Listener Feedback Q&A #37

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-136.mp3
Quarter size (16 kbps) mp3 audio file URL: http://media.GRC.com/sn/sn-136-lq.mp3

INTRO: Netcasts you love, from people you trust. This is TWiT.

Leo Laporte: Bandwidth for Security Now! is provided by AOL Radio at AOL.com/podcasting.

This is Security Now! with Steve Gibson, Episode 136 for March 20, 2008: Listener Feedback #37. This show and the entire TWiT broadcast network is brought to you by donations from listeners like you. Thanks.

It's time for Security Now!. We're going to talk about security for the next hour or so with Steve Gibson, the man who coined the phrase "spyware." He is a security wizard. He's written so many great security programs; taken Microsoft to task for security flaws, chiefly raw sockets in their operating system. And every week we talk about the latest in security. Hi, Steve.

Steve Gibson: Hey, Leo. Great to be back with you once again.

Leo: Yeah. We're going to do a Q&A segment.

Steve: It's only been 136 weeks we've been doing this.

Leo: 136. Well, you know, you're tied with TWiT now.

Steve: Cool.

Leo: So TWiT just has to take a week off, and you pull ahead.

Steve: Okay.

Leo: We're at Episode - I think we just did 136 of This Week in Tech.

Steve: Well, you're going to be taking a trip, I think, out of the country here.

Leo: Oh, you might pull ahead, you're right. I'm going to Australia. You're going to pull ahead when I go to Australia because you, unlike those TWiTs, you like to record ahead and make sure that we don't miss a week.

Steve: We will not. And I'm going to say, as I'm reading the feedback that we get from our listeners, which I really so much appreciate, so many of them talk about like this is - I don't want to say it's the high point of their week. I'm sure they have lives outside...

Leo: I would hope so.

Steve: ...outside of Security Now! and podcasting. But at least within their podcasting domain they write and talk about how they look forward to Security Now! every week and wondering what's going to happen. So I loved one, one I read yesterday when I was preparing the Q&A for today. The guy said, you know, the problem I have with Security Now! - I'm thinking, okay, what's coming?

Leo: Yeah?

Steve: He says, most of the podcasts I listen to are sort of, you know, in the background, jab jab jab jabbing. He says, and then sometimes something that they say will catch my interest, and I'll back up a little bit and then listen. He says, I can't have Security Now! running in the background.

Leo: Oh, good.

Steve: Because he says, sometimes I have to listen to it multiple times to get everything that, you know, all of the content. It's like, well, okay, yeah, good.

Leo: That's okay with us. That's what the transcripts are for, too, of course.

Steve: Right.

Leo: They help a lot.

Steve: You can read along.

Leo: Yeah. Well, Q&A time is coming up, #37. But before we do that, anything you'd like to recap or cover from the last few weeks?

Steve: Two quickies. First of all, Greg, my tech support guy, asked me to mention on the podcast that GRC has not been infected by viruses or trojans.

Leo: Well, that's good to know.

Steve: Isn't that good to know. Just a little public relations mention. Apparently Avast, the AV scanner, has been warning people for the last week that GRC is infected with trojans and spyware. I don't know if this is the Google Analytics stuff that we talked about or what the cause is. But they fixed it. So I wanted to let anybody who's been worried that we got taken over or something, I wanted them to know that an update of their Avast viral signatures no longer reports, misreports a false positive that GRC has been compromised.

Leo: I think a lot of antiviruses now probably check a website or keep an eye on a website since that seems to be the number one vector for attacks nowadays.

Steve: It absolutely is. And it certainly makes sense for them to do that. It'd be nice if they weren't false-positiving.

Leo: Yeah, I mean, you know, it's funny because that's happened for years in applications. We've never had to deal with it as web people. But now I guess, yeah, maybe it's seeing the JavaScript for analytics.

Steve: Well, and in order to try to be preemptive, increasingly these AV tools are having to be heuristic. They're having to apply sort of basically sort of rules of thumb instead of finding a specific pattern. And unfortunately, well, and in fact some of the tools, some of the freeware that I've created has caused false positives because I've had to do things deliberately that were sort of like the insecure thing I was trying to fix or deal with. And so there was code that I had that was deliberately doing the same thing. Like a perfect example is that the Windows Metafile problem, we were causing some false positives because I was doing Windows Metafile stuff to test for the vulnerability which other malware could take advantage of. And so it's sort of the nature of the...

Leo: Well, that's why they often say don't run multiple antiviruses because of course

sometimes the signatures from one will seem to be a virus to the other.

Steve: Exactly. Perfect example. And the second thing I have under the subject of "I love our listeners."

Leo: We do.

Steve: A listener named Mark Odell sent feedback that I just, I mean, just cracked me up. His subject was, "This sounds like something" - it says, "It sounds like you could use one of these." And so he quotes me saying on Episode 134, which is two weeks ago, where I'm talking about going to get my car serviced, and I have a little tiny cute little 4GB thumb drive on my keychain. And I was talking about how it's a good thing I felt comfortable that I've got it TrueCrypted because I was handing my keys over. And I was just imagining, in general keys are a problem. You don't want to give them to your valet if you can avoid it because your car registration is in the glove compartment, they could figure out where you live. If they wanted to duplicate your key, it's a way in. And so I was thinking, but still, just how tantalizing a little thumb drive would be for someone. Anyway, so he...

Leo: That's a whole new thing nobody's probably thought about. But if you have a thumb drive or a dongle on your keychain, there's a security problem with handing it over.

Steve: Oh, absolutely.

Leo: I never thought about that. Very good point. I have my SanDisk on there.

Steve: So, yeah, exactly. So he quotes me saying this. And then - and this is what I love - he gives me a link to a Google search for the query "detachable key ring."

Leo: By the way, that's the new snotty reply email. You just attach a Google search to the response. Hey, I got an idea for you, look this up.

Steve: What a concept. Whoever heard of such a thing. A detachable key ring where I could have two parts, and I could just remove my - actually it would be handy not to have my keys dangling from my laptop when I plug this thing into my computer, too.

Leo: Good point.

Steve: So that's not a bad idea.

Leo: I want to Google that myself. I need that, too.

Steve: Lord knows how many hits you get. I didn't even click the link. But I did get a big kick out of that.

Leo: That's a very funny response. That's great.

Steve: So I appreciate that, Mark, and I just wanted to let you know.

Leo: Good. Very good. Anything else before we get to our...

Steve: Well, I do have a SpinRite story that was so well written, and pretty funny, actually. This is from an owner, Westcott Hyde. Okay. He says, "Hey, Steve and Leo, sorry this is long, but you have to read this! First, thanks. I just had to let you know I finally caught up to you. That is, it took six months and a day, and I have finally heard all the Security Now! episodes, some of them six or seven times, including last week's IronKey. Whew, I did it. Where is my Security Now! degree?"

Leo: Yeah, you need a button or something. "I Survived Security Now!."

Steve: He says, "Now for SpinRite, a quick tale. I heard you mention a record for SpinRite being a three-month recovery. But the record's anecdote was missing [SN-102]. Here is mine, since I have a similar record. An IT buddy came to me, asking if I knew anything about recovering hard drives. I had already purchased a copy of SpinRite sometime before this request, which meant I knew a little more than he did, I suppose, but not much. I had purchased it using the 'just in case I need it' rationale."

Leo: Fair enough.

Steve: "And to support the Gibson team."

Leo: Thank you.

Steve: "Well, I hesitated accepting this mission knowing that this is a breach of etiquette, since SpinRite was purchased for me, by me. But I acquiesced and told him I would help him out if he could convince his company in the meantime to buy a copy of SpinRite in the spirit of good intentions, not knowing it would take so long to achieve results. Read on.

"Long and short of it, this drive had extremely valuable CRM, customer relations data, that was running on a platform that was running on VMware, and the files were buried in VMware's structure. Not an easy prospect. No backup snapshot, or other backup of any kind, for that matter. SpinRite gave the Red Screen Of Death warning notice that the

drive was about to imminently fail and to recover data before running SpinRite."

That's actually something that SpinRite does. It does an initial check of the SMART subsystem. And if SMART is already saying, oh, you know, this is about to die, SpinRite will bring up a red screen and say, okay, now, look. We'll go with this for you. But the drive is already saying it's about belly-up. So maybe you should pull off whatever you can before going for hopefully complete recovery with SpinRite because SpinRite has been known to kill a drive. If it's already right on the edge, SpinRite just in doing its data recovery, just regular reads and writes to the drive, can push it over the line.

Leo: Just the activity.

Steve: And SpinRite is constantly checking the SMART data and will stop if the drive says, oh, wait a minute, I don't know how much more time I've got here. So anyway, but apparently, due to the fact that this was in a system buried in a file system buried in a VMware image that was virtually unrecoverable, there's nothing they could do, and they had to have this data. So he says:

"Well, the drive, even though it was surge protected until the cows came home, was a victim of a lightning strike near the building, or so I was told."

Leo: Wow.

Steve: Struck by lightning. "So I figured, what the heck, it is beyond the RSOD anyway. Let's see what we can get. I only needed the precious CRM data. So a couple of days go by with SpinRite running. I check. No, it won't boot, and no file structure can yet be seen on another Linux Fedora box by simply mounting the original Fedora Core 6 drive. SpinRite keeps going and going and going like the Energizer bunny, with 'unrecovered red sector' after 'unrecovered red sector' error. The first six weeks go by. Now...."

Leo: I can't believe he just leaves it running. This is what cracks me up.

Steve: "Now I could see the file structure, but could still not get any data. SpinRite wasn't done, however. I was just impatient to get the data. Three months later the drive is still spinning. It is simply hopeless, I thought. But there was no failure of the actuator or the drive spindle. It never overheated, and SpinRite was still cranking away. By this time I had learned to tune out the drone of the drive spinning in the background as just something my subconscious learned to accept, and not as a reminder to me that valuable data was actively being recovered. I really thought that the software (which I had never had the occasion to use before) was just hype.

"Hah. After multiple intermittent weekly checks, three-plus months into the process, I gave it one last whirl to see if I could perhaps retrieve some data. (SpinRite still hadn't completed, but I needed to give my power bill a break. Enough was enough.) Avast, not SpinRite hype, but reality. Bingo. The entire file structure not only finally appeared, but a fraction of the files were now fully accessible. The CRM data was there and valid. It was quickly backed up, pressed and plattered on a CD-ROM and on other more reliable retrievable media. Lightning plus good drive gone bad plus SpinRite plus time equals a good deed done for the year. What a cool project. Thanks."

Then he says, "The IT guy who solicited me has since left the company, a folly of the timeframe, I suppose. But please know that I give SpinRite a sales pitch even when talking about things like Kool-Aid or Starbucks, to try to make up for the grief I carry knowing I used it on someone else's drive. However, I know it has resulted in an extra couple of sales for you just telling the story, and to my other IT friends. Again, thanks for SpinRite. Thanks for my Gibson degree in security. And thanks for your dedication to producing the show."

Leo: That's pretty amazing. So how long did it run?

Steve: Three-plus months, he says. It sounds like about maybe three to four months.

Leo: So it's okay to stop it periodically and see what it's done and then continue it on?

Steve: Oh, absolutely, yeah. You stop it, it gives you a percentage completion accurate to four degrees or four decimal digits' worth, so that you're able to resume at that point. It used to be in the old days we would write what I call the little fingerprint file on the root of the drive. But since we're now able to run on unknown drives in unknown file systems, and since I'm unwilling to write in that first unused track of the drive - we know what happens when you do that, you mess up the...

Leo: No, you shouldn't write to the drive at all.

Steve: No, we do no writing to the drive at all. But so I give you a where we were accurate to four decimal points so that you're able to go back and resume SpinRite from that point. So, yeah, so you can stop it and see if it gets better. I don't know if you know that Dvorak is currently running SpinRite.

Leo: No, I didn't know that.

Steve: We got a whole bunch of information from people saying that he was complaining in some venue of his about the family's main machine had gone down. And when I finally got myself back into email - remember that I crashed my Windows 2000 box last Thursday. So it was over the weekend, I think it was Sunday that I finally got email up and going again, and I found John had tried to contact me through every channel known. People...

Leo: He didn't go through me. I could have reached you right away.

Steve: Anyway, so he said, hey, can you hook me up with a copy of SpinRite? I want to see if my family's main gaming machine can be brought back to life. I said yeah, of course. So it'll be interesting to see what kind of success he finds.

Leo: Yeah. It just depends. I have a RAID array, in fact we're recording on it right now, that when I boot up, it says errors found in the second disk in the array. It still seems to work. So I ran SpinRite on it, and there was nothing wrong with it according to SpinRite. So I don't know what that means at this point. And it's still working. So of course I record three different versions of the show, so it's not like we're running a great risk. But I think probably I should rebuild the array, I think.

Steve: Sounds like maybe there's - oftentimes some technology in the RAID controllers, where they'll go through and do some sort of scrub of the array, typically at the expense of your data. It's not a nondestructive process, which SpinRite is. But still it's probably something that is worth - or Leo, just get rid of the drive. Swap...

Leo: They're cheap enough, huh. I probably could, yeah. Okay. I have 12 fantastic questions for you. Are you ready?

Steve: From our fantastic listeners.

Leo: Absolutely. Starting with Postdiction in Chicago. He's losing his memory. Uh-oh. Hi, Steve and Leo, love the show, try to listen every week. Recently an older laptop, a P3850 with 384MB of RAM - oh, that is older - had some memory die, and now I'm down to 128MB of RAM. He says he was running XP on the machine. I don't even know how he did it with 384. But of course it runs too slowly now with 128. He says: I tried dual booting with Windows 2000. Windows 2000 runs great. I was wondering if Steve could briefly go through how he sets up and locks down Windows 2000 on his laptop so I could feel secure taking this machine into the wild. That's kind of a problem because it's no longer being supported by Microsoft; right, Steve?

Steve: Well, yes. First I wanted to clarify that all my laptops, being newer, are now using XP. And I'm comfortable...

Leo: Newer, not new. Are you running Vista anywhere yet?

Steve: No. No.

Leo: Of course not.

Steve: I think I've got it in some VM somewhere. It's like - oh, no, I do have it on one machine. It's like I installed it, it's like, oh, okay. You know, and if I ever - I ought to, like, check my software on it. And I guess at some point I'll need to be doing screenshots for the website. It's like, okay, this is what this thing looks like because people are - are we over 50 percent adoption? What percentage of...

Leo: They sold 100 million copies - I'm sorry, 150 million copies last year, in 2007.

But that's only half of all of new machines sold. So...

Steve: And you can't give them back. So...

Leo: I think it's well less than half. But I actually haven't seen statistics. That's a good question. But we haven't gone all the way there. Now, what's your primary OS? Is it still Windows 2000, or is it XP now?

Steve: No, no, it's XP now. In fact, as I mentioned, I did something dumb last week on Thursday. I killed my Windows 2000 server workstation that I've been using probably for about a decade. I was having problems with the IIS web server in it, which I use for testing all my stuff before I put it up on the public server. And I uninstalled it and reinstalled it, under Microsoft's online advice, and it just brought the house down. It was just, oh, goodness. Now, I didn't lose any data. Everything's backed up. I've got my registry and all that. And I've been, you know, I've talked about my quad core monster machine that I've been sort of wanting to get set up, but who wants to take the time? There's always something more important I have to do.

So anyway, this finally - that was the straw that broke the camel's back. And now I'm sitting in front of a barely configured new Windows XP system that I'm excited about. I mean, I'm sure a couple weeks from now, once I've got everything back in and my development environment set up and tuned and customized, I'll be really happy. I did, I mean, I'm taking it slowly because I want to really nail, you know, just like bolt this thing down in terms of just how it's running. For example, I've got every unnecessary process stopped so that when it boots it uses 131MB of RAM. So which is very lightweight for Windows XP. So most times you'll see, like, maybe 3, 400MB in use. This is 131. So...

Leo: That's kind of impressive.

Steve: So anyway, I wanted to respond to this listener, Postdiction, to mention that I'm not using 2K on a laptop, so I'm using XP. What makes me comfortable with XP is its built-in firewall. Win2K does not have one. And that is the one thing you absolutely want to add to any machine, a portable machine that you're going to take around and plug into things. You absolutely can't have it exposed without some sort of inbound blocking capability. From my standpoint, outbound blocking that allows you to manage individual apps, that's much less important. What's critical is inbound blocking.

Now, one thing you could do would be to take one of these little tiny mini travel routers and always run the Win2K machine behind the little travel router, which is going to get - as we know, routers make really good hardware firewalls. So that would provide you with good protection. But that's probably unnecessary. I would say just choose any well-regarded current personal firewall, put it in there, and make sure it's running. And that's really the only thing you have to do to protect yourself when you're out roaming around is just make sure you're not allowing unsolicited incoming traffic.

Leo: How do you address the issue of no patches?

Steve: Oh, you mean like in the future?

Leo: Well, they're not patching Windows 2000; right?

Steve: Oh, no, they have been. I've been getting updates till last Thursday.

Leo: I thought they stopped.

Steve: No, I think they had to extend it. Or I think maybe it's not - maybe it's what, is it security, or I guess I should know this.

Leo: Security only, I'm sure.

Steve: I haven't worried about it because I've been getting patches constantly. Every time I do Windows Update it says, oh, here's some new stuff for you. So it's like, okay, fine.

Leo: So as long as it's being patched. Now, when they do end-of-life it - I could have sworn they'd done that already. But when they stop patching it, then is it too dangerous to use?

Steve: One of the things that happens is you begin - the target of opportunity begins maturing and moving. I've got people running on Windows 98 ME that have never, I mean, these are not computer-savvy people. They only need to do email and browse the web. They've never gotten infected because all of the new exploits are against the new technology. They're against XP and Vista. Nothing, I mean, it's sort of like your genetics are no longer compatible. So nothing infects Windows 98 anymore. No one's doing things to infect that old funky platform. So it's become safe. It's sort of like a Mac or Linux. It's just no longer a big target, even though it is technically Windows.

Leo: I'm looking at the Microsoft site. They say they will continue to offer security updates through the life of Windows 2000, which means through 2010. So I think they do a 10-year life cycle on all their OSes. Okay, so you've got another year or two.

Steve: It's going to be really interesting to see what happens with XP because apparently they've had to agree to extend XP's life or something or other.

Leo: Well, here's the deal. They were going to...

Steve: It just won't die.

Leo: Yeah, they were going to stop selling it at the beginning of the year. And so many protests that they now say June 30 they'll stop selling it. But support goes on for a while.

Steve: Okay.

Leo: You know, typically they say what they call mainstream support for five years, extended support for five years after that. And then security updates go on for the entire 10-year...

Steve: And if they would just make it right, they wouldn't have - this wouldn't be such a problem.

Leo: Well, yeah, but there's always going to be holes; right? I mean, even if it's perfect...

Steve: Well, and notice also, Leo, I mean, the fact that we're having this discussion demonstrates that the reality of an unsupported OS moves people forward. Microsoft is desperate to get everyone over to Vista. I mean, I don't blame them. It's a pain to, like, be supporting three OSes - 2000, XP, and Vista in all the different flavors. They don't want to be doing that. So they really, I mean, not only economically, but just in terms of the burden for them, they really want people to get off of Windows 2000, at least get up to XP.

Leo: There's still people running NT4 out there, and it isn't getting patched.

Steve: Yeah, and it works.

Leo: It works, and I understand why those people don't want to upgrade. They say, look, it's doing what it needs to do. But it's not being patched, so I don't, you know - and if I'm a hacker, I figure, hey, what better machine than a machine that is completely neglected to the point that it's still running NT4.

Steve: Well, yeah. Now, okay. If, for example - I would guess that those NT4 boxes are servers. They're NT4 servers. I don't imagine anyone's walking around with NT on a laptop.

Leo: Oh, no, I'm sure they aren't, yeah.

Steve: And so in a server environment it's - there were some horrible problems with NT's IIS in the beginning, all kinds of URL exploits. So I imagine that those have been nailed down. And now it's probably bulletproof unless they add something to it that causes a problem. But again, it's stable and bulletproof. At some point it's like, hey, if it's

not broke, don't fix it.

Leo: After 10 years you'd figure everything that's got to be found is found.

Steve: Yeah.

Leo: TJ Schroemer in Houston, Texas wants to know a little bit more about the security of web forms. He says: When I type HTTPS: for, let's say, Discover or DiscoverCard.com, it always ends up back to HTTP:, no "S." The web desk tells me when I click the login button it changes to a secure connection and then transfers my login information safely. Is that right, or are they blowing smoke up my cautious risk management? Is my login information safe if it's loaded into an unsecured page? Is it supposedly changed to secure when I press login? I think we talked about this once before, as a matter of fact, because I was concerned, too.

Steve: Right. We have talked about it. And it does come up from time to time, so I just sort of wanted to revisit the question. First of all, I would blame the web designers for the fact that this is causing confusion.

Leo: A lot of pages do this.

Steve: Yes. The idea is that the way a web form works is, because the web was originally designed as a query system, that is, you had a web browser which browsed, and you had servers which served. The browsers were not supposed to be sending data back in the other direction. It was supposed to be - it was like sort of a passive, document-reading model. So in order to sort of shoehorn information flow in the other direction, they created a kludge which is they took this query-based model where the site is asking for things, and they added the information you want to send back to the query. So you're essentially - what it is you're asking is actually what you're sending. The data that you're sending is part of your question. And the way it's formatted, the server goes, oh, look at this wacky question. Oh, wait a minute. Here's the question part. And then tacked on the end is all this other stuff which is technically not part of the question, it's a submission to the server.

So what that means is that a connection is being made to the server which is first secured - remember, that's how SSL, Secure Sockets Layer, works. That is the HTTPS, the "S" stands for SSL or for Secure. It creates the connection, establishes through some handshaking, which is well designed, an absolutely secure tunnel using an ephemeral key that will not be used for longer than that connection. So it's very secure. And then over that connection goes the query that contains the form data. So it is safe, is what that means. The troublesome thing is that there's no way without viewing the page source and finding the button and finding the URL that is this query, for the typical user, the armchair surfer who's been listening to Security Now!, to figure out whether this button is going to send the data over a secure connection. And I don't know if you've looked, Leo, but it used to be feasible to do a view source of typical web pages. Now they are so gunked up with junk, I mean, it's really hard to figure out what's going on by viewing the source of a page.

Leo: Well, there really is a flaw here. None of this was intended to be human readable. Nobody, Tim Berners-Lee didn't expect you would be seeing HTTP://. What's too bad is that the browsers which do have this supposedly human-readable lock-and-key thing aren't able to tell either.

Steve: Right. Now, I started off by saying that I really blame the web designer. A properly designed site where you are submitting data, the page with the form would come up secure. Even though it doesn't need to be. I mean, the point is...

Leo: Just do it a little earlier, just to reassure people.

Steve: Yes, exactly. Switch to an SSL connection on that page, so then you get the happy little green coloration, if you've got extended certification and IE7 or a browser that supports that, to give everybody a nice warm fuzzy feeling that says, oh, good, look, this is all secure. Now, the good news is, I know that IE, and frankly I'm not sure of other browsers, but I wouldn't be surprised - you probably do know, Leo. IE has an option that allows you to tell IE whether you're allowing it to submit unsecured form data. So IE can be set to alert you when you press the button if that's not - if it would not be a secure connection at the time of sending. Because the other thing that could happen is, if this page were really poorly designed, you could have a secure form display, but the button that submits the data could be insecure. And you wouldn't know that, either.

Leo: I think all browsers have that. Unfortunately what happens is the first time you submit information that's not encrypted, you get a little warning box saying you're doing that, and then a checkbox that says it's an opt-in checkbox. It says, if you'd like me to warn you about this in future, check this box. But the default behavior is not to.

Steve: Ugh.

Leo: So I would go, and you can do this I know in Firefox certainly, you can go into the security settings and, under the warnings messages, make sure you check the box that says "Warn me when I submit information that's not encrypted." Now, you're going to see that all the time. That's the problem. You're going to see it in a lot of pages where you're submitting form information. But at least you'll have a warning if you're trying to do something on a page where you've giving something you'd want to protect.

Steve: Right.

Leo: I think that's why they default to off, is you would see this all the time.

Steve: Right.

Leo: Anytime you submit a form. Because most of the time it is insecure and doesn't need to be secure. It's only when you're logging in it's not.

Steve: Just not very well designed, all of this stuff.

Leo: You know, this was all designed before security was an issue. And we're still stuck with a lot of this legacy. Jeffrey in Columbia, Maryland is weighing TrueCrypt versus IronKey. I think we've confused everybody by talking about three, actually, really good technologies: CompuSec, TrueCrypt, and then IronKey, almost in a row. Steve, after Episode 135 I've been fighting with myself over what I should use now. TrueCrypt and a USB key, or IronKey? To me, as an everyday computer user who's security minded, they both suffer from the same problem. They're both worthless in a sense if your client computer's already been compromised. We did talk about that.

And is TrueCrypt really that much "less secure" because it's software based, and someone can attempt brute-force password crack on it? Please correct me if I'm wrong, but can't you also use TrueCrypt's keyfiles as part of your key ring material to make up for this? To my way of thinking, you still can't beat TrueCrypt. It's free. Personally I don't want to trust another company as far as their servers and such for the storing of my password, regardless of how secure they claim to be or how much they say they can be trusted as being on your side and in your best interest. That's the point I always make about open source versus closed source. In my opinion, this is one of those Trust No One situations you talk about.

Finally, regardless of the situation, it would take a state-sponsored enemy to even attempt - like the NSA - to even attempt to break either of these items to begin with. So what really is the scare in the end to the everyday user? Just curious. Look forward to your thoughts.

These are great. This is the kind of thing actually I'd like to ask you, Steve. I think Jeffrey's right on there.

Steve: Okay. If price is not an object, and if IronKey were as tiny and cute as the little Kingston 4GB thumb drive I have on my keychain, I would use IronKey. I think. But it's huge. I mean, I'm not having that monster thing on my key chain. Many users asked about IronKey. We completely covered it last week. I'm really glad we had Dave on. I think it was a tremendous - and actually I got a lot of nice feedback compliments about how much they liked the idea of us having guest people on, and Dave in particular, which I thought was neat. I agree, it was really fun to have him. But I'm using TrueCrypt on my little 4GB dongle, which my car mechanics are free to do anything with that they want. I mean, that's my choice.

So I guess it's - is it the case that TrueCrypt is safe enough? Absolutely. I mean, do you need a hardware encryption on the device? No. In fact, even USB dongles can be used as third-party, I mean as multifactor authentication. So you could do two-factor authentication. So, I mean, there are ways to do this without the IronKey solution. I think for the really high-end, extreme secure, maybe corporate user or something, maybe IronKey makes sense. But I've got to say TrueCrypt is what I use.

Now, the one interesting danger about TrueCrypt which IronKey doesn't have is that you can pull - a bad guy, say that my, you know, the car mechanic, I turned my keys over to

him, and I didn't have the little key release gizmo that has been suggested, he could copy the encrypted partition, that is just a file, from my system to his, and give me back my keys and my car, and away I drive. Well, now he's got my encrypted file forever. He can have it as long as he wants. There's no way that TrueCrypt prevents him from copying it off. There was the comment made about keyfiles, that is, Jeffrey mentioned that with TrueCrypt you're able to use a file in addition to your passphrase. But again, copy the entire, the contents of the entire 4GB, and you'll have whatever keyfiles it might have been using, too.

So, you know, it makes me a little uncomfortable that he could have this encrypted blob on his machine to do with whatever he wants. Certainly IronKey absolutely prevents that. You're not able to get past it in order to get to your data without authenticating yourself. TrueCrypt does allow you to pull the data off and then use state-sponsored enemies to attack it. But again, TrueCrypt is, I think, just every bit as safe as anyone needs.

Leo: Okay. Excellent answer.

Steve: And it is free.

Leo: Yeah, it has that advantage.

Steve: And you can put it on cute little thumb drives.

Leo: Which I do.

Steve: And the thumb drives cost, you know, 20 bucks instead of...

Leo: I have a Corsair. I buy Corsair 16GB drives because we still use sneakernet here in the office, and it's great for transferring files to Dane so he can edit them. And it comes with TrueCrypt on it, TrueCrypt 4, but at least it comes with TrueCrypt on it. I think that's kind of a neat thing that Corsair is doing. They're just kind of bundling it along, just to - I guess in some way just letting people know you could do this.

Tyler, playing with coLinux, which I'm not familiar with, somewhere in Arizona, writes: Steve, I've found something relatively new I think might pique the interest of a low-level hacker like you, especially considering your interest in virtualization. You mentioned earlier about running Linux BSD in a VM under Windows to act as a firewall. I've tried all the VM emulators on the market, and I've always been disappointed. They all feel heavy in one way or another, either performance or size, impact on the host system. I've just always felt like technology had a long way to go yet.

Then along comes Cooperative Linus, or coLinux for short. This isn't a better VM implementation, just a whole new approach. With coLinux, both kernels run in ring zero. The two kernels are modified via driver to run as coroutines of each other.

Neither is absolutely in charge. In the interest of sanity, the Linux kernel only sees hardware through a virtualization layer. Otherwise if you put in a USB key, which OS would take it. But other than that, the systems run as normal. It's called virtualization because that's what it's most similar to, but nothing is actually being virtualized. Both OSes are running on bare metal. This creates a significant advantage. There's literally no perceptible overhead.

It doesn't use any processor time or memory that's not directly accounted for by some application on the guest system. Not only does the guest OS run at 100 percent normal speed, there's no additional cost to the host system, either. There's no VM engine, so all the memory and CPU cost of the guest OS is accounted for by the guest's applications. With Fedora under coLinux running in the background, I could still even play high-end games with no slowdown whatsoever. Try that with VMware. CoLinux can even be run as a Windows service, making it easier to use as a personal Linux firewall.

The downside - oh, well, I knew there would be one - is security and stability. Oh, well, great, okay. There's no separation between the two operating systems, so a kernel-level exploit under Linux could jump the gap and compromise Windows. Similarly, a kernel panic under Linux might cause a blue screen under Windows. Also coLinux is very much still in development. The whole architecture is still being built and rebuilt as they work toward an optimal solution. However, it already beats the pants off everything else out there. I've finally found a VM environment I can literally live with as part of my day-to-day computer usage is concerned. I highly recommend you have a look at it.

Steve: Well, this is a very, very interesting approach. I wanted to mention it because I feel the same way that Tyler does about sort of like the persistent use of a VM. As we've talked about, you need to commit a chunk of your host system memory to a VM. That's the thing that gets used up first. It's one reason even to have a swap file, even if you've got, for example, 3 or 4GB of memory, a swap file will allow multiple VMs to swap out to disk because VMs are very memory hungry and resource intensive.

What the coLinux guys have done I think is really interesting. It's the reason I wanted to bring this to the attention of our listeners is, despite the fact that you lose some of this notion of isolation, increasingly our listeners are using multiple OSes. That is, they may be Windows people, but they want to play with Linux; or they're Linux people who want to have Windows around, but not in the way. Essentially, this allows the Linux kernel to run as a device driver, as a service, essentially, in Windows, and not be in the way. It's able to give and take memory to and from Windows as it needs it so that if you run a Linux app, the Linux app runs in Windows memory rather than in, like, a separate sequestered VMware or virtual machine of some kind memory.

So, I mean, it's really practical from the standpoint of wouldn't it be nice to be able to have Linux around if your main platform was Windows, but you wanted to have access to Linux stuff. So it's an interesting approach. If you just put coLinux - c-o-l-i-n-u-x - into Google, it'll take you right to their site. And it is new. It took about a month for them to bring it up the first time. So it's an interesting sort of lightweight approach to two OSes running side by side in the same box. And I'll absolutely bet that our listeners will find some use for it.

Leo: It's an, yeah, it's a really interesting idea. But you wouldn't do it for security, which is why a lot of people are doing this.

Steve: Yeah, well, I would say it's one of the - maybe it's half the reason people are doing it. I mean, security is one thing. But often there are times where you just want to have a different OS around, for any of a number of reasons.

Leo: Well, actually that's true. When I run VMware or Parallels on my Mac, it's just so I can run Windows. It's not so I can be secure, obviously.

Steve: Exactly.

Leo: Tim Bousky, listening from Singapore, was wondering about relocated sectors: Hi, Steve. In Episode 134 you briefly described how modern hard drives will automatically relocate sectors that are deemed bad while retaining the original sector number. If that occurs with any frequency, won't it tend to erode the benefit of utilities like Perfect Disk and other defrag utilities, since sequentially numbered sectors won't necessarily be located sequentially on the hard drive? Love the podcast. Happy SpinRite user, devoted Security Now! listener, Tim.

Steve: Well, Tim is technically correct. Although the relocation strategies of drives differ. One of the interesting things many drives will do is they'll have pools of spares scattered around the drive, like some of them have them at the end of every cylinder of the drive, although cylinders now have sort of lost their meaning because drives are regarded as a linear array of sectors as opposed to cylinder-headed sector, the old 3D approach of accessing a given sector, sort of by its coordinates physically on the disk. But what drives will do is they will slide sectors down so that if a given bad sector is found somewhere, what happens is, given that its data can be recovered, which is what the drive requires - of course using SpinRite gives you much more power over that process. But given that the sector can be recovered, it will be - essentially at the end of this run of sectors, however long it is, whether it's a track or a cylinder or some arbitrary span, at the end of a run of sectors is the empty sector pool. The drive literally slides all the sectors down by one, reducing the size of the pool by one, and essentially keeping them linear, but skipping over the bad one.

So if you imagine like a hundred sectors, and the tenth one is bad, well, what happens is those - I'm trying to get my math right - the 89 sectors following the tenth one, they're slid down, all of them are slid down by one, thus encroaching into the spare buffer at the end of that run, and essentially keeping them all linear. So the drive's performance is impacted only marginally in that it has to just sort of ignore one sector. And then it keeps on going in a linear fashion. So it's very cool.

Leo: I think this is the part where people rewind a couple of times to understand. Okay. Sure, I'll let you do that, folks. And we'll stay here until you come back. Number six, Shawn White in Osaka, Japan doesn't want to write too often. It's okay, Shawn, really. Dear Steve, during last week's interview with IronKey's Dave - great episode by the way - he briefly touched on the types of flash memory. He talked - he

did. It was, I thought, very interesting. I'd never heard this, SLC and MLC. Talking about keeping data safe when using portable applications, say for instance the PortableApps.com suite, where our users - our university - uses Firefox, OpenOffice, Audacity, GIMP, NVU, have you any idea how you would estimate the number of write cycles that a flash drive would be subjected to? As we move more and more towards portable computing or cloud computing, will MLC really be enough? Should we be in SLC? Thanks for much for the greatest source of tech info on the 'Net.

So I gather where Shawn's going to school in Osaka they're using a thumb drive and just moving around from computer to computer with all of their stuff on it - OpenOffice and Firefox and all of their programs that they use. And he wants to understand how the number of write cycles will impact that and whether SLC is better or MLC is better than - or SLC is better than MLC for that.

Steve: Let's do a little bit of de-acronymizing here.

Leo: Yes, please, yeah.

Steve: SLC stands for Single Level Cell. MLC stands for Multi Level Cell. And MLC technology was developed, that is, multilevel cell technology was developed to increase the density of data that can be stored in this flash ROM or RAM or PROM or e, you know, lord knows what you want to call it. But what's very cool about multilevel cell, that is, MLC, is it stores two bits per cell. The way it does it is by storing essentially four voltage levels in the cell. You have all off. Then you have one quarter, two quarters - well, I mean, sorry. All off, one third, two thirds, and three thirds. So you literally - we're all used to thinking in terms of binary. But these are, what is that, the quaternary technology, where they realized, wait a minute, we've got enough resolution in our ability to write and read that we don't have to just store ones and zeroes. We can actually store a zero, a one, two, or three. And that gives you two bits of data.

So that MCL stands for multilevel cell. There are a couple problems with MLC, which our listener apparently understands because he was sort of talking about SLC versus MLC. First of all, MLC has about a factor of 10 fewer write cycles that it's able to handle. It can do 10,000 writes before it begins to have problems. Whereas SLC, single level cell technology, can do 100,000. And that's primarily just due to the fact that you're only storing a zero or a one. And so as the cell's ability to store degrades, it's easier to tell the zeroes and the ones because they're, like, absolute; whereas with multilevel technology you're storing a zero, one, two, or three. That is, four different voltage levels, essentially, or charge levels, in each cell. And so as that begins to degrade, you could start having problems. So the multilevel cells have about one-tenth the write cycles. They do have a higher error rate. Although all of these technologies use ECC, Error Correction Code, in order to essentially ignore and deal with those kinds of bit errors.

Now, all of this, any high-end memory, as Dave was talking about, and IronKey is one of these, will use what's called "wear leveling." Because of the sensitivity to the number of write cycles, when you are writing to a given area, there's a sort of a mapping layer that is between you and the actual physical ROM. So that when you're writing to it, you're actually writing to a different zone of the memory, even if logically you're writing to the same spot. For example, in a hard drive, when you write to a given sector you're actually writing to that sector, except in the case we talked about before where there may have been some sparing going on, where a sector was swapped out. But in the general case

you're physically writing, given that all other things are okay, to the same location over and over and over and over and over.

Not so with our solid-stage memory, our non-volatile, solid-state flash memory. In that case, with wear leveling, they're deliberately sort of spreading the writes out across the physical surface. The bottom line is, if you have an SLC technology that's good for 100,000 writes, and you imagine, I mean, you're literally able to calculate how much you're writing compared to how large the memory is and how long it would take for you to write the entire thing 100,000 times. So if you take a 4GB memory, which is high quality, so that it's got wear leveling built in, and you multiply that 4GB by 100,000 - so let's see, by 1,000, 4GB becomes 4TB. And by 100, so that becomes 400TB. That is to say, you can write 400TB of data into that 4GB memory before you reach a wear-leveled 100,000 writes per region. So then compute how much data you're writing. I mean, the fact is, I mean, it makes people a little jittery to think, you know, wait a minute, I could burn this out, I could wear this out? Well, yes. But when you really do the math, it'll take 400TB of data written to it before you reach that point. So most use is far, far less than that.

Leo: Certainly on a thumb drive. Although when you start talking about SSD hard drives, then you start maybe coming up against that.

Steve: Oh, and yes, you don't want to, for example, have very little main memory and use SSD as your swap file, as we found out from, you know, in the early experiments that Mark Thompson did, you know, those can get burned out pretty quickly.

Leo: Yeah, yeah. So that's a great question, Shawn. I'm glad you asked that. Jim Phelan is also backing up his TrueCrypt volumes. He's doing it with Jungle Disk, and we talked about the issue with timestamping a couple of episodes ago. Steve, thanks for the errata tip about the TrueCrypt switch, a little switch in the program that forces the software to update the timestamp on a TrueCrypt container. I was also having trouble getting Jungle Disk to back up my encrypted files. After hearing your tip on Security Now!, I checked my TrueCrypt preferences and found there is a way to set this option in the TrueCrypt GUI. Go to Settings, Preferences; uncheck the box that says "Preserve timestamps on all file containers." It gives you a warning about losing plausible deniability. Okay, because, you know, they could say, hey, you've been updating this. But once you do that and restart TrueCrypt, every time you dismount a TrueCrypt volume it updates the timestamp on the encrypted container. Thanks for the best podcast going. Good tip, Jim.

Steve: Yes, Jim and many other listeners mentioned that this option was in the GUI. So I wanted to make sure our listeners know. We talked about a command line option that you could add to the end of the TrueCrypt EXE invocation. Certainly for the typical user going in and unchecking "Preserve timestamps on all file containers" is a lot easier to do. Now, one reason you might not want to do that is you might not want to make that change globally, in which case using the command line switch would allow you to do it on a container-by-container basis as you are telling TrueCrypt to mount a certain container and dismount it. But I did like the option.

And I just loved, again, I was guessing from hearing that the timestamp was not being updated that those TrueCrypt guys were again doing the right thing. They were wanting to preserve the date stamp at its creation, its original creation date, specifically for the

purpose of plausible deniability. So I love the fact that, when you uncheck that, it warns you that, oh, okay, we'll update timestamps as you wish. But then someone might say, hey, wait a minute, this has today's date. You've been in there today.

Leo: Yup, very interesting. Good tip.

Steve: TrueCrypt rocks.

Leo: Yeah, they're just - they think about everything. They're true paranoids. David O., lurking about in the Bay Area of California, feels that stating "Only the NSA can do it" understates the scope of the threat. And I take responsibility for that. I'm the one, I think, who said that. Hi, Steve and Leo. Love the show. What was - our previous questioner said something, "state-sponsored organizations"?

Steve: Right, right.

Leo: That might be a better way to say it. I'm writing to ask that you consider amending something that you have said on more than one occasion. When discussing the topic of writing several passes of pseudorandom noise as a means of obscuring previously written data on disk media, Leo has been understandably skeptical about the true real-world feasibility of such sci-fi data recovery, stating something along the lines of, "I mean, c'mon, only the NSA has the sort of technology that would be necessary to read disks at this level; right?" So I wanted you guys and my fellow listeners to know it ain't so.

There are commercial services available that offer this level of data recovery - okay, I'm going to be very skeptical here, but I'll keep reading - of truly erased and overwritten disks. I used to work in tech support for a large computer company, and representatives from a data recovery service came in and talked to us once to explain that disks that were erased or overwritten could have their data recovered by them for forensic purposes. It's not cheap or easy, but it's an available service, and anyone can employ them for the price. They discussed the details of how it's done, much in the way you explained using trace levels of residual magnetic charge.

So because there are commercial services available, it means that the scope of risk is not just the NSA at the national security level, but is expanded to the scope of, let's say, private investigators, with angry ex-wife, husband, partner, et cetera; mistrust between bosses and employers; well-heeled snoops and so on. I'm not suggesting that folks should be paranoid or overwrite their data seven times with pseudorandom noise every time they empty the trash. That might be a bit over the top. But I think it would be unjust to your listeners to understate the scope of the risk in a security-related concern on a podcast such as Security Now!. If somebody really wants your data and is motivated, a simple erase or even a zero-all is truly not secure enough. And I agree with that. I wasn't saying that a simple erase was secure enough, or a single pass was secure enough. I understand that. I question these so-called "commercial enterprises."

Steve: Well, and we do all agree that 36 passes is...

Leo: That's too much. Two or three would be more than enough.

Steve: I really think that's the case, yes.

Leo: I'd like to know who these commercial services are. You know, I'm talking not based on my own experience, but on the testimony of people I trust in this area, arena, like Simson Garfinkel, who did this study and is a very bright guy, who said, ah, one pass is plenty.

Tim O'Malley of Beach Park, Illinois, wants to remain virus-free. But pass as many times as you feel necessary. Hi, Steve and Leo, says Tim. The question stems from podcast 129 where Steve makes the comment at the end of the show, and not for the first time, that neither Steve or Leo use antivirus. Now, I know you're not denouncing antivirus or suggesting that people shouldn't employ AV. Before I go any further, I don't work for an antivirus vendor. Still, it begs the question, how do you set up your machines so that you feel safe enough not to have to use antivirus software? I've been listening for a while now and can speculate as to how you have it set up so not to run, but I would rather hear it directly from you. Thanks again, guys. Keep up the great work. Tim.

Steve: Well, Leo, I would say it's a combination of the way we have our machines set up, and more probably than anything else, our behavior.

Leo: Yeah, it's all about behavior.

Steve: It really is. We know I'm a little stricter than you are. Well, I'm much stricter than you are...

Leo: A lot stricter.

Steve: ...about scripting. I just don't like scripting. I'm going to - well, and so I surf with scripting disabled, which normally sort of catches me out a little bit. I go, wait a minute, why is this not working? Oh, that's right, I need to trust this website. Then I add it to my trusted zones in IE, and scripting is back on the way I have my zones configured, and I'm able then to do what I want. But so that, I really like that.

But the second thing is I never, ever, once ever in my life used Outlook Express or Outlook for email because Microsoft has had so many horrible problems with IE's web browsing experience, and Outlook and Outlook Express use the web browsing control in the window. So any problems that you have with IE and scripting, your email automatically inherits. Which is just the dumbest thing I've ever heard of. I mean, who ever thought that scripting was useful for email? But it's been turned on from the beginning, and we've had all kinds of viral problems as a result of that. Now, that's getting better. But those are things that hurt other people.

I've always been using Eudora, and one of the things, one of the options you can check is do not use Windows viewer in Eudora. So I turn that off, and then I use just a generic

text display rather than essentially IE, you know, Windows viewer for my Eudora. And so, yes, when I look at email, I'm not seeing what looks like a web page. I've looked at other people's email, and I've thought, wow, look what I'm missing. On the other hand, I get all the text. Email is for text. So that's really my two things are be careful about email, and in my case I turn scripting off.

But as you said, Leo, or agreed with me, largely it's about behavior. I will not click a link that I get through email. I mean, I just - I am really, really, really reticent. I have a hex viewer that I use, and I'll look at an attachment in the hex viewer to see if it looks like what it appears to be, if I'm inclined to believe that I've received something. But, I mean, I'm really anxious about attachments in email. I don't treat them casually at all. So it's just a matter of really being careful.

Leo: I just don't surf the Internet with a Windows machine.

Steve: Ah.

Leo: I use Windows all the time. I need to. I'm using it right now to record the show, to edit the show. I just don't go on the 'Net with that machine. That helps me. Now, on the Mac, obviously I still have to worry about those kinds of things, and I'm very careful. The good news is that almost everything I run up against is written for Windows. So all the email attachments and everything are all written for Windows. I think it's probably a good idea to have an antivirus in your toolkit to at least scan from time to time, just to make sure that you haven't made any mistakes.

Steve: Certainly if your system seems to have gone a little funky, then it's like, uh, maybe I ought to, I mean, I've found myself scanning, grab the current copy of AVG if I think something's wrong. And normally it's just the fact that the thing's 10 years old is the problem.

Leo: Here's a good one. This will give you a little chill. Sony Daswani submits a succinct request with the subject "to learn password of any ID" and asks, "Please give us how we can learn password cracking." Obviously a devoted listener.

Steve: I just loved it.

Leo: How can we learn password cracking, Steve Gibson?

Steve: That's not why we're here, Sonu.

Leo: He's Sonu.

Steve: You've got the wrong podcast.

Leo: You've got the wrong show. We've talked about it, I mean, peripherally, things like brute-force attacks.

Steve: Well, no, we absolutely talk about what it is that bad guys do to crack your password. You have to...

Leo: Those tables, those, you know...

Steve: Rainbow tables. And brute force, and dictionary attacks, I mean, we've talked about this a lot. But we're not in the business of teaching how it's done. We're in the business of explaining the technology used. And then from that we derive the best defenses against password cracking. The good news is, for Sonu, there are plenty of sites on the internet, and probably even some podcasts, which will help him to learn password cracking. But not this one.

Leo: Well, I think if you're following what we're saying closely, you're getting everything you need to know. It's just a step between what we describe and the implementation.

Steve: Right.

Leo: What we don't do is give you scripts. We don't teach script kiddies. But I think if you're an intelligent person you could listen to what we're talking about and say, oh, okay, I need this, this, and this, and pretty much figure it out.

Steve: Well, yes. For example, you can imagine in the case of, for example, WiFi, which we've covered extensively, all the things we've talked about, somebody who was wearing a gray hat could use - for example we've talked about how SSID, turning SSID off doesn't help, how using MAC address filtering doesn't help, I mean, we've talked about all the things that you should not do. Well, anybody who's not listening to this podcast or hasn't somehow brought themselves up to speed with state-of-the-art security, they're going to fall victim to these things. And, I mean, I still read today as I'm surfing around, oh, yeah, turn off your SSID beacon. Oh, just use MAC address filtering, that's secure. Okay, good luck with that. Our listeners know better.

Leo: They know better. David in the United Kingdom is wondering about radio. Steve, I heard you talk about the security of wireless keyboards a few weeks ago, or the insecurity, I should have said. That got my attention. I have a Logitech wireless keyboard, but I do not access Internet banking through it because of the uncertainty of its security. I'm thinking of getting a new iMac soon and would like the convenience of using their Bluetooth keyboard. However, I'm worried about its wirelessness. My flatmate got malware on his phone last year through Bluetooth. Is it possible that a Bluetooth keyboard could get compromised? For instance, could you get a keylogger in the keyboard? That would be a bad thing.

Steve: Well, okay. We're going to do a podcast here before long on Bluetooth security.

Leo: Oh, good. Because there seems to be a big debate over how secure it is.

Steve: Yes. We're going to cover it as carefully and extensively as we cover all of the fundamental technology stuff. It may be one of those episodes that people have to listen to a few times. Certainly they should not be operating heavy equipment at the time that they're listening to our Bluetooth, you know, how Bluetooth security works episode. For now, let me say that the one thing you absolutely want to do - and Leo, we discovered this when you and I were up talking about Bluetooth on The Lab with Leo a couple months ago - was you want to turn off discoverability. Discoverability is something that really should just switch itself off. I mean, if this were designed correctly, you would make a device discoverable for 60 seconds, and they would all snap back to nondiscoverable, because you really only need discoverability during the so-called "pairing" of two Bluetooth devices.

But what was interesting was we were up in Vancouver recording The Lab with Leo, and we turned on a Bluetooth sniffer and looked at how many Bluetooth devices just there in the studio. And it was, like, 20. It turned out everybody there, the cameraman and the producers, I mean, everybody had their Bluetooth devices, they just left it in discoverable. That's the vulnerability in Bluetooth. So make sure your devices are not left that way. Unfortunately, people turn that on. And then they're like, yay, I got connected, and they go off and do their next thing and forget to turn that off.

Leo: Somebody asked me on the radio show if Bluetooth encryption had been cracked. And it uses AES, so I think it's probably pretty good.

Steve: We'll be talking.

Leo: I think that's going to be a great subject. Stay tuned for that, David. Let's see, Mr. G., one more question. Are you ready?

Steve: Yup.

Leo: I can't believe how fast this went. David Miles of Westminster, Colorado is having second thoughts...

Steve: You're just having too much fun this time, Leo.

Leo: I am. It went fast. I mean, it always goes fast, but this week it went really fast. Hi, Steve, I've been using OpenDNS for over a year - as have I, so I'm very curious about your answer to this question - and over the past few weeks began noticing that Google has been painfully slow. I haven't noticed that. Then Google started accusing me of having a virus even from my Ubuntu and Fedora systems. Actually I do get that from time to time, but I don't think it's OpenDNS. But what it says is,

Google gives you a little popup that says you're behaving as a spyware or virus program would, and then gives you a CAPTCHA saying please identify yourself as a human. I went through a day of getting that a lot. And now it's stopped completely.\

He says, I finally got tired of the OpenDNS "Yikes server not responding" on every other access and returned to the Comcast name server. Suddenly Google is back to normal, no more bogus virus warnings. It seems that OpenDNS is up to some evil. They're apparently harvesting info from queries for unknown purposes. The Wikipedia article on OpenDNS has additional info. Since I first heard about OpenDNS from Security Now!, I thought you might want to take a look at this. Have you looked into this?

Steve: Well, I wanted to say a couple things. First of all, I was interested in your experience because I knew that you were an OpenDNS user.

Leo: On everything. I use it at home, and I use it here. I don't think that the Google warning is related to OpenDNS. I think that that is something Google has a problem with from time to time. It's happened to me, and it's happened to others with and without OpenDNS.

Steve: Right.

Leo: So I don't know, I mean, I can't trace it back to OpenDNS. I really can't.

Steve: Yeah, I run my own DNS server here on UNIX, and it's a server that goes out and resolves all my queries. So I'm not using an ISP's DNS servers. I wanted to mention, though, I mean, conceptually I love the idea of a third-party good-guy DNS service. We know how dependent everything that we do is on DNS. That is, you know, whenever you're looking up a website, unless it's explicitly supplied through an IP address, which is, you know, it can be done, but it's uncommon, DNS is involved in translating that human readable address, like Amazon.com, TWiT.tv, GRC.com, whatever, into its IP. Well, that gives that translator an opportunity to intercede and say, wait a minute, these are bad guys, what do you want to do about this? Or as DNS does, for example, to correct spelling mistakes. If you go to, like, SourceForge.og instead of org, it says, oh, we know what he means, and it fixes it for you.

So there's a tremendous opportunity for DNS to be essentially a filter in a very benign and non-intrusive way. And because DNS is often sort of the poor stepchild of ISPs, they'll, like, have a DNS server that's 30 years old, it's been there forever, that is an underpowered machine because once upon a time DNS didn't require much power, doesn't have much RAM, you know, my point is that an ISP's DNS server can often be very slow. And if an ISP's DNS server is slow, it will bottleneck all of their users' access because their users' access has to go through that. So it can be the case. It looks like David Miles has an ISP with a good-performing, fast DNS server. But it can be the case that using a third-party DNS can be substantially faster. And the experience is wow, you know, the Internet is faster is how people describe it.

Leo: I actually don't use OpenDNS for speed because both my Internet service providers, one of them is Comcast, are just as fast as OpenDNS. So I don't use it for that reason. I use it, it's got phishing filters, which is great. If you sign up for a free account, you can also use it as an adult site filter, has actually very good filtering, and it's done - since I put the OpenDNS server's info into the router at home, it's done throughout my house. So unless I enter custom DNS information in my computer, which I do. But the kids, you know, they haven't figured out that the router's doing the DNS. And so it's getting blocked at the router, and it's really a very effective filtering solution that costs nothing. Also they have DynDNS for people who have a moving IP address, that's very handy. I don't, you know, I've talked to these guys. I can't say 100 percent they're benign. But I have a pretty strong feeling that they're benign. I have a very good feeling about them.

Yes, one of the things they do, and this is how they support themselves, if you enter a 404, instead of getting the Internet Explorer 404 message or a nonexistent web page message from your browser, you get a nonexistent web page message from OpenDNS. It looks like a Google search page. It has ads down the side just as Google does. And it has, which I find very useful, corrections, suggested corrections. And frankly, I use that all the time. And when I mistype something, if it can't do the obvious fix like the og to org, it'll give you that page. That gives them some revenue because this is a free service. I don't have a problem with that.

So I've been recommending OpenDNS. It's a very easy thing to implement. You could either just change the DNS settings on your computer, or do as I do, do it on the router and then blank the settings in the computer. And from then on you'll be using it instead of your ISP's. I think it's a good service. And I don't see, I haven't seen any evidence that it's tied to the Google, those Google warnings. That's something that goes wrong with Google every once in a while. Have you ever seen that?

Steve: No, actually I haven't. It's interesting that you say that it believes you're acting like some sort of malware and wants you to prove that you're human.

Leo: Yeah. I have an image on my blog of it coming up. And it could be, you know, maybe it is because of something happening with OpenDNS. Maybe - I don't see why it would because once the DNS is resolved, Google doesn't know that I've done a different, used a different DNS server.

Steve: Right.

Leo: How would that impact that? You know, it's just as you - they've made too many requests in too short a time. Maybe I just use Google a little more than I oughta. I don't know. Or maybe, I don't know, maybe that's a sign I have some spyware on my system. Maybe I should be using that antivirus. I don't know. I'll check it.

Steve: Yeah. I guess my hope is that Google has been changing their behavior as they've grown huge. I guess you know that their purchase of DoubleClick has now been approved. It was approved by the EU, which was the only thing really holding them back.

And so they're going to be acquiring DoubleClick, that is of course one of the infamous trackers of...

Leo: I'm so upset about that, yeah.

Steve: I know. And so I think OpenDNS is still a relatively small group. I think they've got, like, about a dozen guys. And I hope they don't, like, start leveraging their success into needing to generate more money and becoming more commercial because that would be a shame.

Leo: I'll read the warning. It says "We're sorry, but we can't process your request right now. A computer virus or spyware application is sending us automated requests, and it appears that your computer network or network has been infected. We'll restore your access as quickly as possible. Try again soon. You might want to run a virus check or spyware remover." I've never gotten that one. Mine...

Steve: Interesting. So this is Google saying you're asking us too many things, essentially.

Leo: Yeah.

Steve: Wow. Interesting. I've never seen that.

Leo: Yeah. So that's actually one I've not gotten. I've gotten the one that just says, you know, you're acting like spyware. So I don't know, it's a very interesting question I'd like to know more about. If our listeners know more, that would be a good subject for us to talk about in a later show, too.

Steve: Yeah, yeah.

Leo: Steve, we're out of time. I want to encourage everybody to go to your website. We're not really out of time, we have as much time as we want. We're out of questions.

Steve: We filled up their RAM. Their RAM runneth over.

Leo: My RAM is full. I need a nap.

Steve: Next week we're going to have fun. We're going to address this issue of RAM hacking. All this wacky, spray Freon on your memory chips and put them in your refrigerator and recover some of the drive and all that kind of wacky stuff.

Leo: That's a very, very good topic. That'll be fun. Of course in the meantime you can go to GRC.com, that's Steve's site. Now, GRC, Gibson Research Corporation, of course is the home of SpinRite. And if you need SpinRite, you should get it there. And do get it because it's the greatest. It's the disk recovery and maintenance utility of all time. You know, just having it here to run just gives me such great peace of mind. I also suggest you go there for 16KB versions of the file. So if you want to listen to this on a bandwidth-challenged system, that's a good way to do it, save yourself some bandwidth. Doesn't sound too bad. We also have transcripts, so you can read along as you listen, and show notes, too. That's all at GRC.com. Steve, we'll see you next week.

Steve: Did you run SpinRite at level 4 on that RAID drive, or just 2?

Leo: I think 2. Should I run 4?

Steve: Yeah, you should. That's probably why something is going on. 2 just does a - it's a read-only pass. If any sector causes trouble, then it drops into level 4 and massages the sector to figure out what's going on. But it's possible that you could have just run it - also maybe the drive was less hot than it is normally when it's been in there, like you shut the machine down, you recabled it, you ran SpinRite on it, that would have given the drive a chance to cool off. And so it might be something thermal also. But give it a shot on level 4.

Leo: It's a weird, you know, unfortunately it's such a useless error. It just says there was an error on the BIOS bootup message. And then everything was fine. So I don't know. I will, I'll do a level 4.

Steve: On your way out the door to - where is it you're going?

Leo: Australia?

Steve: Australia, yeah. Just fire up SpinRite. It'll be done by the time you get back.

Leo: It won't take that - see, now, people are going to think SpinRite takes long. It does, you know, it takes, you know, you do it overnight at worst case.

Steve: Generally like three hours for 80GB, I read someone's mail earlier. So three hours for 80GB.

Leo: Okay. Thank you, Steve. We'll see you next week on Security Now!.