SECURITY NOW!

Transcript of Episode #140

Listener Feedback Q&A #39

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-140.mp3
Quarter size (16 kbps) mp3 audio file URL: http://media.GRC.com/sn/sn-140-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 140 for April 17, 2008: Listener Feedback #39. 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!, the podcast that talks about online security, privacy, hard drives, eBooks, and occasionally coffee. Actually you're not as bad - Steve Gibson, ladies and gentlemen. You're not as bad as Paul Thurrott.

Steve Gibson: Oh, is he a major - that's right, I have heard Paul talking about coffee.

Leo: Yeah, Windows Weekly has become in some respects the coffee show. But you have - now, you still - the last time we talked about coffee was some time ago. You had installed the plumbed espresso machine, which all gaze on in wonder.

Steve: Yeah, I've switched back over to Starbucks mode. I get up now every morning at 4:40 a.m., believe it or not. And I'm the first person into Starbucks. It opens at 5:00. And I read an hour and a half with my Kindle what's been going on for the last day. So do about an hour and a half, about 90 minutes of news updating while I sip on my first quad Americana.

Leo: Quad. Only a quad?

Steve: Yeah, because, you know...

Leo: What happened to the quintis?

Steve: I do quinti ventis, or quinti venti lattes. But, yeah. And when I'm traveling I'll do lattes. I don't know why, but that's sort of my habit. But normally just Americanas, mostly because I don't want to be drinking all that milk. I don't need all that milk. And then I take my second one - oh, and I prepay for two refills. So then I take my second one to go. At about 6:30 I pull out and then start the day. And then about three hours later that's gone, so I go back. And it's just sort of nice to get out of the house and stretch my legs and breathe some air and remember what the sun looks like. And so then I get my third - my second refill, my third coffee, and that's my final one for the day. And that's all we're going to talk about coffee. I'll turn it back over to Paul.

Leo: Back over to the coffee show.

Steve: Turn it back over to Paul.

Leo: Wow, yeah. But it's a good point. It is kind of nice when you work at home all the time to have somewhere to go once in a while just to - I do the same thing for lunch. I could eat lunch in, but I like to stretch.

Steve: Just to remember that there are still people outside. I mean, there's still people wandering around. And actually I had some lattes last week because I did a little bit of traveling. We haven't mentioned this on the show, but next week's podcast will be my coverage of last week's really, really interesting RSA Conference 2008, as they called it, which is of course the industry's big, the preeminent major security conference. I was contacted by them, and they said, hey, Steve, we noticed you're not registered. How about if we give you full access to the conference and press credentials? And I said, well, that'll be great. So while you were in Australia taking pictures, I was in San Francisco looking at the street people and...

Leo: And learning about security.

Steve: Yeah, exactly. I'm not a city person. I like my suburbs.

Leo: I'm thinking - was it at Moscone, RSA?

Steve: It was at Moscone. And I mean, there were some things, some memorable things. I met the guy who misspelled the "referer" header.

Leo: Oh, with one "r."

Steve: With one "r." Remember you and I talked about that once a long time ago.

Leo: Drives me crazy.

Steve: And I think my favorite little takeaway slogan, we've all heard how people say "Information wants to be free"? Information wants to be free. Well, at RSA they added another clause to that. "Information wants to be free, and code wants to be wrong."

Leo: Yes. Isn't it true.

Steve: I love that. Code wants to be wrong. That's every bit as true as information wanting to be free. And of course this whole show is about code's success in being wrong.

Leo: We should rename the show to "When Code Attacks."

Steve: Yes, but when doesn't it? As a matter of fact, that's a perfect segue into my first errata. But I've got a bunch.

Leo: Well, go ahead.

Steve: Okay.

Leo: Go to it, and I'll do - we'll save the Audible for a little later.

Steve: Okay. We had a big black Tuesday, the Microsoft 2nd Tuesday of April, where there were eight security updates, every one of them critical. And probably the most notable - and I should say that there's much proof-of-concept code that's been released, and there is exploit code in the wild. The most significant one is our old friend the Windows Metafile. And I did a little research looking back, and pretty much every year - remember of course the classic Windows Metafile was the one that you and I, mostly me, made so controversial back at the beginning of '06, little over two years ago, where it was clear to me looking at the nature of what was wrong that this was just something that had been left behind but was originally put in on purpose. Well, then a year later, around this time in '07, there was another major metafile problem.

And here we are in '08 with a serious one. This affects both 32-bit and 64-bit OSes, Windows 2000 SP4, all supported releases of XP, Server 2003, all versions of Vista, and Server 2008. I mean, it just - it's every OS that Microsoft has. And quoting from Microsoft's own details, they said, "A remote code execution vulnerability exists in the way that GDI" - which is the Graphics Device Interface - "handles integer calculations."

So there's an integer vulnerability of some sort. It says, "The vulnerability could allow remote code execution if a user opens a specially crafted EMF or WMF" - that's Enhanced Metafile or Windows Metafile - "image. An attacker who successfully exploited this vulnerability could take complete control of an affected system." Well, it would be affected then. "An attacker could then install programs; view, change, or delete data; or create new accounts." And then they say in a separate section, "In a web-based attack scenario, an attacker could host a website that contains a web page that is used to exploit this vulnerability. In addition, compromised websites and websites that accept or host user-provided content or advertisements could contain specially crafted content that could exploit this vulnerability. In all cases, however, an attacker would have no way to force users to visit these websites." Okay, big deal. "Instead, an attacker would have to persuade users to visit the website, typically by getting them to click a link in an email message or Instant Messenger message that takes users to the attacker's website."

So this is the new model of exploit that we're seeing are these web-based attacks that take advantage of the inherent vulnerability of web browsers, which is inherent just because, first of all, they're so complex. And of course scripting is a big problem. This isn't, in this case, a scripting-oriented problem. This is a problem - actually it's a heap overflow vulnerability in EMF and WMF files. And then there's another different stack overflow in Enhanced Metafiles. So it's just like, oh, goodness. Basically it means anything that can cause Windows to display a picture - and, I mean, that's opening your email with preview mode active or doing anything that shows one of these specially crafted images. And you notice that Microsoft is now - they've enhanced their jargon here to talk about "or advertisements." Because we have seen instances where ad content, ad imagery was not sufficiently vetted, or in some cases there's no vetting at all, where an ad server was hosting infected advertising images, without intending to of course, on a huge number of sites.

Leo: It happened on MySpace. It was a terrible thing.

Steve: Yeah. So anyway. So if for some reason it is not possible for people to patch, there is a registry tweak that can be put in to just disable metafile processing. That's probably not a big deal because frankly, you know, metafiles are there. They've been there from the beginning of Windows. But they're not typical images in email or web pages. As we know, web pages are JPGs, GIFs, and PNG files, typically. So you could successfully disable Windows Metafiles. And maybe that's what the corporate guys are going to do, use a script to push out a change to disable metafile processing.

Leo: Because they don't really like to update all the time.

Steve: Well, exactly. I mean, we've seen cases where updates cause more problems than they cure. An update causes an absolute problem, whereas the vulnerability is a potential problem. But I'm sure that our listeners are staying on top of this. This was the standard Windows patch. All of my Windows machines were showing a list of these problems that needed to be fixed. Actually when I got back from San Francisco they were all waiting for me. So anyway, it's just another typical black Tuesday. This one, well, I guess we've had a couple non-event patch Tuesdays this year so far. But this is a big one, so I just wanted to give our listeners a heads up and say don't miss this one.

Leo: Boy, I tell you, I'm surprised that that's still happening with the Windows Metafile. Aren't you?

Steve: Well, and across all the OSes, Leo. Which says they're sharing code. Maybe this is, like, 32-bit code that runs in a 64-bit context under the 64-bit OSes. Or it was a problem that existed even in source that survived compilation into a 64-bit target environment. I mean, it's amazing to me. But it's like, whoops. I mean, this stuff is really complex. And as we know, our new slogan is "Code wants to be wrong."

Leo: Apparently it does. Apparently it does.

Steve: I noticed when I was doing a little browsing around, I thought I would mention a little preemptively - or maybe it won't be by the time this is heard in two days - we're about to cross the four million Perfect Passwords delivery on GRC's Perfect Passwords page. It's about 3,600 a day are being generated. That's actually sets of passwords, so there's actually more than that many. I think it's, what, three different variations twice.

Leo: I thought your page was more a demo of the fact that it could be done, not the place to get...

Steve: Oh, no. People use it as a source of entropy. They go there, I mean, and of course we've read in our Q&A, hey Steve, I'm using one of your Perfect Passwords to protect my WiFi. I don't blame you for not being able to type those in, and then they give us their tips for how they manage to type them in.

Leo: Oh, you're not talking about the Perfect Paper Passwords, you're talking about the Passwords page.

Steve: Correct.

Leo: That's my confusion. When you put "perfect" in there, now I'm confused. Okay.

Steve: Right. It's not the Perfect Paper Passwords, and you're right, that's just a demo.

Leo: The 64-byte crazy web key.

Steve: Yeah. I use them myself whenever I just need something. I'll just go get one there because it's safe and guaranteed to be unique, blah blah blah.

Leo: Boy, no wonder. That's a lot of passwords.

Steve: Four million, yeah. And I just wanted to see...

Leo: Did you see, by the way, the entropy problem that - was it the Maryland State Lottery was having?

Steve: No, interesting.

Leo: They were using a random number generator. Let me make sure that this is correct.

Steve: Ooh, but it was a bad random number generator?

Leo: It was. And they got a - it was one of those power - the short ones, like the numbers game? And they got 7077, 7707, 7007.

Steve: Ooh.

Leo: It was kind of a problem. Kind of a little bit of a problem. I have to - let me not libel the Maryland State Lottery. I'll try to find out which lottery it was. But there was a state lottery random number generator that didn't work so well.

Steve: I did want to give our listeners a little bit of heads-up on my early experience with my monster quad core workstation. I wanted to say that the quad core seems to be a total waste as a personal workstation. If you're going to- unless you would be using that machine for media compression. It is just a killer solution for media compression. Oh, my god, Leo, is it fast. It's just unbelievably good. I mean, it...

Leo: Well, because you're using a tool, whatever it is that you're using to do the compression, that is multithreaded.

Steve: Well, yes, it's multithreaded, and it's multicore. I mean, it recognizes what you've got, and it really uses it.

Leo: But you won't see any benefit - that's the real issue is I think a lot of programs aren't multicore aware. They're not SMP aware. So you won't see any benefit.

Steve: Well, and that's just it. I'm looking at this thinking - and I wanted to give our listeners a heads-up. My feeling is, if I knew now what I knew - if I knew then what I know now I would have gone with a state-of-the-art, top-of-the-line, single-core, maybe dual, but go for the 3.something gig rather than slowing down in order to get four cores. Because I rarely see more than one ever in use. It just doesn't happen in a personal workstation environment.

Leo: Well, having had much experience with this, I can tell you it really depends on how you use your computer. If you don't have many - it works very well if you have multiple programs running simultaneously. And your OS is smart. You're using Windows. If the OS is smart enough, it will divide those tasks up. So I have many - I use a quad-core processor also, dual Xeons. And I run a CPU, as I'm sure you are, a CPU monitor. So I see when the cores are used. And you're right, if you're just, you know, if you're using one or two programs at a time, and they're not multithreaded, you're not going to see much benefit. But if you're using a multithreaded program, or you're using - your OS is smart enough to divide tasks among separate programs, then you'll see a benefit.

Steve: Well, but I guess, I mean, most Windows programs, and I'm sure Mac is the same way, they sit there in their idle loop waiting for input.

Leo: Right.

Steve: And so I guess in my own personal use - I'm reading email, I'm writing code, I'm in an editor, I'm browsing the web, I mean, I'm not - there's nothing I'm doing where programs are, like, really busy all the time doing things behind the scenes. The only thing I can really think of would be image compression, would be big media file compression. And there you just sort of want to go away and not try to use your system at the same time.

Leo: I have a few programs that will use all four processors pretty heavily. So it really, it does, it's totally dependent on your usage. But occasionally I will peg them. Right now they're not. They're all sitting there. I've got Skype running, I've got a browser running, I've got email running. Nothing's happening. Two percent, all across the board.

Steve: Well, I did have a nice little note from Matt Ludlum in Weybridge, London, who made a comment from our prior Q&A where there was a - we mentioned a Firefox add-in which if you - it allowed you to hover over the Submit buttons of forms, and it would pop up a little window showing you the URL. And he mentioned something that I had not noticed before, and that is that IE7 has that functionality built in. I went back and looked at IE6. I've now got VMware installed on this workstation, so I can run multiple versions of Windows, each with their own version of IE.

Leo: There, by the way, would be a good multicore use.

Steve: Yes, and actually it is. I did notice that when those were doing something, VMware was good about borrowing some more of the system resources. Of course, there what I really wish is that I could put more than 4GB in a 32-bit machine. Because, boy, RAM is so cheap now, and VM as we know, virtual machines just burn memory because they all like to have their own. Anyway, Matt's point was very well taken. IE7 you're able to hover over Submit buttons. And down where it normally shows you the linked text when you're hovering over normal links, it also now does it for Submit buttons, which I thought was very nice. Yeah, so you're able to check the security state before you click.

And then finally Elaine, our illustrious transcriber, she sent me a note shortly after that last round of many podcasts you and I recorded before you were taken off to Australia, and she said - and I have a little note here that I left to myself so I wouldn't forget to mention it. I said, "Elaine reports," quote, "the Carlsbad Caverns are in New Mexico."

Leo: Yeah.

Steve: And I was saying that I was thinking they were in Carlsbad, California, and we said something to that effect a couple podcasts ago.

Leo: I wasn't listening, or I would have mentioned that.

Steve: So she said, "Carlsbad, California is the home of the expensive spas, not creepy bats."

Leo: Right.

Steve: And she said, "I have a feeling you may hear from some of your less geographically challenged listeners on this." And I wrote back and said, "Yeah, I don't get out much."

Leo: Too many people.

Steve: Yeah. Wrong Carlsbad.

Leo: Carlsbad, New Mexico. What'd she say, New Mexico?

Steve: Carlsbad, New Mexico.

Leo: You know, we went in caverns - a cavern, the Marakoopa Caves, in Tasmania, that were quite awesome. I have a few pictures of that. Quite amazing.

Steve: Cool.

Leo: Caves are fun. But to the bat cave, Robin. Actually, before we do that - we do have listener feedback. This is Episode 39 of our listener feedback.

Steve: And Episode 140...

Leo: I was wondering when you'd mention that, yeah.

Steve: Ah, baby.

Leo: You've passed TWiT.

Steve: Yeah, passed TWiT, and we've got 10 to go till we're at the magic 150, which is a nice round number. Well, I have an interesting note that I wanted to share because it sort of describes an interesting journey that somebody had with TrueCrypt and Macs and PCs and SpinRite. This is Jonathan Schmidt, from Ohio, who writes, he says, "SpinRite saved my Vista Mac." And he says, "Hi, Steve. I've listened to you on Security Now! since the beginning, and I really appreciate all you do for the Internet community. You, too, Leo. In fact, because I love your podcast and the others at the TWiT network, I signed up for the automatic PayPal payments..."

Leo: Thank you.

Steve: "...and send $5 a month. I know it's not much" - well, I think that's pretty good, actually.

Leo: No, it's more than I even ask, so thank you, yeah.

Steve: Yeah. He says, "I know it's not much, but I hope it helps." It certainly does.

Leo: It does, yeah.

Steve: "In support of you and your efforts, I also purchased a copy of SpinRite, which is the reason for this email. In the recent episodes of Security Now! concerning whole-drive encryption, I got the itch to try it out. I downloaded the latest version of TrueCrypt, 5.1a, to try on my MacBook Pro running Vista via Boot Camp." So that's the dual-boot mode. He says, "Incidentally, 5.1a's bootloader now supports Intel Macs." So that's cool. He says, "I first ran into a problem when trying to encrypt my Vista partition. Apparently TrueCrypt doesn't like the partition table set up by OS X and Boot Camp. It said that there was not enough room available on the drive to install itself. So I did some digging and found out that you can actually install Vista directly on a Mac without OS X. All you have to do is boot from the Vista install disk and remove all partitions, and then install Vista just like any other Intel machine, no hacking required."

Leo: Right.

Steve: "I proceeded to go through the Vista install just like any other Intel-based machine."

Leo: There's only one problem, which is you don't have the OS X drivers, but we'll address that.

Steve: Ah, right. He says, "I proceeded to go through the Vista install just like any other Intel-based machine. After the clean install I ran TrueCrypt 5.1a and started the whole drive encryption with three-pass wipe to be safe." Wow. And he says, "Voila, everything started working. TrueCrypt began encrypting the whole drive. Unfortunately, when it got to about 78 percent of the way through, TrueCrypt gave me a CRC error, indicating that it was a problem with my drive. Although I tried several times to continue, TrueCrypt gave me the error every time. I thought, crap. I went through all this work only to have a drive error. Now what do I do? Then I thought of SpinRite. Of course, SpinRite doesn't run on a Mac. {Hint, hint.)"

He says, "So I pulled out the drive and stuck it in my regular Intel desktop. I ran my copy of SpinRite overnight at Level 4. Sure enough, this morning when I came down, SpinRite reported that it found and fixed an unrecoverable error. I popped the drive back into my MacBook Pro and powered it up, and TrueCrypt prompted me to pick up where it left off. I did so, and it did not give me an error this time. It is continuing to encrypt the rest of the drive as I type. Pretty amazing. SpinRite saved my Vista Mac. I always knew that SpinRite would come in handy someday. Thanks so much, and keep up the great work at GRC and Security Now!. Sincerely, Jonathan Schmidt."

Leo: Excellent.

Steve: And, let's see, there was one thing I noted. Oh, when he said that TrueCrypt didn't like something about the partition, I'll bet I know what that is. I'll bet that when the Mac was repartitioning the drive and set up a boot sector on the Vista partition, I'll bet that it didn't zero out the rest of the track, and that when TrueCrypt, being as careful as it is, looked at that first track of the Vista partition, it saw debris there, whatever happened to be there before. But it just assumed that it was something horrible like Macromedia junk. Or it saw that apparently...

Leo: The copy protection that Macromedia uses, right.

Steve: Exactly. I mean, it probably saw there was something there. Normally that track is all zeroes. And so I'm sure the TrueCrypt guys take a look at that and make sure that it's zeroes before they install themselves. And they probably saw that it wasn't and said, ooh, you know, whatever's there, we don't want to hurt it, and we can't go in here. So that may very well have been what caused the problem for him.

Leo: Yeah, it's very careful about these things, which is as it should be.

Steve: Absolutely, yup.

Leo: And it sounds like he was installing, instead of running Mac at all, he was installing, I mean, I'm not sure, maybe I misunderstood him, but...

Steve: I think he wiped Mac OS X off the machine completely.

Leo: You do want to use Boot Camp. And the main reason that this procedure, I wouldn't recommend this procedure, is that one of the things Boot Camp does is make a CD with Vista drivers for the Macintosh hardware. Otherwise you don't have drivers, specific drivers for the hardware. So do at least do that, create that CD. Then you can wipe it.

Steve: Right. And then you use that to install the Mac drivers for the hardware.

Leo: Right. So what normally happens with Boot Camp is it does the partitioning, then you install Windows. But before it does that it makes a CD of drivers. After you install Windows, you put the CD in, you install the Mac hardware-specific drivers. So if you don't use Boot Camp at all, you won't get that disk, I guess is what I'm saying. But you don't have to use Boot Camp, of course not. I don't know why you'd buy a Mac to run Vista only, but you can, if that's what you choose. Shall we get to the questions? We have a lot.

Steve: Yup, let's do it.

Leo: Let's do it. Let's read those questions. What did I do with them? I put them away.

Steve: That's your announcer voice.

Leo: Let's go to the questions. As soon as Leo finds the questions. I have about 80 windows open at the same time here. You know, I think I've put them away.

Steve: No wonder you have quad core.

Leo: You see, now you know why I need it.

Steve: Actually I sort of shut things down and keep the system from getting...

Leo: Well, I do. What I really don't want it to be doing is running Internet processes. There it is. But I do have other processes running because that's one of the reasons you might want a quad core is for the headroom. I guess that's what I would say is it's all about headroom. So that when you need additional cores - like I'm seeing right now as I look at my graph that there have been a few spikes during this. But I don't have to worry because I know that Skype running in its own process there, in its own processor, is not going to run out of juice. So it's about headroom. For me, anyway. And you're right. And I have several quad core machines. And it may be that Windows doesn't do as good a job of dividing tasks up. I don't know.

Steve: Yeah. I just, again, I'm happy saying to our listeners, you know, maybe just go with a fast dual core or a single-core hyperthreaded processor, and go for more speed. I think for typical users, unless you're into media compression, or literally having things really doing number crunching at the same time, I think it was a waste of money.

Leo: Really.

Steve: Yeah.

Leo: Well, see, the interesting thing is that basically you're not getting a choice these days because Intel has decided that everything's going to be at least dual core.

Steve: Right, and in which case you've got multiple cores, and that's a good thing. But for me, given what I understand now, I don't think I'd do it again. I think I'm seeing these are underutilized. And I'd rather have more speed and fewer cores.

Leo: Well, I don't even know if you get that choice anymore. But...

Steve: Right.

Leo: Okay. Message to Intel. You might want to look at an AMD chip. Chris Clark from Western Australia's Perth - I didn't make it to Perth. Apparently nobody makes it to Perth because it's in Western Australia, far away from the rest of the continent, whatever you call that. It's not a continent. It's a big island. Let's his fingers do the walking: Hi, Steve and Leo. I've been wondering what you'd think of a technique I've used for a long time to create passwords that are easy to use despite their apparent complexity. He says he uses muscle memory. I create a simple geometric pattern that moves my fingers around the keyboard in a way that makes the output look like complete gibberish. The pattern is simple; the resulting password is not.

Basic example, start at the bottom left-most key of the U.S. QWERTY keyboard, hit the first four keys - bottom left-most key. So that'd be zxcv. Move up a row and so on, be asdf, then qwer. And then he adds 1234. After a little practice the keystrokes become second nature, can be tapped out in a second or two. That's true. That would be easy. Even though it's a nice long password. As an added bonus, since you're memorizing the pattern, not the text, your eyes and brain never really learn the password. I don't know why that's a bonus. Only your fingers know the secret. Obviously my own finger-dance is a lot crazier. Oh, good, because I was going to say that one is probably one of the first things a hacker would guess. It's a lot crazier than the example that I gave just now. But you can see how even the simplest example could produce a longish password that looks pretty random. Maybe "looks" is the operative word there. Even if strictly speaking it's not random at all, would a password like this stand up to a sophisticated brute-force attack?

Steve: Well, it was an interesting question. My concern is that it sort of sounds like this methodology results in using a single password a lot. And we know that it's generally a bad idea to use one password as, like, your password, and reuse it wherever you go. We understand it's much more secure to somehow have a system that generates different passwords for different uses. So I wanted to throw the question in because there's some interesting ideas here. For example, if you had an algorithm that was more sophisticated, for example, like the first letter of the website is your starting key on the keyboard, and then you do something. You go up if you can; otherwise you go down. You go right if you can; otherwise you go left. But, I mean, you could imagine an algorithm where starting at any given location on the keyboard you could do something consistent that would generate a password which is very unlikely to be found in any kind of a brute force.

First of all, when he talks about a sophisticated brute-force attack, you'd first be exhausting your dictionary, and then combinations of words, it's very unlikely that an attacker would start attacking the physical location of keys on a keyboard. And frankly, there's so many possible algorithms. Of course in a typical keyboard you've also got diagonals, not just up and down. You've got up left, up right, down left, down right. I could see coming up with an algorithm that could, given any starting place, could consistently generate an interesting password. So there's something to it.

Leo: I'd be careful about some of the more obvious combinations, though. I think that probably brute-force attacks include things, look for things like ASDF and go, oh, keyboard...

Steve: Oh, yeah, and doing QWERTY, you want to stay away from that one because it's probably in the dictionary.

Leo: It wouldn't be so difficult, frankly, to write a brute-force attacker that includes some of the more obvious keyboard algorithms. I'd be careful about that.

Steve: Yup, absolutely right.

Leo: Someone who asked us not to use his name from Tennessee wrote he strongly disagrees with advice about old operating systems: Steve, I usually agree with what you have to say, but I thought your advice about old operating systems on Episode 136 was way off base. In many cases the vulnerabilities that are found in newer operating systems exist in the old ones. Not always, of course. For example, wasn't the animated cursor exploit one that went way back? And some of the most recent networking holes went way back into old operating systems. Well, we just talked about WMF, which goes way back.

Steve: But not that far.

Leo: Not that far. And some of the most recent network - oh, yeah. I'm aware of at least one incident involving an employer in my area where systems were compromised and information stolen. The hack involved older operating systems, past end of life, that were on a network. Employees received malware in an email that infected their systems. The malware then went through a series of known exploits for unpatched and past-end-of-life operating systems. I'd argue that the only safe way is to run a system with an OS past end of life is, A, to ensure it's never connected to an Internet-connected computer, air gapped - that's a good word, I like that, air gapped - or that it is behind a well-configured external firewall that only permits absolutely necessary and well-monitored traffic through the outdated system. That makes sense. I'd agree with him.

Steve: Yeah, and I have to, I mean, I still like the idea of using 9x-era machines, 98 2nd Edition, for their relative invulnerability. They have, for example - none of the exploits in today's or this month's Patch Tuesday affects those older machines, for example.

Leo: Well, you wouldn't expect new patches for the older machines. They don't even patch them anymore.

Steve: No, no, but I mean none of the vulnerabilities exist in the old - none of this month's vulnerabilities affect 9x-class OSes.

Leo: We know that for a fact?

Steve: Yeah.

Leo: Or just that they didn't patch them?

Steve: No no no, they're not vulnerable back there to that specific one. Just like the Windows Metafile.

Leo: I guess his point is that you can't always say that they don't. They may not patch them doesn't mean that they're not vulnerable.

Steve: That's very true. And so I don't...

Leo: That's the problem is that they don't patch them.

Steve: Yup, it's past end of life. And so I just - I thought his opinion deserved being aired.

Leo: I'm kind of on his side. Windows 9x, you're just - I think he's right. If it's going to be in any way connected to a network machine, you're vulnerable, even if it's not on the 'Net itself.

Steve: Right.

Leo: I mean, you would agree with that; right? Certainly there are - he's cited a case where there are exploits that look for, particularly for older machines. I think a lot - I'm willing to be that a lot of the reason things like Sasser are still on the 'Net is because there's a bunch of dusty old Windows 95 or 98 machines running in closets at enterprises that don't bother to look at them ever because they still work, they do what they were supposed to do, so they don't patch them, and they don't fix them, they don't keep an eye on them. And they're out there chugging out viruses all the time. I think that's, I mean, that's why Sasser is still around.

Eric, listening from Sanford, North Carolina, wants greater security and less service. Oh. Steve and Leo, first, thank you for providing a valuable service to computer users everywhere. I'm an avid listener who never misses an episode. And thanks for making me look smart to my friends and family while helping them with their computer problems. Secondly, you and your feedback listeners have mentioned turning off unnecessary services and processes in Windows 2000 and XP. Could you do an episode or a feedback question detailing how to slim down Windows and slam some vulnerability doors shut? My identity, my processor, and my limited RAM thank you in advance. I do what you recommend. I keep up to date with Microsoft OS patches, use the Komodo firewall, run spyware and antivirus weekly, use a NAT router at home. I do my best not to contract a CTD - I like that, Computer Transmitted Disease - by traveling to only a few favorite sites. Wow, this poor guy. But I do take a Sunday drive on the 'Net occasionally. Good. Third-party ad banners are a cause for concern, for one. Thanks. And that's just what we talked about with this WMF vulnerability.

Steve: Yeah. I wanted to respond to a couple of points that Eric made. First of all, I do think that's a great subject for a show topic. And I've got it on my...

Leo: Turning off services, yeah.

Steve: Yeah, well, essentially, doing the research to see specifically which things can clearly and safely be turned off. I mean, I know we were talking about the Black Viper site, which gives a lot of advice, just the other day. In fact, it was when I was setting up some of these machines in a VMware environment. I noticed that in XP the wireless zero config service is there and running. And it just bugs me because, I mean, this is a workstation with no WiFi. And Windows could know that I don't have any wireless stuff.

Leo: You mean it actually starts up even though there's no wireless adapter?

Steve: Yes, it's there and running, just like oh, you know, we're going to...

Leo: That's crazy.

Steve: ...give you zero config. Well, zero config is right because I've got nothing to configure.

Leo: Zero WiFi.

Steve: Oh, my god. And so in my case, for example, I use static IPs, but there's the DHCP client just sitting there running, waiting to give me an automatic IP. It's like, well, don't need that, either. And so there are just so many services that end up running. And when I'm done trimming them down, I've just got this short little list. And in this case of this XP that I set up, I just was doing it yesterday, I was running in 66MB of RAM. And it boots instantly. And it's just such a small footprint. And especially, for example, in a VM environment where you want to minimize the RAM impact of running a virtual machine. It really makes sense to pare down the RAM footprint of your virtual machines because it leaves more RAM for everybody else, for the external systems. So that's one point.

The other is I've been big in the past on turning off services for security vulnerability reasons. So, for example, Unplug n' Pray turns off the SSDP enumerator. Shoot The Messenger turns off the Messenger process, which most people don't need. DCOMbobulator shuts down the DCOM service. And so those were things that were done in a pre-Service Pack 2 of XP mode. Specifically, they were things that really made sense when you were not behind a firewall. The world really did change with Service Pack 2. And I wanted to bring up that point, that having these services back behind a firewall is far less dangerous than it used to be. So that's one point.

On the other hand, the more things you've got running, the more opportunities there are for local exploits. We always, you know, we hear about privilege elevation attacks, where you're running in a non-privileged account, but some malware gets in. And you're thinking, oh, well, I'm safer because I'm a non-admin user. But there are privilege elevation attacks, typically that use some sort of kernel exploit in order to get advanced privileges from a non-privileged account. And that's typically done by leveraging some services that are running, which a non-privileged account is not able to start. So they have to be there first, and then a non-privileged account is able to take them over. So again, it's another good reason, even if you're behind a firewall and a NAT router, why having less services, having fewer less services - having fewer services...

Leo: Yes.

Steve: ...really does make sense. So I think it's a great topic for an episode, and we're going to go there.

Leo: And as far as I know, I mean, I think Black Viper's done a really good job. You've recommended them in the past, I mean, that's a great place to go. I don't know if you can much improve on it, frankly. He's, it looks like, done trial-and-error on every single service, one by one.

Curtis Wyatt writes from Las Cruces, New Mexico. Hello, Steve. Oh, this is a good one for me. I'm considering online bill paying. I've been using this for years. What do you think of this? Is it safe? Are there any drawbacks to giving my electric, gas, and cell phone company bank information over to these guys? What do you think?

Steve: Well, I think it's the typical tradeoff between security and convenience. Now, in a non-online mode, you're mailing your check to these people. So they've got your banking information. I mean, they've got your bank account number, and they know who you are, they're matching it up with your account and so forth.

Leo: But only the particular person you're paying the bill to.

Steve: Oh, exactly. Exactly. But...

Leo: So you're adding one additional person who knows this information, that's this third-party bill payer.

Steve: Okay. And so I guess my point is that any time you're aggregating information in a database, and your data is there with a whole bunch of other people's, there's a single point of vulnerability, which is what has all the bad guys salivating. When we were talking earlier about this month's new metafile exploit, one of the things that I heard a lot about last week at RSA is the increased prevalence of targeted attacks where specific executives or employees of companies that the bad guys want to get into, they'll send email that is about their organization or about their company or about their job because the email that is focused on that company is able to have a much greater penetration rate if it knows who it's going to instead of just random blanketed spam going out everywhere, talking about somebody in Jakarta who's got money that he needs to transfer into the U.S. if you'll give them a hand. And so this kind of attack is the sort of thing which is now causing financial institutions a great headache. And again, so I would say relative to online bill paying, it's like, well, it's a tremendous convenience. And you probably trade off a little bit of security for it. But lots of people do it.

Leo: And you could make a case that you trade less security since you aren't using, well, okay, actually here's the dirty little secret of it. You may be using the mail after all. Because a lot of companies don't accept electronic funds transfers from the bill pay service. In many cases what the bill pay service does is print a check, put it in an envelope, and mail it. So I guess in that case you're not saving any security. I guess the ones that are using electronic funds transfer, by not using the mail you might be getting some security; right? I mean, mailing a check does expose you.

Steve: Well, now, I assumed that he was talking about setting up an account with the individual organizations, as opposed to...

Leo: Well, that would be safe. Well, I use a central clearinghouse. I use Intuit's Paytrust, and have for years. It was originally PayMyBills. Paytrust bought them, then Intuit bought Paytrust. So there's a flaw right there. God knows how many different companies, at least three, have owned my information. There's also the potential risk, I've heard it said, this may be completely apocryphal, but that some data entry is done by prisoners. And who knows if the bill pay services use that or not. But I suspect that some of them do.

Steve: When they're not making license plates.

Leo: Right. So there is some data entry, quite a bit of data entry involved. I mean, no machine can look at a bill, figure out where it goes, and make sure the amounts are correct and everything. Some human's reviewing that, and you trust that human. So, I mean, I think there are some real security issues involved.

Steve: Well, and my concern, any time I hear someone talking about my bank account information, as I understand it, if electronic funds transfer is used to suck money out of my account, it's gone. I mean, there's no indemnification against fraudulent transfers out of a bank account.

Leo: That can't be true.

Steve: I don't know. Because I know that I've talked to the FBI when we were - I had some conversations with some of my local friends years and years ago when I was setting up my eCommerce system, just sort of asking them, so, you know, what do you - what's to watch for? And they had some, I mean, some real stories about people who innocently got involved with eCommerce, had their merchant accounts set up incorrectly so that there wasn't a limit on the amount that could be transferred, and they lost their entire balance, and there is no recourse. It was gone.

Leo: No, no, that is old information. They did change the laws on EFTs. And so there is a limit and indemnification. The law was changed a few years ago.

Steve: Oh, good.

Leo: However, I'm not saying that that protects you. You should probably check with your bank and see what their policy is. But certainly on - this was an issue with using an ATM card instead of using a credit card, was originally when you used an - until a few years ago, when you used an ATM card you didn't have the same protections.

Steve: A debit card, right.

Leo: A debit card. So as a result you could, in fact, if you lost your debit card information, really be drained. But that's - but there was a law passed to change that. Now, I would check with your bank about EFT and what kind of indemnification you have. So you're right. Now, maybe it is then a little bit safer to do what I do, which is use a single third party, because they handle the transaction. So only they know my bank account information. And then they pass along - of course if you're sending somebody a check, they've got the bank account information.

Steve: Exactly.

Leo: That's kind of the weak link in all of this is that so many merchants don't accept the kind of - it would be nice, the whole idea of online bill pay would be this kind of everything's done electronically, but it isn't in many cases. Your bank may offer this. Intuit, I trust Intuit, and certainly their privacy policy says - it's very clear. Their privacy, we do not sell or rent your personal information to anyone. We do not share your personal information with anyone outside of Intuit.

Steve: We'll just sell the whole company to somebody else when it's no longer something that we want.

Leo: Right, it's been - this is the third owner now of all of my personal information. However, having said that, I have been doing it for over a decade and have never had a problem. But you're right, I mean, what we cover on this show is theoretical problems, not - we cover what could go wrong.

Steve: Well, yes. We're heavy on the technology. I should mention that while you were gone in Australia, Leo, I lost my credit card. That is, I lost access to it. By that I mean it escaped on the Internet. I got a call from a robot.

Leo: Really.

Steve: Yes. It's the third time this has happened to me. I use it extensively all over the 'Net. And I got a call from a robot that said, "Please hold for a security consultant." And I got a gal on the phone who said, "Were you buying anything in France last night when you were probably asleep?" I said no. And she said, well, the first charge was for a dollar. And then there was a charge for 1,500 and some dollars for some sort of sports boutique, whatever that is, in France. And I said no, that's not me. And then there was a third try. All three were caught and blocked by their automated security since I had no past of any kind of transactions like that. And I said, okay, cancel the card. So we canceled that number and issued a new one. And...

Leo: Happened to me, too. I bought something from an Argentinean company. And shortly thereafter I got a $7,000 charge. But that's the good thing is the credit card companies call you.

Steve: Yes, exactly. And anything fraudulent they will take off your bill. And as it happens, I did go over my statement, and nothing got past them. Apparently that little

$1 charge was their test charge to see whether the number and credit card information, which they had clearly received from someone, was valid. And it was.

Leo: So do check your statements. But, you know, the banks use interesting - actually this would be a great subject for a podcast at some point. They use business intelligence software to - remember, there's billions of transactions every day. How do they find out what's a weird transaction? They use software to monitor your kind of patterns. And anything out of the ordinary this software flags. And this is a very effective software. It seems to catch most of this stuff.

Steve: Well, and in fact in one case it's a little too effective. I've never been able to purchase gas with this credit card.

Leo: Sorry.

Steve: No, it shuts it down every time I use it to fill my tank. Then I'll be at a restaurant, and they'll say, really sorry, Steve, but this card is - I was like, what. Anyway, so I'll call them. It happened, like, three or four times in a row. And finally I said to the person, look, every time I buy gas with this card you guys shut it down. They said, well, that's because, unfortunately, that's what the bad guys do. They buy gas with a credit card because there's no attendant present. They're next to their car. They can make a fast getaway if the card is declined. And so it's a simple way for them, in relative safety, to check to see if the card is good and they can get away with it.

Leo: Oh, that's interesting.

Steve: And so consequently, I mean, I have another card that isn't so particular, which is the card I deliberately pull out when I want to buy gas because it stays alive afterwards; whereas this other card - and it's these people who caught this stuff happening in France, it's like I'm glad for that. I'm glad they, I mean, I'm willing to make the tradeoff. I won't buy gas if they'll shut down any fraudulent purchases because their software is so particular.

Leo: Well, it should be. But that's a little weird. I haven't had that trouble. Although when I tried to use an ATM card in Canada at one point it said no. I called the bank. And all you do is you call the bank and you say, you know, I am in Canada. And they said, okay, good, we were just, you know, we were a little nervous. And from then after I never had trouble using my ATM card in Canada because they know, oh, he goes to Canada every few months.

Steve: Yeah, you had your Canada bit set.

Leo: Right. So I'm glad they do that. Let's see. Where am I? John, listening from an undisclosed location, wishes he had a Wayback Machine. Well, who doesn't? Hi, Steve. I really, really like the Security Now! podcasts, so I ended up subscribing to

Security Now!. By the way, that's free. I hate that word "subscribe." That's what Apple uses on iTunes. And I think it confuses people because, first of all, you're in the iTunes store, and then you press a button that says "Subscribe." It sounds like you're going to be charged something. All our shows are free. If you want to donate, that's fine, that's completely optional. You go to TWiT.tv, and you can press the Donation button. It certainly does help, especially now that we're starting to add this video. They're spending a lot of money on things like lights.

But he says, I ended up subscribing - for free, I add - to Security Now! and have downloaded all the Security Now! podcasts that iTunes has to offer. Well, that won't be many because we only put 20 up at a time because we don't want that feed to get so long.

Steve: Ah. Thus his need for the Wayback Machine.

Leo: I would like to get the rest of the earlier episodes so I have all of them, but I saw on your website information that stated, "You may download and listen to selected episodes or subscribe to the ongoing series as an RSS podcast." I've already subscribed on iTunes. Does that mean I cannot download the episodes? See, this drives me crazy because it really is this impression that in some way you're limited. There's no limit. I wanted to ask, don't want to violate your policy. No, look. They're free. Get as many as you want. The easiest way to get them all is to go to TWiT.tv/sn, if you want to go directly, and that's the Security Now! page. TWiT.tv/sn. You know, I realize, Steve, that a lot of people don't even know that TWiT.tv exists, that there's a website.

Steve: Really. Oh, because they found us through iTunes, and...

Leo: Right.

Steve: Right, right, right.

Leo: And so they don't see the donation stuff. People are constantly surprised. Oh, you accept donations? Yes. Oh, you can get all the files? Yes. You just go to TWiT.tv/sn, and you can go one by one through every - every episode's on there. In fact, I'll tell you a little shortcut. If you go to TWiT.tv/sn1, you'll get the first episode; sn2, you'll get the second episode; sn3. Unless we make a mistake, sometimes we forget to do that, and if we have, please let us know. But all 140 episodes are available there. Now, how do you download them all at once? There isn't a way to subscribe. The reason is you don't want a podcast feed that has 140 shows in it. It would be too large, and it would cost us a lot of money because we pay the bandwidth for the downloads of the feed information.

Steve: Good point, and people might just be downloading them when they really aren't going to listen to them all or don't really need...

Leo: But they do, they do. I mean, that's the way iTunes and everything else works is it downloads the feed. Like every hour it checks the feed. So if your feed size - our feed sizes are already pretty big because we put 20 episodes, that's about 50K. It would be a MB or more if you included all 140 episodes. That means every hour everybody who subscribes would download a megabyte file. Do the math. I can't afford it. We used to do that.

Steve: Oh, I was going to say, I think your math is wrong. But you're not talking about the content, you're talking about...

Leo: Just the feed.

Steve: The RSS, the XML definition file.

Leo: Just the feed, exactly. So we keep the feed to 20 episodes of any, of all of our shows, the most recent 20 episodes. That's 20 weeks, goes back five months. But if you want to go back farther, then you have - I don't know of an easy way to do this. Somebody maybe want to write a script, they can, to download all the episodes. That's actually not a bad idea. I can probably put something like that out that would just do it all. But not through iTunes because iTunes is dumb, dumb. I guess you know what I could do is create - no, I don't want to do that because people will subscribe to it, and then I'll get hit by it. Really it's very expensive. Bandwidth is not cheap. The good news is, of course, thank goodness, AOL pays for the bandwidth for the show. But they do not supply the feeds. And even the feed itself can add up. So go to Security Now!'s page on TWiT.tv. I imagine, Steve, you have every episode at GRC.com, too.

Steve: I have them all there, GRC.com/securitynow.

Leo: So you can go there, as well, doesn't matter. Same to me. Now that Steve, you go through - you do the Podtrac link; right?

Steve: Yup.

Leo: So that's the main thing is that we get counted - actually it doesn't even matter. For any episode that's older than a month we don't get paid anyway. So forget that. We only get, you know, we get paid by our advertisers. Actually Astaro just pays us a flat fee. Audible pays for the number of downloads. And but they only count the first month's worth, which is kind of annoying since many people like our good friend John like to listen to old episodes. You want me to read this next name? Hkan Lindqvist.

Steve: That's very good. I don't know how to pronounce it.

Leo: Hkan Lindqvist - I don't know what that first name is, I don't know what that is - in Sweden deeply gets the point of HTTPS security. I would just like to emphasize - by the way, we love it, we have many listeners in Sweden, in Scandinavia in general. We love that. And Australia, I found out we have tons of listeners in Australia.

Steve: We have a ton, yes. In fact, a bunch of them are here in our questions, as a matter of fact.

Leo: That's one of the things that's most fun about podcasting, or netcasting as I always call it, is its international scope. I would just like to emphasize something regarding banks, et cetera, that have their login form on a plain HTTP page and a statement that says "Your login information will be submitted securely" or something to that regard, and maybe even redirect people back to that insecure page if they attempt to switch that page to HTTPS manually. Ooh, that's annoying. It is actually not just a lack of fuzzy warm feelings for the visitor, but a catastrophe waiting to happen from a security perspective. Even if a form is submitted over HTTPS, it really does show that their security department doesn't understand HTTPS at all, which makes it worrisome that they have an online presence at all. Wait a minute. No, I think he's going a little far here.

First of all, any phisher that makes a look-alike page will have no trouble emulating the login form. Of course not, they just copy and paste the source. How hard is it to write "Your login information will be submitted securely"? What kind of security measure is that? What it should say is something like "Never ever enter your login information unless your browser shows the padlock, and ideally verify the certificate chain." Actually he's got a good point. That's a very good...

Steve: Yeah, he really does.

Leo: I mean, everybody does this. I'm not sure why they have the insecure page that goes to a secure page. But he says encryption's only half the point of HTTPS. HTTPS is designed not only to encrypt data, but actually show who you are communicating with so far as the trusted root authority knows, which at least much more than the web surfer can know. So if you have the actual form on an HTTPS page, it not only gives the user a warm and fuzzy feeling, it's actually the only way for the user to be able to check who they're communicating with. The browser only can show that after establishing the connection, HTTPS connection. If your bank misuses security technology that badly, it may be worth switching banks. Except they all do. Even Amazon does that. He's got a very good point, though.

Steve: Yeah.

Leo: It's more than just security, it's certification.

Steve: It's authentication. Yes, you are, I mean, as we've talked about, we have root authorities who go to some length of trouble. And one of the things we're going to talk about next week, I attended a really interesting discussion last week at the RSA conference about these EV, the extended validation certificates. And I've got a much better sense today for why it makes sense. I was surprised that only 5,000 of them have been purchased. There's only 5,000 merchants who are using them. GRC may become 5,001, although they are expensive, and it annoys - that still annoys me. But I can understand, given what they do, that they are, they're earning more money than they are with the regular certificates, that's for sure. But the point is that HTTPS does provide you with authentication of the identity of the far end. And we've talked often about right-clicking on the page, looking under the page properties and view the certificate and look at the chain of trust back to who trusted the site you're at...

Leo: Why wouldn't you do that all the time? Is it because it's costly in some way?

Steve: I'm sorry?

Leo: Why wouldn't you, I mean, for instance, now, Amazon, I slandered them a little bit. When I get to the login page it is an HTTPS page. But as I'm shopping, it's an HTTP page. Why isn't it always HTTPS?

Steve: That's a very good point. I think it ought to be. Once upon a time, I mean, and we're at 15 years ago where we had 8MHz PCs, you could argue that it was too expensive to establish the HTTPS, that is, the SSL or TLS, which are sort of the same thing, handshake. There is that cryptographic setup process which is a little expensive. But that's just gone away now. Servers are so powerful, individual end-user machines are so powerful, and we went from HTTP 1.0 to HTTP 1.1. One of the changes is the notion of persistent connections, where a browser, instead of initiating lots of little short-lived connections, it will leave the connection up between the browser and the server as you move from page to page. So that helps in many ways, one being that you're not needing to create individual secure connections. So that dramatically lowers the burden. I've been giving serious thought to just switching all of GRC over to secure connections. There just isn't a reason not to. I think once upon a time Google was not indexing secure pages, and that's long since changed. So they're just - I don't see any reason not to leave it secure.

Leo: Yeah. So, I mean, yeah. Let's do that, folks.

Steve: And so you have persistent authentication, and...

Leo: Then you'd know, yeah.

Steve: And you're actually snoop-proof, the entire dialogue is snoop-proof. You know, and as it is now, I enforce SSL connections on sensitive pages. The Security Now! entry page is that way. The Perfect Passwords, of course, is shielded that way, and these Perfect Paper Passwords when it's displaying its stuff. So it just makes sense to leave it secure the entire time. I don't think there would be a downside to it.

Leo: Yeah. That's, well, okay.

Steve: Anyway, I really liked...

Leo: Good for you.

Steve: ...liked his observation that it was - it's about authentication. And banks that don't put you on a secure page for filling in the data don't allow you to verify their identity. Now, that's one of the things that we'll talk about next week that the EV certificate changes, is it's in the user interface, is that you can see the name of the site's certificate up in the browser's UI, which over time I think people are going to really get, I mean, basically it's like having to right-click on the page, do properties, check the certificate, and see who the certificate owner is. It does all that work for you and just sticks it up there in the user interface. I think it's a great idea.

Leo: I have to say, I'm looking at my bank page, which as I remember at least at one time didn't do this. Bank of America does do it. You're HTTPS from the moment you're there. So that's...

Steve: Good. They got a clue.

Leo: They got a clue. And I don't want to slander anybody. So Amazon does do that when you log in, but not while you're shopping. And that would be a nice - why not just do it all the time?

Steve: Well, and a perfect example is Google Mail. We've talked about this. If you go into http://mail.google.com, it takes you through security in order to log in, and then drops you back out to nonsecure pages. If you go in as https://mail.google.com, then you'll be secure and you'll stay secure. But they ought to always - since you had to go secure anyway to log in, it demonstrates that your client has the ability to establish an SSL connection. So why not just leave it up? I mean, why not take them into that mode?

Leo: Right. In fact I'm noticing now Bank of America has now added - oh, this is great. This is something recently. They've added this second layer, actually I guess it's third-layer authentication. You now have to use, or you can, you don't have to, have them send a code to a cell phone each time so that it's like having a dongle, a football. I like that.

Steve: It's another factor.

Leo: I like that. Added that third factor, that's great. Don't mind the inconvenience. It's nice to know that I'm the only one who can get on this system.

Moving along, Jim Busser of Vancouver, British Columbia, Canada wants to remind us of one thing: Hey, Steve. Really enjoy your and Leo's show. To supplement your answer as to whether you, and by extension anyone, should prefer to use TrueCrypt versus IronKey for protecting USB drives, a particular advantage of IronKey is its lack of need for administrative rights. Where a user must frequently access portable files from a PC over which they have no admin rights, as in many government and corporation institutions, and also the included secure surfing service from untrusted access points, really should not be overlooked. Oh, that is - I didn't realize. So in order to use TrueCrypt you have to be an admin, even to unlock.

Steve: Yes, because it needs to install - you either need to be an admin, or that system needs to have the TrueCrypt driver preinstalled by an admin. But it does this drive mapping where it creates a virtual drive, and that requires kernel-level access in order to instantiate a drive. So, I mean, and it's why I thought this was a really good point that I wanted to remind users of. I gave a presentation at our local North Orange County Computer Club two weeks ago. I loved it, it was their 32nd anniversary, but that's 1000000. So they said, well, in binary that's a big round number. So we need cake. And we need Gibson to come talk to us. And so I gave a presentation on TrueCrypt and whole-drive encryption and about the things that we've been talking about. And actually one of the ex-managers of the club came up and said, hey, Steve, I want to make sure, I'm needing to often in my environment - she was in a school administrative environment - often needing to use my USB drive on machines that are locked down with non-admin rights. And so that says I cannot use TrueCrypt. I said, unfortunately that is the case.

Leo: Excellent point. Excellent point.

Steve: Yes. And so IronKey does not suffer from that limitation. And so I think that's a very good point.

Leo: Yeah. Dan Menear in Northglenn, Colorado needs some DNS clarification. Steve, you stated you run your own OpenDNS server. If the goal is to protect you from being a victim of DNS poisoning, how do you get propagated DNS information? Excuse my ignorance on this subject, but I do not see how this would protect you. Thanks.

Steve: Well, if I said OpenDNS, then I misspoke. I just...

Leo: Because you run your own DNS server.

Steve: Yes. And so what I run, the server I run is called a "resolver." That's the technical name for what ISPs' DNS servers are, meaning that the ISP clients, that is, regular end-users, they've got their Windows set up to ask their ISP DNS server to resolve DNS names. So the client, the Windows or Mac or Linux machine, whatever, it sends the DNS query to the ISP. And then that server does what's called a "recursive lookup" because DNS is a hierarchy starting with the root and then dotcom and then - or dotorg, dotnet, dotcom and other top-level domains - going to the second level and third level and so forth. Anyway, so what I run is I run the same thing that the ISP runs. I have a little

FreeBSD box which is running BIND, v9 of BIND, which is sort of the industry-standard DNS server. And so it does - it is my resolver, meaning that it has a list of all the root name servers, and so...

Leo: So it goes right up to the top of the tree if it can't find it.

Steve: Exactly. So basically - and I gave it lots of memory, and it caches. And so it is a full resolving DNS server. So all of the machines in my network ask that machine to handle their DNS. And exactly as you say, Leo, it'll go to the root, and then it'll find the COM servers, and then it'll ask for the, like, for example, Amazon.com, and then if it needs to, www.amazon.com. And it will itself obtain the IP address. So my point is...

Leo: So the root servers would have to be poisoned for this to be a problem.

Steve: Well, what it's avoiding is it's avoiding trusting the ISP because an ISP, being an ISP, they're a big target. If their server were compromised, all of their customers, for example, if someone managed to get a bad IP for Amazon or Microsoft or eBay or something, then you would think you were at Amazon, and in your browser it would say www.amazon.com. And they would just sort of not switch you into secure mode. They just, you know, unless you were really paying attention, you might not notice that you were submitting your log-in information not on a secure page. And so it's easy to spoof browsers by poisoning DNS. And so I'm not that concerned about it, it's not the reason I'm running my own DNS. Actually I'm running my own because I don't have an ISP. I'm buying T1 service from a bulk provider, and so I have to do my own DNS. But that's the story.

Leo: Right. Makes a lot of sense. And the only thing I would say is that, if you do that, it's incumbent on you to make sure that you keep BIND patched and keep your BSD patched. BIND has had problems in the past.

Steve: BIND has had a ton of problems in the past.

Leo: Right. So you can't just turn it on and forget it if you're going to run your own DNS server.

Steve: The other minor downside, and this doesn't seem to be any problem for me, but one of the advantages of an ISP's caching resolver is that if I turn my machines on in the morning, not that I ever turn them off, but I use a machine that doesn't already have Amazon.com, the IP cached locally, it's almost certain that my ISP does because one of their other customers will have asked for it within the window of the DNS expiration. So that server immediately returns the IP that it already looked up for somebody else to me, which is a performance advantage. Whereas if I'm running my own resolver, as I am, well, it's got to go and do that work because there's nobody here but me.

Leo: And the other thing I would wonder is, certainly the folks who run the root servers would prefer that people don't do this, right, because for that very reason it could actually really increase the traffic on their servers.

Steve: It would increase the load somewhat. Although they've got long TTLs, Time To Live, on their records. So generally you've got a list of all the com, net, and org servers, and there's a ton of those. So you're really not going back to the root that often except when you need to update that master list of the top-level domains.

Leo: Well, how long is the TTL? How long is the Time To Live? When you say long, you mean a day or two, right, or...

Steve: Oh, yeah, a day or two, yeah.

Leo: So at least every few days you have to download that entire list.

Steve: Yup.

Leo: I mean, that's...

Steve: It's just like a big RSS feed.

Leo: Right, it's not insignificant. So I would imagine - obviously they don't prevent you from doing this. Anybody can set up a DNS server. But I would imagine they would probably prefer it if people stuck with the established ones rather than setting up their own.

Eric Larsen, listening from Denmark, wants guessing stopped. Well, good for him. Hi, Steve. I just installed the new version of TrueCrypt and have starting using the whole-drive encryption feature with pre-boot authenticated. TrueCrypt recommends using a password of at least 20 characters. That can be a bit difficult to remember. In a recent show you talked about IronKey and how it only allows 10 tries, thereby preventing any brute-force attacks. I like that since even a four-character password's pretty secure in a situation like that. Wouldn't it make sense to perhaps, say, limit to 50 tries with a pre-boot authenticate in TrueCrypt? No one having tried 50 wrong passwords is going to get it right away, or going to get it right anyway, and no brute-force cracker will run more than second before having used up the 50 tries. A second, a millisecond. I'm sure the people at TrueCrypt have thought about this. What do you think is the reason for not putting in a limit to the number of wrong guesses? Thanks, from a guy who looks forward to every Thursday. Thank you, Eric.

Steve: Well, the reason is they can't. And this is a point that David made that I thought was really worth revisiting, and that is that the reason IronKey is able to enforce 10 tries is you have no access to the hardware counter in the IronKey. But in a TrueCrypt environment there's the whole PC. So you could take a snapshot of it, use up your 50 wrong passwords. It would have a counter somewhere that's inherently exposed unless somehow you could use the TPM, for example, on the chip, and the TPM isn't really set up to be used as a secure counter. But the point is that any kind of counter is going to be on the hard drive or in RAM or in a combination of those or in the registry or somewhere, where you fundamentally have access to it. So you get to 59 and just reset it to zero and try - I mean, sorry, 49, reset it to zero, and try another 49, reset it to zero, try another 49. There's just no way, unless you've got a secure counter, that you could enforce a password retry limit. It works in a client-server mode because you as a client cannot reach into the server and reset the server's counter. But when you're sitting here in front of a machine you're trying to log into, you don't have a client-server model, you're right there at the TrueCrypt server where you could reset that counter. And it's all open source. So whatever they're trying to do, everyone's going to be able to know what it is.

Leo: Right. So they can't.

Steve: You can't even obfuscate the counter. Right. It is absolutely there in the clear.

Leo: There's an advantage to having hardware and closed source.

Steve: I don't think I just heard that correctly. Did you say that, Leo?

Leo: I did say it. But that's a very - that's a good point. That's an advantage. Scott Edwards in Newcastle, Australia, has a nice reminder: Hi, Steve and Leo. I was recently trying to explain encryption decryption to my workmate. I was telling him how difficult, impractical, impossible it was to crack strong encryption. To do this for the first time I used the show notes to find the billion billion billion billion scenario - it's a number with 1,296 billion billion billion billion digits - just to make the point that strong encryption is strong. So I just thought I'd a drop a line and let you know the show notes are great. It was very easy to find the info I needed to make the point. I look forward to the show every week. Keep up the good work. Happy SpinRite user. PS: What are you up to when you come to Oz? Thank you, Scott. I was there for a two-week photo expedition. It was so much fun.

Steve: Oh, I just wanted to remind our listeners that thanks to Elaine's weekly transcriptions and the fact that we now have sitewide search available on every GRC page, if you go to www, or actually that's optional, GRC.com/securitynow, in the upper right-hand corner is a search box. I imagine that Scott put "billion billion billion" into the search box, sort of remembering that he'd heard something like that. And he pressed the button, and bang, he was immediately taken to the episode, which he could then listen to or re-read in order to get exactly what that was. And then of course after it finds the right episode you can easily use the Find in your browser to find the billion billion billion where it exists in the transcript. So that's there for everybody, for every one of the 139 previous episodes, and shortly for this #140.

Leo: It's a really nice feature. Thank you for - actually Steve pays for that out of his own pocket. That's nice of you to do that. And if you want to see what I was up to in Australia, and I'm sorry I didn't get to anywhere besides the state of Tasmania because I was there as the guest of Mikkel Aaland, and he's writing his new Lightroom Adventure book. 20 photographers and I, the amateur, went down there to take pictures of Tasmania for two weeks. And they will be part of a book called "The Adobe Lightroom Adventure II," which should be out in a few months. I don't know if I'll have a picture in the book. I mean, there were so many really good photographers. But you can see my pictures. You can in fact see the whole story on my website, Leoville.com. If you go to the blog, I put up about - every other day I would put up a post with lots of pictures in it. And then you can see what I consider to be my best pictures in the photos section of the blog. You go to SmugMug, you can see the Lightroom Adventure Tasmania folder. Flickr also has some pictures, although there are more on SmugMug. I only put what I considered to be the best pictures on Flickr. SmugMug has everything. Not everything. I took 3,600 pictures. It only has about 80 of them, 80 of the most best pictures.

Steve: Wow.

Leo: It was fun. Oh, it was a great time. And what a great place. And I apologize to Scott and everyone in Australia that I didn't get to meet. But we just - that was it, it was Tasmania, two weeks in Tasmania, which was not nearly enough. But I'll go. I'll be back. Australia's a great place. Boy, it's wonder- have you ever been down there, Steve?

Steve: Never have yet, no.

Leo: Great country. Really, people are so friendly. And it's a beautiful, beautiful country, at least the part I saw was.

James Manger, also in Australia, from Melbourne, makes a very good point about wandering thumb drives. Hi, Steve. Leaving your thumb drive on your keychain with a mechanic - we were talking about this, the two-part keychain.

Steve: Right.

Leo: Because you have a thumb drive on your keychain, and you always have to take it off. Might not allow them to read your private files directly, given that they're protected with TrueCrypt and a long crypto-strength password, but they could do even worse. They could put malware on the thumb drive. Huh? Did you ever think of that? Huh huh huh?

Steve: Yeah.

Leo: Malware could automatically infect the computer you subsequently plug the thumb drive into. Perhaps your computers are mostly safe from this if you've deliberately switched off autorun, although I'm not sure if that's sufficient. For U3 it might even require more. I think you mentioned a small TrueCrypt EXE that could go on the thumb drive so you could access your data from a computer where TrueCrypt was not installed. What happens if the mechanic has replaced this EXE, for instance, on the thumb drive with malware of the same name? Whenever you deliberately run this EXE, you'll infect your computer. The malware might even behave like the real TrueCrypt EXE, prompting you for your password. At that point the malware can read your protected data and probably find a way to send it to the attacker. The problem is, you protected the confidentiality of your data, but not the integrity of the disk.

Steve: And he's right. I mean, the more we look at TrueCrypt versus IronKey, the more I wish IronKey was small and cute and could innocuously sit on my keychain the way this gorgeous little Kingston microdrive does. This little 4GB Kingston that I use is just, I mean, it's just wonderful to have it there. And unfortunately the IronKey is crushproof and filled with epoxy and industrial strength and metal and all that, but I just can't have it on my keychain.

But I'll tell you, I mean, once again, there's an advantage of IronKey is that it is hardware encryption. Nothing is exposed, there's no drive there until you log in, and then it does the deciphering. Whereas, exactly as James says here, even my little thumb drive, it's got that TrueCrypt EXE has to be accessible because without that, it's that that mounts the device driver that prompts me for my password and so forth. So there really is that vulnerability. I mean, he's exactly right. I'm going to have to come up with some solution to that problem.

Leo: Hmm. Yeah.

Steve: I think I could probably password-protect the TrueCrypt EXE, and that might work, although you couldn't show that it's absolutely - well, maybe. I don't know.

Leo: Do an MD5 hash.

Steve: I'll think about that.

Leo: Do a hash on it, you could probably...

Steve: Yeah, but then you're not going to. I mean, it needs to be something that you have to do every time. If I were to password-encrypt it so it's a self-extracting EXE, then I would instantly detect if it were replaced. There would be no way to - well, but then again, by the time you run something to find out that it's not what you expect it is, the damage is done.

Leo: Too late, it's been run.

Steve: Yeah, there's no good solution to that one.

Leo: There's always a risk, I think, when you give somebody the hardware. When they have access to the hardware, that's risky. That's why the IronKey goes to such great lengths with the epoxy and so forth.

Steve: And I have to say again, David, I think, and the IronKey folks, they solved that problem. I mean, I was sort of thinking, oh, boy, this is overkill. But the more I think about it, it's like this is what you have to do if you want to have a secure thumb drive that you can hand to someone and say, here, I don't care what you do with this, you can't get me.

Leo: Yeah. Meanwhile, our Shocking but True Jolt of the Week, anonymous, nearby in Irvine.

Steve: Now, he actually had his name in his message to me, in his email, and I thought, no, no, I'm not going to use his name for this one.

Leo: Really.

Steve: Listen to what he says.

Leo: Steve, man, you have no idea how right-on you are about any digital resource left in, on, or around your car when you leave it for service. I have provided IT support for a dealership group local to you in Irvine, VW and Honda, you probably know them. You're probably aware that almost all auto technicians have PCs and laptops in their service bays. Well, when a vehicle is brought to their service bay, the first thing they do is inspect the vehicle for CDs and iPods to rip and import music. After that, any USB drive, portable hard drive, laptop is tapped into to see if there are any files that might be useful. Probably looking for porn. Then the items are placed back in the vehicle, and the owner never knows their music, personal files, and applications have been copied. Oh my goodness.

Steve: Yeah.

Leo: The service technicians I've observed have amassed a huge library of music and applications that never seem to stop growing. They supplied their own network attached storage to share their booty without - they put it in their own NAS server. This is like organized crime - without tapping into the company IT resources. Be aware, anything you leave in your car can and will be accessed by tinkering fingers. Holy cow, Steve.

Steve: Isn't that horrifying?

Leo: Holy cow. Well, I was always told not to leave your keychain anywhere, have a valet key and put stuff in the trunk, and then take your keys because of course they could copy the keys, they've got your registration, they know where you live. I mean, I've always worried about that. But who knew they were doing this?

Steve: Yeah, as you say, it's almost like it's organized.

Leo: It's organized crime.

Steve: They've got their network attached storage in order to have storage space. And anything you leave behind they - well, you can imagine just them sitting there ripping your CD collection because lots of people have a bunch of CDs in their car. So they empty out your CD changer, stick one disk in after the other to suck all your music out, and then put it all back. And if people have ever wondered if, wait a minute, I thought that was disk number four, how did it become number six, well, now we know how the CDs move around magically. They weren't put back in the same place.

Leo: Wow, that's really quite amazing. Holy comoly. Holy comoly. That is the shocking admission of the week. And I see why you took his name out now. And I bet you're going to be even more careful when you - of course people always leave CDs in their car. I mean, I wouldn't leave a laptop in my car. That seems kind of kooky. But now I won't leave my IronKey in there, either.

Steve: IronKey is safe, Leo. You just want to make sure you get it back.

Leo: Somebody, we had a couple of impromptu, informal meetings...

Steve: Oh, I guess actually the IronKey, if they guessed wrong 10 times, it would shut down.

Leo: They could break your IronKey. You're right, you don't want to leave it in there.

Steve: Because they'd absolutely plug it in and do some guessing.

Leo: Well, let's try some passwords. Geez, Louise. There were a number of people at the - we had these meet-ups in Australia. And a guy, it was so cute, I think it was Ivars came up, said look at this, and he has his IronKey on his keychain.

Steve: Oh, neat. And actually I saw the IronKey guys also. And their reaction was, they said, boy, this podcast has legs.

Leo: Tell them to buy some ads.

Steve: I guess wherever they were going they were saying, hey, yeah, I heard Steve and Leo...

Leo: IronKey, right on. He also had a PayPal football on his keychain. So this guy is a serious - I think it was Ivars. It was a serious listener. If it's not Ivars, I apologize to whoever it was. Hey, we're done. We're not out of time because we could go on and on. But we are done.

Steve: With Episode 140.

Leo: Yeah, you're proud now, you've passed TWiT.

Steve: Baby.

Leo: Happy now. Are you happy now? So next week we're going to cover the RSA Conference which happened in San Francisco. It is the security conference of the year. It's the big one.

Steve: And code wants to be wrong.

Leo: Which should be the title of this podcast. Leo wants to be wrong, too. Steve, thanks so much. Remember, go to GRC.com, folks. That's the place to get your 16KB versions of this for the data challenged, download challenged.

Steve: And I will remind people that all the questions they heard Leo read just now, they came to me via GRC.com/feedback.

Leo: Good point.

Steve: So that's where you go, GRC.com/feedback. There's a web form there where I get no spam. And so submit your question, and it'll get to me.

Leo: Yes, yes. And you can also search for any previous episode at that spot. Find many great programs, too, to help you, including the Perfect Passwords generator GRC.com/passwords, Perfect Paper Passwords, Wizmo, Shoot The Messenger, DCOMbobulator, and the famous, the world-famous ShieldsUP!, it's all there. And did I forget? I did. SpinRite, my favorite, the one and only, the best hard drive maintenance and recovery utility. GRC.com. Steve, we'll talk again next week.