are typically gold on gold because you don't want to generate resistance noise at that junction. But these are not soldered. They're not, like, screwed down tight. They're a design which is meant to be a compromise between convenience and quality. And in this case that convenience compromise bit Bob. All he probably had to do was just wiggle the connectors, or pull them off and put them back on. That's something that Greg tells people to do all the time, just pull the connections off, reseat them. In the process that wipes the contacts across each other and removes the grit or dust or maybe a little bit of oxide that forms. And that's why we use gold, because it is resistant to oxidation in oxygen. So something we have never talked about before, but it's something that can bite you. And it bit Bob.

Leo: This is why you're the hard drive expert. I love it. Steve, questions, questions, questions.

Steve: Yeah, we've got some great stuff from our listeners.

Leo: We have a questing audience. They just want to know more, starting with Alan J. Doyle. I think this is a Twitter because he's @AllenDoyle, from Eagan, Minnesota. He says: Steve, if your backed-up data gets a virus, won't your cloud backup also be infected? Do the file-by-file cloud storage solutions have an advantage here?

Steve: You know, we've never talked about that question, which I think is a good one.

Leo: Yeah. Are you backing up the virus? Backing up the infection?

Steve: Yeah. When we've talked about it generically, we've talked about that, for example, if you have backups, and you're keeping track of them, then it's necessary, clearly, for example, if you're making images of your whole system, you want to go back to an image which is before your system got infected. Hopefully, if you're backing up just your data, documents and so forth, those tend to be more benign. But, I mean, it certainly raises a very good point, that if you are backing up contaminated files, or doing whole system backups, then Allen's point is right. You're backing up indiscriminately all of your files, and that means you're saving viruses and malware along with everything else.

So again, I mean, the only thing you can do is keep your defenses up, be aware of the danger, and maybe be a little bit better with, I mean, compensate for this fundamental problem by not discarding older instances of your system when it was in a known good state. The downside of restoring from something older is there'll be a lot more changes to your system since then. So you're resetting the clock back to that point. So the stuff you have done that you wish you could keep gets lost.

And it could end up being like sort of a hybrid, where you take a snapshot of your diseased system so that you at least have all the things that you've done, even though they include things you don't want to have, like malware. Then you restore from the most recent known uninfected image, which puts you back in time. Then you carefully bring the things from your infected image, file by file, back over, your documents and maybe your email, if you trust that, and so forth, in order to - but again, only the stuff you need. I mean, doing this does give you an opportunity to do some spring cleaning, too. I mean, every time I'm setting up a new system from scratch, and I'm sure you've had this, Leo, because you're as much of a "ooh, let me try and use that utility" as anyone.

Leo: Oh, constantly, yeah.

Steve: Yeah. And so setting up a new system is always a chance to say, okay, now I'm not going to install all that crazy stuff that I installed last time. Of course we just end up with new crazy stuff.

Leo: Yeah. Yeah. Well, I mean, in general, though, backups, if you're backing up data, data by - as we've said many times, you have to execute - a bad guy has to execute something. And where data gets you in trouble is where you have Adobe Reader or some other, remember that JPEG metafile issue on Windows? So the thing that reads the data file itself has a flaw, and then you could create - you can craft malware-laden data files that activate. But that requires you to have an application that's also got a hole in it. So data by itself is generally okay; right?

Steve: And that reminds me that, when you do restore from an older image...

Leo: Get the latest versions of your apps.

Steve: Well, and the updates, security updates.

Leo: Exactly.

Steve: You want to immediately bring your system current so that it's got its guard up to whatever degree possible.

Leo: I frequently, on the radio show, will talk about this. And I always say, use a genuine install disk. Install Windows. And the very first thing you do is run Windows Update until you can update no longer.

Steve: I know, over and over and over.

Leo: Over and over till there's no more updating to be done.

Steve: Updates of the updates of the updates.

Leo: And I also say don't install stuff if you don't know you need it because people always say, oh, I'm going to put all my 500 programs back on. There's a real opportunity not to do that; right?

Steve: Yup. Spring cleaning.

Leo: Spring cleaning. And things like Shockwave, every app potentially brings with it a problem. So don't install it unless you know you need it.

Steve: Yes.

Leo: Okay. Continuing on. Where did I put it? I closed - oh, shoot. I closed the questions. Let me re-open those questions. Thank goodness for open reset. Here we go. Question 2, Alvaro Stevenson in Monterrey, Mexico - I'm sure I'm butchering your name, and I apologize. How to tell if your home router's buffer is too big? The buffer bloat problem: Steve, I've joined an online game server with a 60ms ping, then I transferred a big file between two computers in my home network, and my ping increased to 200ms. After the transfer, back down to 60. Does this mean my home router's buffer can add up to 140ms of latency? Love the show, Alvaro Stevenson. What do you say?

Steve: Well, I would say yes and no. The real test that you need to use, and this test doesn't do that, is you need to use the buffer that you're sharing with what you want to test. And a number of people have asked, so I thought this was a very good point. When you're transferring between two machines on your home network, it's not going - that data you're transferring is not going out through - it may be passing through your router, but it's not going through the buffer out to the Internet. And that's the key because it's - remember that the whole glitch here is where you have a change in bandwidth. Nobody's upstream bandwidth is as high as their own Intranet's bandwidth. We're all running 100Mb, probably, inside our own home networks, but substantially less than that.

So it's where you transition from the high bandwidth to the low bandwidth. That creates a pinch point, and then the buffer is there in order to keep from just throwing away everything that won't be able to run at that lower speed. So to do the test, it's necessary - what people are doing, and successfully, would be to, like, check the ping time, then download something big from the Internet, like one of our podcasts. And while it's coming down, try to ping your game server. There you're going to see the real effect of the buffer bloat. And I'll bet it's way more than 140ms.

Leo: Or run Netalyzr.

Steve: Yup. That, too.

Leo: I mean, that's basically what the Netalyzr is doing, right, is...

Steve: Yeah, although I like the idea of, like...

Leo: You could test it, yeah.

Steve: I know that gamers are all hopped up about their ping time because they see that as the interactiveness of their online gaming. They want to not get penalized for latency in the connection. So this is a way of, like, doing a test that you're familiar with, which is game server ping time, and exacerbating that problem by simultaneously downloading something and seeing how bad it gets. And looks like we're going to have a solution for that one of these days.

Leo: Boy, I tell you, we've had great results. I'm now getting all the hosts to run Netalyzr just to see what's going on in their network, and it's been really useful.

Steve: Neat.

Leo: If you Google ICSI - it's at University of California at Berkeley's Computer Science Institute, ICSI - and Netalyzr is N-e-t-a-l-y-z-r. So I wish people wouldn't use kind of unconventional spellings because...

Steve: I wonder if they had to fit it into eight characters because they're still using DOS or something.

Leo: Ah, maybe that's why. Because if you just spelled it right, then we wouldn't have to spell it all the time. So Netalyzr. ICSI and Netalyzr.

Let's go to Cuyahoga Falls, Ohio. Susan Kennedy lives there, and she wants to know about TNO-level security password protection: Steve and Leo, long-time fan, Leo from The Screen Savers and TechTV, Steve from much earlier on GRC.com. So I was thrilled to find the Security Now! podcast. I can't catch the live feed, but I never miss a show. My question has to do with Trust No One, TNO-level password security. What if the encryption key for my cloud data resided on a corrupted section of my storage provider's disk? Would I lose all my data, even if I remembered my password correctly? Thanks for a great show. Susan.

Steve: Well, this is a nice question about maybe a bigger problem, which again we've never talked about. We sort of assume that nothing can ever go wrong with our data after we've sent it off somewhere.

Leo: Not so. Oh, is that not so.

Steve: And remember, one of the funniest things from "Hitchhiker's Guide to the Galaxy" was when Arthur Dent and Zaphod and their little crew landed in the middle of a rugby match, I think it was, like landed their space ship right in the middle of a rugby field with a match ongoing, and Arthur was really upset. He said, wait a minute, we just can't land here. And Zaphod said sure we can. And he said, we just turn on the SEP field. And Arthur said, what? And that's Somebody Else's Problem field, where...

Leo: I don't remember that. That's funny.

Steve: Oh, isn't that wonderful? Oh. And Arthur said, what are you talking about? And Zaphod said, well, go try to look at the ship. And Arthur explains it when he - his gaze just sort of slid off of it. He wasn't able to really lock onto it.

Leo: Someone else's problem.

Steve: And it's somebody else's problem. Oh.

Leo: That is funny.

Steve: So it was a field that just told your brain that, well, whatever that is, it's somebody else's problem.

Leo: Somebody else's.

Steve: And the problem is that we would like to believe that when we send our precious data off to the cloud, now it's no longer our problem, it's somebody else's problem. Problem is, they can drop the ball. They can have corruption and so forth. And I'm sure their license agreements and their stuff say we'll make best efforts. But...

Leo: You're trusting them.

Steve: All we can do is all we can do. And one of the reasons I like the larger guys, like Amazon, is they've got the clout to do redundant storage across multiple datacenters so that any two can go down, in Amazon's case, and you still have your data. Smaller guys are going to give you, again, the best job they can. But it's still important. So to answer Susan's question, yes, if there was corruption there, that's still a problem.

So my feeling is that that's part of your backup strategy, but you do not rely on it exclusively. External drives, network-attached storage, these things have become inexpensive enough that they're still a good thing to use, even if, or even though, you're still trying to make your data backup somebody else's problem. You're only safe if it's still a little bit your problem.

Leo: You know what happened to Carbonite some years ago, they're one of our sponsors, we talk about them all the time, they bought some Promise hard drive controllers. And of course they use RAID and redundancy and so forth. But they were small at the time, and I don't think they had multiple datacenters. And these controllers corrupted all the data. And I think they actually sued them. They sued Promise over this whole thing. Most people, I think, were able to get their data back because they discovered the problem. It was only people who didn't have - who had a crash in the period of time between when the problem happened and when they fixed it.

Steve: And you know what it was, Leo? I do know what it was. It was the very, very early days of SATA. The very early SATA drives and controllers weren't quite synchronized in the way they talked. And there were some early problems that came up. I remember Promise was having a problem with their SATA controllers.

Leo: It was very upsetting, I know, to David at Carbonite. He's the guy, the founder. And they have since made sure that that could never happen again. So in some ways, the good thing about Carbonite is it did happen. And that's always - it's the same thing with backup. You don't really take it seriously till you lose it.

Steve: Fool me once.

Leo: Yeah, exactly. So it does happen. We do trust them. I think that is certainly something to be aware of. I think it's a great question.

Tom Paladino in Commack, New York suggests some poor man's backup solutions: Thanks for the great podcast, for all the various cloud backup solutions on Security Now! 349. Want to point out there are a few services that can be used for free in what I call a "poor man's solution." SugarSync, he uses that for all his documents. And SugarSync, like a lot of these, offers the first few gigs free, in SugarSync's case 5GB. He says that's more than enough for simple document data. However, for most users a large portion of their data is comprised of music, images, and videos. That's a lot more than 5GB.

Fortunately, Google has three services that can provide free online storage for all of these, Google Picasa, will let you synch images to the cloud and now has unlimited file storage for images. I didn't know that. I pay for extra storage on Picasa, so that's good to know. Google Play, formerly Google Music, has a synch client that pushes all of your music to the cloud and allows you to download your entire library back to a local PC if you need to. That's free for, I think, 25,000 titles. And lastly, YouTube now allows HD video uploads longer than 15 minutes, allowing users to upload their important family videos online for free. It's important to note that both Picasa and YouTube can be set to "private" if you don't want to share those, as you probably wouldn't if they're family videos. Thanks for a great podcast. Tom Paladino, Commack, New York.

Steve: So I thought this was very clever. And it really does make sense. And I wanted your own experience and opinions, if you had any, with those services because I don't. But I like the idea of recognizing that, exactly as Tom said, the things that we create are documents and...

Leo: Are small.

Steve: ...are small relative to this explosion that we've had in the size of media, which everyone's having fun with. But, boy, is it big. And so rather than trying to have one solution where one class of your backup media is essentially forcing you into expensive data plans, Tom's idea is a great one. It's like, hey, if you've got free photo upload for Picasa, put them up there. And, that is, segregate this a little bit. It's a little more work. It's not everything in one place. But it makes a lot of sense.

Leo: I agree completely. I think you're making a mistake if you're backing up - you've talked about this - these large media files, like ripped DVDs, to a backup

service. That's nuts. Don't do it.

Steve: Yeah, it's crazy.

Leo: And as it turns out, when people say, oh, I have 500GB of backup, most of that is stuff you don't really need to make a copy of. I can't believe that you have 500GB of financial records or any of those kinds of things. There are a lot of services that offer free photo storage. Google Drive now is 5GB for free, and it's very cheap to buy more. I have 200GB on Google, and I use that for everything.

So, yeah, I think there are a lot of good solutions. I don't think you have to - it's really more of a question of how complicated you want to make it and how automatic it is and that kind of thing. Lots of good solutions. I mean, the cloud is coming, big-time. If you use Apple stuff, you might just use iCloud. It's not cheap. They seem to charge a lot. In fact, it's 50 bucks for 25GB a year and 100 bucks for 50GB a year, which is a little much, if you ask me.

Gregory MacGregor - a nice Scottish name. Guess where he is? He's in Spain. Gregory MacGregor wonders, why is ARM better than - no, wait a minute, that's not the right one. Did I jump ahead? Yes, it is. Why is ARM better than x86/x64 for mobile devices? Steve, it's not a security question, but since you're the geekiest person I know on Leo's network, I'd love to know if you could answer this. Ever since non-x86 tablets started showing up in the past few years, I've wondered, what it is that makes the ARM architecture the one to consider for mobile products in the future? I know they give great performance per watt, but I've never known if that's by design, or if it's just that x86 isn't efficient enough today.

Why am I asking this? Well, Windows 8 is just around the corner, and I wonder if it makes any sense to invest in an ARM tablet given the legacy hardware and software I can use with an x86 tablet. Is there a chance the x86/x64 architecture could eventually match ARM's mobile-friendly specs? Or do we really have to chose between mobility and legacy support? Tough choice. Thanks for the show. You rock.

Steve: So, okay. First of all, I would point Gregory at a really fun series that we did - you'll remember, Leo - about designing computers from first principles, where we started from scratch and looked at how computers work. And one of the things we talked about was CISC, that is, C-I-S-C, Complex Instruction Set Computers, versus RISC, Reduced Instruction Set Computers. And ARM, which stands for Advanced RISC Machine, although it used to stand for Acorn - I like the new name better - it is a RISC architecture.

And essentially the designers in England who put this together were essentially like a mom-and-pop shop. I mean, they were just some - they weren't, like, Intel or Fairchild or Texas Instruments or anything. They were just some guys who said, gee, you know, we've been using the 6502, and it's cheaper than the 8088 or the 8080, rather, because it's got, like, so many fewer gates. And so it's a smaller die, and it was easier to design. And so they just sort of said - and they weren't able to get a chip they wanted, so they designed one. And they designed a simple one. And they decided we're going to make it simple, and then we'll put the complexity in the software rather than in the hardware.

Well, Intel's path was different. Intel began with a complex sort of traditional mainframe instruction set in a small - with a small bit length, namely an 8-bit microprocessor, but

with an inherently complex instruction set. So it was very mainframe-ish. And then they extended its size, and it got increasingly complex as they went. And now they're to the point where they desperately wish that they could keep instruction set compatibility with a simpler design, but they can't. And you can imagine that the engineering effort that they put into trying to compete with this ARM chip, which just whizzes by them in terms of performance and power consumption.

So the answer is, I've sort of summarized what we discussed in great and I think really interesting detail back in those computer architecture podcasts. So Gregory and anybody else who hasn't heard those, I'd really recommend you go back and look at them. They're all at GRC.com/sn, where you can find them. There's a search there. Thanks to Elaine's transcripts, you can put in, like, "computer architecture," probably, and you'll find them. I don't remember what batch of weeks they were in that Leo and I did these.

But essentially what matters is transistor count. Transistors burn power. And due to the way ARM started and has, to their credit, remained, there are many fewer transistors needed to run their instructions than Intel needs to run theirs. And so the ARM dies are smaller. There are fewer transistors. They run more efficiently. Everything about it has just turned out, coincidentally, to be what made sense in a battery-powered device. And Intel, I don't know how they could get around this, how they could - if they could, they would have. And so I think that's just a fundamental legacy difference that Intel is stuck with.

Leo: It isn't actually by accident because ARM - Apple put a huge investment into ARM because they were developing processors for the Newton. And so it isn't actually by accident. It's intentional. ARMs were always intended for mobile devices, ARM processors. Intel has tried to do these scale processors. They're trying to do low-power processors. They even have some. They realize they're missing out on a huge growth area with mobile.

And interestingly, Windows, when they decided to put Windows 8 on tablets, they didn't make it x86. They rewrote it for ARM. It's Windows on ARM, or Windows RT. And I think that the legacy issue comes from the fact that Microsoft just kind of unilaterally decided that, if you're going to write software for the tablets, that it's got to be Metro style. And only Microsoft has its actual Office style stuff on tablets and desktop. So I don't - we'll have to ask Paul about that. My suspicion is in time you'll be able to do a lot of the legacy stuff.

Steve: It'll be interesting to see Windows on ARM. That is, it'll help us to understand...

Leo: Right, the difference.

Steve: Yes, exactly, what's the slow part? Is it the x86/64? Or is it the Windows-ness, which is like - does Android on an ARM tablet, how does its performance compare to Windows on the same hardware? We'll be able to see it.

Leo: Yeah. I don't know what the status is of Intel's XScale solution. I mean, they're really hoping, I'm sure, to get a mobile part out there. Be interesting. They're able to get the die sizes down so low now, with Ivy Bridge and so forth. I wonder.

Steve: Yeah. I don't think you can engineer around the architecture. What they're doing is they're, like, stalling, they're stopping the processor and trying not to have it run...

Leo: Right, speed step and all that, yeah.

Steve: Exactly. And so, but when it does run, it says give me a larger straw because I need to suck more juice out of your battery.

Leo: Actually, interesting, just looking up XScale, which is Intel's mobile part, it's based on the ARM architecture. So never mind. I think ARM won. Wow, that's really interesting. Who would have thought Intel would use ARM?

Moving along to Question 6, Bill Schwartz in Quincy, Mass. A huge SpiderOak disadvantage: I've confirmed with SpiderOak support, which was excellent, that they provide no means for maintaining privacy between different users, or different computers, on a single account. All users and all machines must share an encryption password. So if, say, two household members share a 100GB SpiderOak account, all privacy between them is annihilated for all files placed on the SpiderOak network. Moreover, each household member then absolutely depends on all the others to keep the common encryption key safe and private. One slip by one person compromises everyone's data. This will be a big disadvantage for many people. I don't think so, but - I mean, come on. Really?

SpiderOak told me their forthcoming enterprise solution of course will offer such intra-account privacy, but at a higher cost. They also told me this feature would be implemented on the non-enterprise side, as well, but not anytime soon. For now, the only answer is for each household member to have his or her own account, which means multiplying costs. And since most users will probably never need anywhere near the full 100GB on the minimum paid plan, this will be a very big price disadvantage for SpiderOak. It's too bad. Everything else about SpiderOak seems great to me so far.

This is not a problem at all with Jungle Disk, my current cloud provider. As an aside, I have been feeling that Jungle Disk has absolutely lost the magic it once had since Jungle Dave sold and then left the company. I've had many problems, including massive overfilling for months and months that took forever for them to solve. Their app is still not completely compatible with OS X Lion. I'd love to find an alternative to Jungle Disk. Thanks for the great review. Hope this helps.

I've been using ARQ, on your recommendation, Steve, and I think this is a good alternative.

Steve: Yeah, I do, too. On the Mac side.

Leo: Yeah, on the Mac side, yeah.

Steve: Oh, and it does sound like, because he's talking about OS X Lion, it sounds like he's also...

Leo: I mean, ARQ is an excellent alternative to Jungle Disk if you're on OS X.

Steve: Yup. Now, I'm with you, Leo. I wanted to bring it up and, like, highlight the issue because that's a good one, maybe. And our listeners' individual usage patterns...

Leo: You'll have to be the judge, yeah.

Steve: Yeah. If you had a crazy teenager who - like Henry - who is sharing your account and had to have all your keys, and you were really storing super-confidential stuff - you probably don't have that kind of data. But I could imagine, if I was depending upon a service like this for encrypting the keys to my kingdom, frankly, I'd just spring for a separate account and let other people use a different account than mine as opposed to intending to share it. But it's a good...

Leo: Or use TrueCrypt or something. Right?

Steve: Right, right. Exactly. Or use one of these, as we covered last week, use a hybrid solution where you separately encrypt your own stuff and drop it into the family shared folder, so you're safe in any event. Yeah.

Leo: Yeah. And they do offer free accounts of, what is it, 2GB. So maybe give your teenager the 2GB, and then you don't have to worry about it, get a 2GB account.

Moving right along here, which we will do as soon as I reopen - why do I keep closing this window? Question #7, Rick in Rhode Island. He thinks you're wrong, Steve. You're wrong about Java.

Steve: I have been.

Leo: I don't know the details of cloud software using Java, but your dismissal of Java so quickly seems misplaced. There are plenty of standalone, not browser-embedded applications built using Java, and I don't think they're any less secure than native applications. Well, that's true. It would be good if you could elaborate on how these cloud apps run, browser vs. standalone. One reason standalone apps are written in Java is portability, and it works very well in that respect.

Steve: So I completely agree with Rick that in that mode it makes sense. I did have a couple tweets from people who said, yes, Steve, I guess I can understand your position on Java. But I'm a Linux user, and we're not Windows or Mac, and so we're often the stepchild that doesn't get the attention that the big platforms do. When they're written in Java, because Java creates cross-platform compatibility, we're able to use those solutions. And I think that's a very good point. I mean, as opposed to not being able to use them.

My complaint was different than that, though. It was that I don't like the idea of requiring

a user to install Java just to run a backup client. If you're a commercial enterprise making money from people, selling them your client, sell them a native client for their platform. These things are not hard to write or create, I mean, that's why there's a billion of them is that it's not like it's rocket science. Everyone's creating them. But my feeling is, wow, I'd like to have one written for Windows that doesn't force me to be running Java. And we have seen problems with memory management in Java clients, taking up half a gig of memory. Other people report that they don't have that problem. But if you're writing - I guess my sense is, if you're creating a commercial application for a platform, it's just not difficult to develop for that platform. If it's freeware, then you kind of get what you don't pay for.

Leo: I would say the most widely used Java app out there right now is a game called Minecraft. And it's written...

Steve: Whose shirt I apparently wear.

Leo: You wear that shirt. You have a diamond block shirt. And Minecraft is run by millions and millions of people, and you have to have Java to run it. Actually, come to think of it, the Citrix products also run in Java. So, but his point is well taken, which is a Java application is no less or more, actually may be more secure than a standalone application because of some of the built-in sandboxing features of Java.

Steve: Yup. And like I said, when I saw that Netalyzr...

Leo: Netalyzr's in Java, baby.

Steve: ...is in Java, I thought, oh, if you can do that in Java, then I need to keep that on my radar because that could be very useful for solving my own lack of cross-platform software development.

Leo: Java's awesome. I think your point, and what you've always said is, if you don't need it, uninstall it. There's no reason to have it. Like Shockwave, if you don't need it, don't install it because of the security issues.

Steve: Right. Because your browser will run Java things, and that's how 600,000 Mac people got infected.

Leo: Right. I think, I bet, I'm just going to - I'm pretty sure, I know you can say disable JavaScript. I'm pretty sure that the browser will also let you disable Java. So that would be the other thing, to go into...

Steve: Well, remember, the new update from Apple preemptively disables Java.

Leo: Right.

Steve: It shuts it down and then, if you turn it on, and you don't keep using it, shuts it down again.

Leo: I have on my Safari, on my OS X system, "Enable Java" unchecked. So you can have Java on your system. Just don't let the browser invoke it.

Steve: Right.

Leo: That would be good, too; right?

Steve: Yup, really good. And that's what Apple has done, is they are turning it off for you and keeping it off.

Leo: Yeah. And what Chrome does is they say each time, they ask you, do you want to allow this to run?

Steve: Right.

Leo: I see that all the time because whenever I run - whenever I do GoToMeeting, one of the reasons GoToMeeting I think uses Java is because it's compact, and they download a new copy every single time to make sure you have the latest version. And so I see that "allow" thing every time I run GoToMeeting, and it's fine. Just say, yeah, yeah, I wanted to run GoToMeeting, no problem.

Charles Hill, Washington, DC: A defense of Backblaze. Steve, I'm listening to your retelling of your email discussion with the CEO of Backblaze. I can't help but think you were missing a critical point he was trying to make. I am referring to their "user-provided private key" method only. You're saying that since the key is given to Backblaze, and they decrypt the user's data on their server, it isn't TNO because you have to trust Backblaze in that instance. Well, I don't disagree with that.

But what the BB CEO, the Backblaze CEO is saying is also valid: "There is no such thing as TNO with an integrated solution, so what's the issue?" That is, when you say if they were to do the en- or decryption on the user PC, it would be TNO because only the user would have the key in plaintext. His point - I think you might have missed it - is, if you're using software provided by the client, whether it's Backblaze, SpiderOak, or something else, you're trusting their client software is doing only what it claims to do. In other words, not sending back the data secretly or storing it somehow. All Backblaze is doing is shifting the point of trust from the client software to the server side. The issue of trusting the third-party exists, regardless.

So your suggestion of changing their architecture to just decrypt on the client isn't a solution. You still have to trust their client software. The CEO of Backblaze's point

was just that. You're just kidding yourself with TNO with an integrated service, so why not just be honest about it and trust them? The only real TNO solution that I've been able to come up with is to separate the encryption out from the storage. Personally, I just have a cron job tar/gz up my new files every week, run it through GnuPG to encrypt it, then FTP it up to an Amazon S3 account. Signatures are kept locally. It's a bit geeky, but it works for me. Or am I totally off base?

Steve: Well, I certainly see his point. My concern, and I know the concern of listeners who want TNO, is that it's not TNO. And their documentation makes it sound like it is, but it isn't. And we do know, because it comes up all the time, that our government is able to subpoena records of companies. In fact, I forgot to put it in the news this week, but I tweeted it, it was late last week, Leo, you may have seen it, the story about the FBI proposing legislation to force...

Leo: Oh, yeah, CISPA, yeah.

Steve: Yes, force social networking and other websites to build in monitoring technology. I mean, so this is not made up. And Gleb Budman, whom I exchange email with, the CEO and cofounder of Backblaze, who is a really nice guy, he said, "We've never been served with that kind of requirement." Well...

Leo: Boy, that's a shock.

Steve: If they were, they couldn't say. I mean, he would have to deny it. He would have to say that. And I just - the point is the architecture could be secure. Other people do it. These people didn't. And they really didn't say that. I mean, they said the other. And so that's my concern. It's like, yeah, this works. I'm sure it's fine. But it's not Trust No One because they've got the keys. And by definition, that just doesn't do it.

Leo: But his point is well taken, that when you run somebody's client software on your system, that software could be malcrafted in such a way that...

Steve: Absolutely.

Leo: ...it could be stealing data. You don't know that.

Steve: And I actually do like, I mean, he made the distinction of an integrated solution. Most people, 99.99 percent, want that. But that's where I like the solution I'm using of using a grandfathered Jungle Disk account and S3. Amazon is my storage provider. Jungle Disk is my encryption provider. They're not in cahoots any way. They're separable actions. And of course his, as he says, "geeky solution" is that to an extreme, where he's got a cron job that runs a script that tars and gzips and encrypts and FTPs, blah blah, but it works. There's absolute security because he only has to trust himself. He knows all these little pieces, there's no way that they can be collaborating.

Leo: Moving along to Question 9 from Shawn. Shawnt. Shant H. in Glendale, California. Does the government have a master encryption key? Well, this fits right into that wiretap story. How's it going, Steve? I'm a computer science student, long-time listener to the show. This week, in one of my classes, the professor announced to the class that all 128-bit encryption can be cracked by the U.S. government because they have a "master key." Wow.

Steve: Oh, I know.

Leo: What school is this?

Steve: I know. Glendale. It's not like it's out in the sticks somewhere.

Leo: Hmm. Now, being an avid listener to the show, I am 99.99999 percent sure that this is untrue. As far as my understanding goes, the math only allows a single unique key to decrypt the message, therefore there can't be a master key to decrypt the message. I think my argument in class will be weighted with your backing. Thanks for the great show. Shant.

Steve: Well, Shant, I don't want you to get in trouble with your professor by calling him an idiot.

Leo: He's an idiot.

Steve: He's an idiot. I mean...

Leo: And I wouldn't trust anything else he says after this

Steve: Yeah, that's a real worry. Okay. So the reason Leo and I can be so sure, I mean, I love the fact that we can be so sure. It's because all of this is open. See, this is the beauty of the open development, open academic process that encryption always enjoys is we know exactly how AES works. We've done a podcast on it. We've looked at it. It's been pounded on. It's the result of a competition among a bunch of all really good strong crypto and for a number of reasons sort of a compromise between how much power it used and how complex it was and whether it was, you know, all these criteria that they used, it was chosen. So I just - I love the certainty with which we're able to say no, there's no such, I mean, it's ridiculous, patently ridiculous because - and we're only able to say that because we absolutely know. It has not always been this way. You'll remember, Leo, the days of the Clipper chip, which was sort of a...

Leo: Al Gore, baby.

Steve: ...government-sponsored, secret crypto that no one knew what was inside. It was

like, oh...

Leo: And nobody used because there was a backdoor.

Steve: Exactly. So maybe this professor is confused, although the Clipper chip was not 128-bit encryption because it was too long ago for 128-bit to be used commonly. Probably 64, back in the DES era.

Leo: Or 48 even, maybe.

Steve: Maybe, yeah. But, I mean, oh, goodness, no. The beauty is contemporary encryption is completely open. It's what I love about it. And that's why, when an encryption technology, an encryption company says we're doing this, this, this, this, and this. Here's our diagram. This is what we're doing. I mean, we know what the blocks do. We know when you connect them together following standard practices that you're going to get a strong, a provably secure result as far as anyone knows. And we have to throw that caveat in because something could catch us out. But as well as we know, this is what it's going to do and how it's going to work. I love that aspect of crypto.

Leo: Me, too.

Steve: It makes for great podcasts.

Leo: Me, too. And it's why I like open source and recommend open source crypto solutions above all others, because you can verify that it works, that it doesn't have any backdoors.

Steve: Yeah. And I would say open algorithm, to take a step back...

Leo: Open algorithm, I agree, yes.

Steve: Yes. Because most people may not look at the bits. And mistakes can be made in algorithms. But again...

Leo: As we know, with WEP.

Steve: Yes, exactly.

Leo: I just hope this guy's a professor of philosophy, not computer science. That's all I can say.

Steve: Yeah.

Leo: I just hope he's not a computer science professor because that would be really depressing.

Steve: Yeah, well, so Shant...

Leo: Clipper chip was 80-bit, says "Considerate" in the chatroom.

Steve: Ah. Shant, tell your classmates the truth, but don't get in trouble. I don't want to, you know...

Leo: Yeah. You could pass notes. Last question, Tom Hartnett, St. Louis, MO, commenting about the forthcoming NSA facility in Utah. Get ready for this: I've heard you discuss before that there are no shortcuts possessed by the government for cracking encryption. I agree. However, I thought I'd pass along this article nonetheless. A paragraph in it caught my eye: "The $2 billion facility, slated to be complete by September 2013, is allegedly designed to be able to filter through yottabytes (that's 10^24 bytes) of data. Put into perspective, that's greater than the estimated total of all human knowledge since the dawn of mankind. If leaked information about the complex is correct, nothing will be safe from the facility's reach, from cell phone communications to emails to what you just bought with your credit card. And encryption won't protect you - one of the facility's priorities is breaking even the most complex of codes." Hmm.

Steve: Oh, Leo, that is my favorite voice you have ever done. Oh. Say goodbye to Australian. That was...

Leo: That's my Walter Cronkite.

Steve: Oh, my. Oh, we're going to bring that out from time to time. Oh. I just - oh, goodness. That's one for the record books.

Leo: So what do you [laughing]. Thank you, Steve, thank you. What do you say to this?

Steve: I just - I want to hear it again.

Leo: "The $2 billion facility."

Steve: Okay. I completely agree with the designed to be able to filter through yottabytes of data. Absolutely. We know that that's the main focus. And the fact is, so much today is not encrypted that huge amounts of value can be found in the clear, in plaintext, in all

these yottabytes. They're going to have to have some amazing indexing and organization system to deal with all this. But that's what they've got this big facility for.

The idea, though, that there's some magic secret to decryption, you know, it says, "And encryption won't protect you - one of the facility's priorities is breaking even the most complex codes." Well, I agree that that's what its priority is. I mean, it's...

Leo: It doesn't mean they can do it.

Steve: No, it's the dream of the NSA and the FBI and the CIA to be able to cut through encryption like butter so that it doesn't impede their investigations. And I don't...

Leo: That's why they're asking for a backdoor, by the way, in Skype and everything else, because they can't.

Steve: Right. Right. We've talked about TrueCrypted drives being sent from South America up to the U.S. law enforcement to see if the FBI could crack it, and unfortunately they were - I mean, okay. Whether it's unfortunate or not, I can't comment. But they weren't able to crack it because the crypto holds. The crypto is absolute at this point. So, yes. I mean, the only danger on the far horizon is potentially the myth of, the allure of quantum computing, where the theory is it simultaneously tries all keys at once, which just melts crypto. It's just over. But we're a long way away from that. So I'm not worried. And we ought not be. I'm not worried about Utah. I mean, basically they want to feed every possible communication channel that they can find through the equivalent of - remember the wonderful movie, "Colossus: The Forbin Project," which actually is being remade, by the way?

Leo: I love that movie.

Steve: I've watched it many times.

Leo: Oh, what a great movie. I'm glad it's being remade. It could stand to be a little updated.

Steve: Yeah, it really could. So it'd be great...

Leo: I'm not going to say anything. But somebody I know very well was approached by the federal government to help write something that would go through data and find keywords. And in fact we know pretty well that Echelon exists. The U.S. government's never admitted to it, but the British have. And all Echelon is, is exactly this. And this is what the NSA does. They monitor electronic transmissions. Unencrypted electronic transmissions. I'm sure they store encrypted transmissions, but I don't know if they can do anything with them. If they're properly encrypted, they can't.

Steve: Oh, and the roof looks so cool, with all the...

Leo: I haven't seen the roof. Antennas?

Steve: Oh, with all those big dishes, with all those dish antennas pointing in every direction. It's like, whoa.

Leo: But if anything, this should argue for people using encryption, using strong encryption because, if you don't, then you know that everything you say and do electronically, including email, telephone, cell phone, everything is being monitored by the NSA. And they do key - I don't know what the keywords are, but I'm sure if you said a few choice phrases about Uncle Sam and explosives, you'd probably get flagged. And that's what they do. That's what they're doing. There's so much encrypted transmission now that they probably wouldn't even have time to decrypt it.

Steve: Well, and there've got to be taps on major backbones of the Internet.

Leo: Oh, yes.

Steve: So in terms of communications, so like everything that crosses through anywhere that they're able to install a tap, they've got, I mean, I feel sorry for their yottabyte computers, all the garbage, I mean, all the nonsense.

Leo: Think of all that crap they're getting.

Steve: All the tweets and the Facebook postings, and it's like, oh, you wrote on my wall, and oh, it's like, okay, well, good luck with that, NSA.

Leo: This is a picture of the actual building in Utah.

Steve: Ooh, look at that lighting. Nice lighting.

Leo: The country's biggest spy center. I can understand why people are nervous. If you want to protect your privacy, what you should be doing is make sure that you let your member of Congress know that the FBI, NSA, CIA request to put backdoors into things like Skype and Facebook be denied.

Steve: Yup.

Leo: Because that's, really, that's what they're saying is, look, we can't decrypt

Skype, so we would like you to put a backdoor in it so that we can wiretap it.

Steve: It's like the problem that RIM had. We talked about RIM's problem over in the Far East often last year because there was such a brouhaha about the foreign governments demanding that their BlackBerry-using citizens could be spied on.

Leo: Yeah. By the way...

Steve: The technology just isn't there.

Leo: ...the contractors building this building have to have top-secret security clearances.

Steve: Yeah, because they've got special tile. Oh, good, goodness.

Leo: Flowing through its servers, according to Wired's Threat Level blog, which we love, will be all forms of communication, including the complete contents of private emails, cell phone calls, Google searches, parking receipts, travel itineraries, bookstore purchases.

Steve: Yeah, just everything, basically. We're all electronic now. Everything's wired up. And these suckers want to filter through all of that nonsense. I say good. I say let 'em have it.

Leo: You may remember that this was something George Bush wanted. It was called Total - actually, I think it was actually the Vice President. He wanted, remember, Total - what was it called? Total Information Awareness? TIA? In a fight against terrorism. The problem is, when your federal government knows everything, why limit it to terrorism? Let's go after people with unpaid parking tickets, everything else.

Steve: And unfortunately there is a history of that, too. There is a history of this stuff being repurposed, people believe for the health of the country, but not everyone is in agreement about what that means. And that's a problem.

Leo: Now, let me read you this paragraph because I'm sure this is where this story comes from. One senior intelligence official who until recently was involved with the program says that the "Bluffdale center will have another important and far more secret role." It's critical, he says, "for breaking codes. And code-breaking is crucial because much of the data that the center will handle - financial information, stock transactions, business deals, foreign military and diplomatic secrets, legal documents, confidential personal communications" - I guess not so confidential - "will be heavily encrypted. According to another top official also involved with the

program" - and this is what I think is bogus - "the NSA made an enormous breakthrough several years ago in its ability to cryptanalyze, or break, unfathomably complex encryption systems employed not only by governments around the world, but also many average computers in the U.S."

Steve: Well, I don't know.

Leo: Do you think so? Think it's credible?

Steve: We have no way of knowing.

Leo: We can't know.

Steve: The one thing that I remember from the early reports was that they already have a huge amount of data encrypted using older, weaker codes - for example, 64-bit encryption - and now we have the technology to feasibly crack that. So they've got communications from foreign powers encrypted in - I mean, old communications encrypted in the then strongest codes of the time. So what we need to remember is, when storage is available, the encryption we use needs to be strong relative to our ability to decrypt into the future until a point where it no longer matters. And so my best guess is that they've got way, I mean, 64-bit encryption is, you know, we pooh-pooh it, but it was strong then. It's still strong now. We're just staying way ahead of what's feasible by going to 128 and 256, which is, you know, 128 is already really, I mean, that's, like, plenty strong.

Leo: Yeah. In fact, he says, "a lot of foreign government stuff we've never been able to break is 128 or less. Break all that and you'll find out a lot more of what you didn't know - stuff we've already stored...."

Steve: Exactly. So my guess is they're rubbing their hands together about bringing this processing power to bear on stuff that's a decade or two old. There can be really juicy tidbits that still matter in data that is only that old.

Leo: Sure. They've stored it. They just couldn't crack it until now.

Steve: Right, right. I think the stuff we're doing today is probably safe, given everything we know.

Leo: And the guy who wrote this article [James Bamford] is the author of "The Shadow Factory: The Ultra-Secret NSA from 9/11 to the Eavesdropping on America," if you want to read more. He has a little bit of an ax to grind. I mean, he wants everybody to get scared. I think, though, if there's anything, this would argue for using encryption more, not less, because they are watching.

Steve: Yeah. Exactly. Not assuming privacy. Unfortunately, we can no longer...

Leo: We don't have it.

Steve: Yeah.

Leo: Yeah, we don't have it. If you don't mind if spooks in Utah are reading your mail, no big deal. But if you do, I would say public key cryptography with long...

Steve: 2048-bit public key and a 256-bit symmetric key. That's going to be...

Leo: That's going to be fine for a while.

Steve: It really is.

Leo: I hope.

Steve: I guess...

Leo: Unless you and I are being paid by the federal government to say that.

Steve: The only breakthrough I could see, Leo, would be if they actually had a factoring breakthrough.

Leo: Right.

Steve: That would be...

Leo: It would be a mathematical - it would be a breakthrough in mathematical theory, I think.

Steve: Yeah, well, see, and the thing I like about our symmetric crypto is it is so simple. I mean, we did a beautiful podcast on AES where I explained in detail what that algorithm is. And it's just - it's like there's nowhere for bad guys to hide in that algorithm. It's just so clear and clean. And we all know what the vulnerability, such as it is, of current public key crypto is. It's the factoring problem, which the smartest people in the country who are in the private sector have looked at, private and education, and have not been able to crack. Now, maybe the NSA has cracked factoring. And if they've cracked factoring, then, yeah, well, public key crypto, at least the standard RSA style, there are other types, but that's then gone. But again, maybe.

Leo: Maybe. It's an interesting idea.

Steve: Yeah, I mean, that's the vulnerability. Factoring is the vulnerability because that's what we all depend on right now. I mean, that's the Achilles heel. Not the symmetric crypto, but the asymmetric crypto. And the reason those keys, the asymmetric crypto keys have to be 1024 or 2048 bits long is that the actual strength is not nearly as great as it is with symmetric crypto, where a 128-bit key is fine. We need to have, like, 10 times that many bits to get the equivalent strength. So that would be the Achilles heel. And maybe somebody's with his headphones on, listening to us say this right now, Leo, going, oh, shoot. Figured it out.

Leo: Steve Gibson is at GRC.com. That's where he sells SpinRite, the world's best hard drive maintenance and recovery utility. Go there. Buy SpinRite. If you want to leave a question for the next feedback episode, GRC.com/feedback. His health page is there, GRC.com/health. His free stuff, all sorts of stuff. Really, it's becoming a better, bigger resource all the time for great information. Steve will be back on Sunday, 3:00 to 4:00 p.m. Pacific, 6:00 to 5:00 p.m.

Steve: Whoops, 2:00 to 3:00 p.m.

Leo: 2:00 to 3:00 p.m. Pacific 5:00 to 6:00 p.m. Eastern, Part 2 of our up and down the sugar hill, conquering the sugar hill series. I'm looking forward to concluding that. I hope we'll conclude. Maybe not. Maybe this is becoming an ongoing series. I don't know.

Steve: Well, the people who like it really love it, but...

Leo: People who hate it, don't.

Steve: Yeah.

Leo: And there's some in each camp. I'm loving it. And I have to say, now that I've got this book, the Rosedale book, I'm very excited.

Steve: Good.

Leo: Steve, we'll talk to you Sunday. And then, of course, again next week when we do a Security Now! every Wednesday, 11:00 a.m. Pacific, 2:00 p.m. Eastern time, 1800 UTC. You can watch live. If you can't see it live, don't worry, we make audio and video available always after the fact at TWiT.tv. Steve has 16Kb versions available at GRC.com, as well as transcriptions. So you can consume it in a variety of ways. 352 episodes now in the can. Thank you, Steve Gibson. We'll see you next time.

Steve: Thanks, Leo.

Leo: On Security Now!.