Listener Feedback Q&A #55

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-173.mp3
Quarter size (16 kbps) mp3 audio file URL: http://media.GRC.com/sn/sn-173-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 173 for December 4, 2008: Listener Feedback #55. This show is brought to you by listeners like you and your contributions. We couldn't do it without you. Thanks so much.

It's time for Security Now!, the show that looks at security, now. Right now. Right this minute. Steve Gibson is here. Hi, Steve.

Steve Gibson: Hey, Leo.

Leo: From GRC.com, the man who discovered spyware, coined the term, created the first antispyware program, has written so many useful security utilities like ShieldsUP!, Shoot The Messenger, Unplug n' Pray. And every week we talk about the latest security news and answer questions and also kind of explain, I think you're really good at teaching, what all this is.

Steve: One of our questions this week reminded me that I also wrote the very first personal firewall leak testing utility.

Leo: That's right, which was called, cleverly, LeakTest.

Steve: It was called LeakTest, not surprisingly.

Leo: That's right. And they are now, yeah, everybody's doing it now.

Steve: There's a bunch of them now, yeah.

Leo: All right. Now, Mr. Steve Gibson, is there any security news? I think there is. I've seen a lot of stuff.

Steve: Oh, Leo, there's a bunch of stuff. I wanted to mention, we talked in October, so that would have been a number of our episodes ago, about Microsoft's uncharacteristic out-of-cycle patch, which they did because they discovered that a zero-day vulnerability - that, remember, is one where someone is seeing traffic that seems to be odd, and it's in the wild addressing a vulnerability which is not at that time known. So this was something in the so-called "server service," the RPC, the remote procedure call in the server service of Windows. And it uses port 445. And so there has been since this time a spike in port 445 traffic because there are now some worms and bots that are spreading and successfully spreading to exposed and not-yet-patched Windows systems. There's a word - a word - a worm called Conficker.A and a bot program, IRCBot.BH, which have both been seen using this vulnerability to spread themselves.

Leo: Oh, boy.

Steve: Interestingly enough, the worm goes in and fixes the bug. It patches it in RAM to fix it after it enters the machine so that nobody else can get in using the same exploit and kick it out of the machine. So it sort of locks the door behind itself after it gets in safely.

Leo: Wow.

Steve: So I did - and I don't know whether the worm is using completely random IPs because I haven't looked at it closely. The best strategy that has been used in the past, and this was the case with the major worms that we had several years ago, was that the agent would be combining looking at local network addresses, that is, IPs near it, and also IPs far away from it. The reason that's significant is that port 445 is the traditional Windows file and printer sharing port. It's blocked by many ISPs just for this reason, to protect their own customers from this, and because you really don't expect to see people deliberately doing file and printer sharing of their machine's resources out over the Internet. Now, thanks to ISPs blocking this port, you can't even if you wanted to. And of course we would certainly think that any router would be blocking it, and any firewall, and any corporate-scale firewalls.

But the reason worms will still use local IPs is that, if one managed to get into a

machine, then it would like to propagate itself within the local network, not only out across the Internet, trying to find other vulnerable machines on the Internet because, if you did have a single router protecting a LAN, or a corporate firewall, for example, protecting it, but within the network you were doing lots of filesharing, and so you had holes poked in all of the individual firewalls that exist on the machines within the LAN, then something bad could be propagating locally. So it's still a concern.

I just wanted to bring it up. I think it's an interesting note for our users, but also it's something you definitely want to make sure you're patched. I mean, these are unpatched machines which, for whatever reason, still exist on the Internet. And it may well have been, for example, IT personnel who are not applying out-of-cycle patches, even though the fact that it is out of cycle implies that it's something you really need to take a look at, otherwise Microsoft wouldn't have done this because they know how much it upsets the IP, I mean the IT personnel.

Leo: Yeah, yeah. That's [indiscernible] change.

Steve: We also talked, might have been last week, about this Vista kernel crash which Microsoft has decided - actually they decided they're not going to patch it. They're going to let it go to a service pack, wait for the next service pack because they're not feeling, at least as of the time that they are talking about this, that this represents such a big problem. You'll remember that you need to be an admin or have network operator rights on your non-admin account. It's an inherently, as far as anyone knows, a local sort of attack that allows a kernel potential remote code execution, although that has not been demonstrated. Anyway, what has surfaced in the meantime is a very simple route command that anyone who's interested can type at a command prompt that will crash your Vista machine.

Leo: But they have to be at the term- they have to be at your machine.

Steve: Right. Or have a way...

Leo: I mean, I can crash your Vista machine without typing anything. I just punch the box. I mean, if they have access to your machine.

Steve: So it's route space add space 1.2.3.4 slash 240 space 4.3.2.1 [route add 1.2.3.4/240 4.3.2.1]. And if you give that route add command, that turns out that that's an illegal subnet mask for the network that was just defined, and it causes this overflow to occur in the kernel and crashes Vista. So Microsoft says, well, we don't care. We'll fix it in the next service pack. So...

Leo: It's an ugly bug, but not really a real security threat. It's a bug.

Steve: Okay. It's a bug. And it's a bug until it's a security threat, Leo.

Leo: So it presents, you think, a potential for a security threat.

Steve: Well, yeah. I mean, all of these things, for example this worm, this worm RPC server service, it started as something that would crash the service. And then they figured out how to turn it into executing code of their choice that was part of the packet that was arriving at port 445. So all of these things, I mean, you need to really understand them to know if they can't be turned into a remote code exploit, it certainly would be mischievous, for example, if email was crashing your Vista machine. An email could certainly give that command. So that's bad.

Leo: Right, right.

Steve: Anyway.

Leo: No, that's a good point.

Steve: Yeah, I just thought that users would get a kick out of knowing, you know, "route add 1.2.3.4/240 4.3.2.1," hit Enter, and that's - it's over.

Leo: Yeah. I suppose you could attach that as a Windows scripting host command.

Steve: Yeah, precisely.

Leo: So you could send somebody an attachment that would crash their machine.

Steve: Yeah. Someone you don't like very much.

Leo: Yeah. Although you could also send them - I would think you could send a program that would crash the machine. Maybe not.

Steve: Yeah. Very good point. If they executed a program, well...

Leo: It's harder than it used to be. Used to be you could write to ring zero and stuff. Now you can't. So it might be harder to blue screen a machine than it used to be.

Steve: I wanted to update our listeners that TrueCrypt, the program we love, and I think we have a Q&A about it later on, has been updated to version 6.1a. All users of TrueCrypt are encouraged to upgrade. They fixed random little scattered bugs around, nothing terminally critical, but it's better to have these things fixed. And they've also added a feature that I really appreciate. They've given you the ability now to override the default logon screen when you use the whole drive encryption and are being prompted

for a password at boot time. Normally, it used to be that you just had this sort of this very uninteresting screen that you had no control over. And they decided, oh, let's let people either blank that screen and/or replace it with their own. So there's now a mechanism for doing that. So it allows you to sort of customize your whole boot encryption experience when someone turns a machine on and is presented with, essentially, the need to log themselves into the system. So I thought that was neat.

In a really interesting and sort of freaky story, some students who were screwing around - I'm trying to think, oh, it was University of California. Using state-of-the-art standard digital photography and computer modeling, they have demonstrated that they can duplicate a key, a traditional house key, for example, from 200 feet away by taking a picture of it.

Leo: 200 feet away.

Steve: 200 feet away. They can take a digital photo of a set of keys and...

Leo: So you kind of hide your key in your hand when you use your keys.

Steve: Isn't that interesting? I thought that was just a - it just ran across my radar, and I thought, oh, that's just too neat.

Leo: There was a demo at DEFCON where they could take a picture and then, you know, cut out a key. But 200 feet is quite a long distance. Do they have super telephoto lens and special cameras? Or was this just a regular old...

Steve: I think it's, I mean, what I like about this is, at its core, I mean, sort of irrespective of the details, it sort of highlights something that we've taken for granted. And that is, you know, you and I cannot look at a key and go home with a file and file a blank key down into that size. But computer technology, digital technology, digital photography and, I mean - certainly this key that you take a picture of, it's probably going to be rotated and off-axis. It's not going to be exactly face-on the way you want it. You know, the technology now exists to model it from the photo, rotate it, and end up driving an NC machine to grind yourself an exact duplicate. So, yeah.

Leo: Wow. Wow. That's really amazing. So you only need one side of a - I'm looking at my keys. Maybe keys should be more three-dimensional or have double-sided things.

Steve: Well, of course a lot of fancy car keys - we went through the pre-electronic car key phase where there were keys that had, remember, like different depth holes cut in them, and the...

Leo: Those would be hard to duplicate.

Steve: ...[indiscernible] car has got, like, a wiggly slot down the side. Although it turns out that all of that is just sort of to prep the electronics because it's actually an electronic handshake in many car keys today. So...

Leo: Yeah, I have a chip in my key. You couldn't duplicate the metal and get into the car.

Steve: Right, exactly. Although, you know, traditional house keys and keys that are just, we'll call them "dumb keys," apparently you can, from quite a distance away.

Leo: Now we have smart keys and dumb keys.

Steve: Apple, for the first time ever, posted an advisory telling their users that they recommend the use of antivirus software.

Leo: And then did a funny thing.

Steve: It was on one of the support pages of Apple. They said...

Leo: Yeah. And then they pulled it down.

Steve: Yup, they said use McAfee, Symantec, or Intego VirusBarrier were the three that Apple specifically recommended. And so it got picked up by the news. It's like, oh, look at that, you know? Apple has long enjoyed sort of this we don't - "we're not such a big target" posture. But it is the case that increasingly, as the people who generate these sorts of exploits also have Apples - and as we've said, you really can't make an exploit for a machine you don't own. And of course as Apple has acquired a larger market share, prices have come down. The people who do these sorts of exploits have them now. And so we're beginning to see this kind of trouble. And certainly McAfee, Symantec, and Intego have jumped on and said, okay, we've got AV for the Macs, as well.

Leo: You didn't mention, but that bulletin was pulled down shortly after they put it up.

Steve: No, I didn't know. No.

Leo: Yeah.

Steve: Oh, okay.

Leo: And I think, well, I think that what happened, the marketing department

freaked out, said wait a minute. That's one of the things we say is we don't have viruses. You better pull that down. And it wasn't in response to any particular threat. But it's good advice, of course.

Steve: Yeah, exactly. And speaking of which, I wanted to make sure that people who are iPod Touch and iPhone users have got themselves updated. Probably it's been pushed out. But anything prior to version 2.2 has web-based remote code execution problems, which Apple has fixed with version 2.2 and later.

Leo: Okay.

Steve: And lastly, or actually two last things relative to Phorm. We've talked about Phorm many times, the pretty horrible ISP background monitoring, behavior profiling, and actually stuffing cookies all over your system and in theory inserting their own ads onto web pages. In a very controversial move, British Telecom, BT, who has been at the center of this controversy because they were secretly hosting Phorm technology for proof of concept in trials without notifying their customers, well, there had been traditionally lots of activity on their boards, on their customer service forums about this. Well, they, a couple days ago, formally changed their policy, said no more discussion of Phorm will be hosted here. And they deleted all the previous threads of content that had been there. And that sort of raised a bunch of eyebrows. It's like, wait a minute, what's going on? I mean, people are calling this censorship and getting themselves bent out of shape.

Leo: Very good.

Steve: And on that note, yes, on that note I wanted to mention that there is now a Firephorm add-on for Firefox.

Leo: Firephorm.

Steve: Firephorm, F-i-r-e-p-h-o-r-m.

Leo: Ah. And does it block Phorm?

Steve: You'll find it - yes. Reading from the description here, it says "Firephorm is an extension for users of ISPs that deploy the Phorm Webwise system. It can add Phorm opt-out cookies to web page requests to avoid storing a Phorm tracking cookie for each website you visit. It can avoid Phorm Webwise.net, redirects and protects your preference to opt-out from being overridden. If your ISP deploys Phorm's Webwise system, then we strongly recommend changing" - oh. And they say, "Aside from that, if your ISP deploys Phorm's Webwise system, then I, the author, strongly recommend changing ISPs as soon as possible." So we don't like this, but we do have an add-on that you can use under Firefox that will make it lots more tolerable.

Leo: Yeah. Great. Awesome.

Steve: So two bits of errata, Leo.

Leo: You mean we made a mistake?

Steve: I have a note - no, I have a - just, well, it's not really errata, but it's non-security news.

Leo: Okay.

Steve: I have a note here to ask you about NoScript. How are you doing, a week later?

Leo: I took it off.

Steve: Okay.

Leo: I took it off. I did get a note from the NoScript folks who had actually, and I'll probably put it back on, an interesting suggestion. I'm in a different category, though, than the average user. I have to - I look at sites and report on those sites. So I have to see the site in its full experience. Otherwise I can't give it a fair review. So, you know, if I were an everyday user I would absolutely use NoScript. And clearly the security benefit is great. And maybe you already are going to say this. But one of the developers of NoScript sent me a note, Mark Zipp is his name, and he said - because I had mentioned that I was worried about missing parts of a website. He said in the NoScript Options General section there's a box labeled "Temporarily Allow Top-Level Sites by Default." And he says, "Then try one of the radio buttons below. I use full addresses. This is allows any script being run by a full address." So I'm going to try that because that will give me the full experience automatically, but I guess gives me some level of scripting protection because any subdomains are turned off. You could even get more promiscuous and turn off base second-level domains. Do you see what I'm saying? So...

Steve: Yup, I do.

Leo: Was this the note you were going to talk about?

Steve: No, although, okay, well, because I just was curious whether or not, I mean, I didn't know what your answer was going to be.

Leo: How about you? What's your experience been?

Steve: I'm happily using it and plan to keep using it.

Leo: I think if I weren't a journalist looking at sites and reviewing them, I would be much more likely to use it.

Steve: Yeah. The only caveat is you need to sort of in the back of your mind remember that it's there - that it's there, you want it, and it's protecting you - because you will go to sites where it doesn't seem like everything is on the page. And it's like, oh. And it's trivial to either temporarily allow it, if you don't think you're going to come back. That way you're not clogging up your system in general. But if it's a site you use a lot, it's like, oh, okay, and you just give it permission, and then everything's back to normal. So I just think it's a great utility. I mean, remember that I was using something like this for years under IE, using IE's zone system, where I was - and we talked about before I switched to Firefox, that's the way I was operating. So I'm a little more accustomed to this notion. And as we know, I'm seriously anti-scripting.

Leo: Right. I'll tell you what happened. I went to a site, I think it was during a show, to look at a site, and parts were missing.

Steve: Oops.

Leo: And I forgot that I was running NoScript, and I started talking. I said, well, wait a minute, I'm not seeing - and then I realized, oh, wait a minute. And so it's just it's too risky for me to be running it and not get the full experience. That's just the risk, part of the risk I pay for bringing you content.

Steve: And my final note is I have an update on one of my favorite sci-fi authors, Michael McCollum, who does the Sci Fi - AZ site. Remember that a couple months ago I mentioned that I was excited that the third book in the trilogy, the Gibraltar Trilogy, which will be called "Gibraltar Stars" - he has "Gibraltar Earth," "Gibraltar Sun," and "Gibraltar Stars." I finally sent him a note because he'd sent me something back in May, actually, and I'd had it pegged to get back to him. And I wanted to offer my services as proofreader because I would love to get my hands on the third of that trilogy. And so he says he'd love to have me proofread the book.

Leo: Cool.

Steve: But what he also said was he said he's been experimenting with the Natural Voice system. Oh, and by the way, he's completely Kindle-ized. I also wrote to him about all my recent experiences with Kindle.

Leo: Well, you helped him a lot with that, I know.

Steve: Well, actually that was back...

Leo: Oh, you helped him with Sony.

Steve: Sony, yes, that was the LRF format, the Sony format. Since then he bought his wife a Kindle this summer, and he said she's bankrupting him with all of the books that she's buying. But she absolutely loves it. He loves the fact that it's got a built-in web browser in it. And so he now has Kindle versions of all of his eBooks, which excited me a lot.

Leo: Now, do you buy them from Amazon, or do you buy them from his Sci Fi - Arizona site?

Steve: Just directly from his site.

Leo: I didn't know you could do that with Kindle.

Steve: Yes. Oh, yeah, you're able, absolutely, to use any - it's the Mobipocket format. I can't remember what the file extension is [.prc]. But...

Leo: If you download it, do you go to his site in the Kindle browser? How do you get it on the Kindle? Do you email it to yourself?

Steve: You probably - you could do it in any of the ways that you can. For example, you could email it to yourself. You could just download it and then just dock it with your computer and move it over.

Leo: I have never docked my Kindle. Not once.

Steve: I did once to sort of explore it, to see what the format was of memory and so forth, and to look around. But and I think once there were some clippings that I had that I wanted to get off of it. And so I used that in order to pull them off. But you're right, I mean, it's been more than a year now, or about a year probably, since I first got it. I got it in early November of '06. And so, and I don't know if you know, but they are once again sold out. There are 10 to 11 weeks of backlog.

Leo: You know what that means.

Steve: Probably because Oprah loves it, so...

Leo: No, Kindle 2, I think.

Steve: Ah.

Leo: Yeah. All the rumors are Q1, Kindle 2. And the 11-week backlog just or about gets you right to where you'd want to be if you had another Kindle coming out.

Steve: Yeah, that's true.

Leo: So actually more power to Amazon because what they could have done is sold all those people Kindles, Kindle 1s.

Steve: And then you'd be really upset that you got - that you missed - yes. Sort of like Apple does all the time.

Leo: Everybody does that. That's normal. That's the normal way you do business. You want to sell those old devices right up to the day you introduce the new one. And so if that's, in fact, what Amazon's up to, credit to them for doing that.

Steve: Well, and so what Michael has done is he thought, you know, there's one more format that I haven't tried, and that's audio. And so he is, for the Gibraltar series, he is using Natural Voice's Paul. Which is - actually it's funny because he and I chose the same voice. Paul I think is the best one that the Natural Voice people produce. He also slowed it down by one notch, as I do. And he's running the text of his books through it in order to generate, in the case of the first book, I think, it's a 12-hour audio file. And people have told him they would be very interested in purchasing his books in audio format, even if a robot were reading them.

Leo: Hmm. Well, we'll see.

Steve: And frankly, I mean, Paul is a very good-sounding voice. I use it for notifications. If either of my T1s go down, suddenly I hear, in the same way that I have Fred doing yabba-dabba-doo when a credit card clears for SpinRite, I hear "Your primary Internet link has gone down."

Leo: I think there's a big difference between a robot reading it and an actor reading it. But we'll see. You know, there's such a - I think that this comes from people who don't listen to audio books, that they go, oh, it's just somebody reading it to you. There's a perform- it's a performance. It's not just somebody, "And now here is book one of...." It is a performance. And a robot's not going to do that. But, you know, if you like it audio, and it's, I mean, god knows it's expensive and time-consuming to have an actor record it - I wish I had the time. I'd love to do his books.

Steve: Oh, you'd be good.

Leo: Oh, I'd love to do that.

Steve: Yeah.

Leo: Maybe I'll do it in the evening in my spare time.

Steve: And that wraps up all my random stuff.

Leo: Excellent. All right. We've got some great questions from our great audience. Mr. Steve Gibson, are you ready for our questions, brave and true? Let me see. Did I close the - I think I might have closed them. Let me reopen the questions.

Steve: The question is, are you ready, Leo?

Leo: Apparently you are, and I am not. How about that. I have them here. I have them right here. Really I do. What Steve does is he goes through the list at - in fact, you can go to GRC.com/feedback and submit questions. And then you pick questions that are representative; right? You're not trying to get just one - often it's the question that you hear the most; right?

Steve: Yes. Very often I'll see it over and over, and it's like, oh, okay, fine, I'll, you know...

Leo: Yeah, I'd better answer this, yeah. We'll start off with Art, shall we? He posted this in the newsgroups. Little different. Another great way you can participate. He asked about Sandboxie. We talked about it last week, about their help with people bringing USB drives into their organization. You know, the Pentagon just banned USB drives.

Steve: After a major breach.

Leo: Yeah. And he explained that they had had lots of trouble implementing this and wondered whether there might be any way to sandbox USB drives. Are there, is there any way, perhaps, to do that?

Steve: There really is, it turns out. There are many features of Sandboxie that we didn't go into in detail because, I mean, I could just have bullet points for a whole show. But one of them is a forced drive or directory. And it's used, for example, and Ronen suggests, for example, that you could put the drive letter of your system's CD/DVD ROM in, and it would automatically sandbox the autorun so that anything that runs from your CD/DVD drive is automatically sandboxed and has no opportunity to make permanent changes to your system. Well, similarly, in the case of a USB drive, since the drive letter can tend to float around a lot, what I would do if I were serious about this is I would put every letter in there. And you can make a list of these so you're not restricted. Put every letter in there that isn't assigned permanently to a drive or a share. And what that would mean is when someone sticks a USB drive on, and the system assigns it a drive letter, anything that you run from there is automatically sandboxed.

Leo: Wow.

Steve: So it's another of many features that's in Sandboxie.

Leo: Would that then prevent any form of infection? So if it's malware that's being run, it couldn't expose itself?

Steve: Well, our topic for next week is what are the limitations of virtual environments, like full-on virtual machines and Sandboxie. I didn't want to - I realized in my excitement, which is genuine, for Sandboxie last week, I may have sort of oversold what it can do. But so what it does, it does really well. But it's not to say that you have nothing to worry about for the rest of your life, even if the sandboxing were perfect. There are still things that can go bump. And so that's what we're going to be talking about next week. But for this sort of thing, for example, just making sure that anything that might be run from a plugged-in USB drive or from a CD that is run, it's easy for Sandboxie, which is a service and a device driver, I mean, it's really down in the kernel, sitting there watching so that nothing has a chance to get around it, it does have that feature just sort of built in.

Leo: Well, and you remember I did ask him a little bit about how it relates to virtualization.

Steve: Right.

Leo: And he was clear that it's not the same thing. It's a different kind of thing. But it clearly can be used. I'll look forward to next week. That'll be interesting.

Louis Gerard in Montreal confirms another listener's report about wider PayPal authentication availability. He's in Canada. He writes: Hi, Steve. Following up on your last episode, I successfully ordered my PayPal Security Key in Canada, too. 10 business days. $5 Canadian. That's actually less than what U.S. users pay, he says, 20 percent cheaper. Did the Canadian dollar go down that much? Wow. So, great. I use mine. I love it.

Steve: I just wanted to confirm, again, this is representative of a bunch of email that we received from people saying, hey, thanks for bringing this back to our attention. We're Canadian, and - eh?

Leo: Eh? Don't say that. They hate that. Especially if they're from Montreal. They don't say "eh" in Montreal.

Steve: Oh, okay.

Leo: They say [grunting]. Meanwhile, Jon Peter Hansen, not having any luck getting the football. Hi, Steve and Leo. I'm unable to order a PayPal football. When I click to order it, I get "The Security Key is currently not available, please try again later" message. This could mean they're just out of stock because they got Security Now'd. But I think I read somewhere that it was only available in the U.K. and Germany. I'm not sure where Jon Peter is; but I'm thinking Scandinavia, perhaps?

Steve: Based on his name, Jon Peter Hansen, sort of sounds Scandinavian. So it does sound like they are rolling them out, that they are expanding the geographic coverage. I think it was originally the U.S. and the U.K. I know that there was a second region outside the U.S. Maybe it was Australia? I don't really remember now. But I think there were two regions. And certainly we know that Canada was not supported because we have a lot of Canadian listeners, thanks to your exposure through TechTV. And they were like, wait, I can't get it up here. So now they can. And in general, I guess it was when we talked to - did we have someone on from VeriSign?

Leo: We did. Yeah, yeah.

Steve: And he was explaining that they just sort of have to do this incrementally. Their goal is to have it available anywhere PayPal is.

Leo: We love this.

Steve: But, you know, they're rolling it out one country at a time.

Leo: The idea being when you log into PayPal, you press the button on the key, and it generates a one-time-use number that you add to your password. And unless somebody knows your password and has this key, they can't get into your account. So that's really great.

Steve: And I have to tell you, it is so comforting, Leo. I've been active lately, doing some PayPal stuff. I mean, 'tis the season, as they say. And to have to provide that, push the button, give them your code, it's just it's the right thing. It's non-burdensome. You really wish that this was the way everything, all authentication net-wide, was up to speed and using because it would solve so many of these problems with keystroke recording. And we're going to be hearing about that, too.

Leo: Yeah, yeah, I agree. I'm a big fan. Although I have a feeling that the hardware devices like the VeriSign card or the football may end up being superseded by things like my bank uses SecurPass where they send a code to your cell phone. Everybody's got the cell phone. It's a lot cheaper for them to do that. And that has the same security because they have to verify that it's your cell phone number. You have the cell phone. Right? That would be equally as secure.

Steve: Yes. Yes.

Leo: I love that. I use that. I've turned that on on all my - my accountant's not crazy about it because she can't get into my accounts anymore. But I just turned it on on all my bank accounts. And the only way I get into the bank account is I log in, and then it says, okay, you need to send your SecurPass. We're going to press the button, it sends the SecurPass to my cell phone. I then have to enter in that number, and only then can I get in. I love that.

Steve: Yup, and it is, it's on the list of the authentication technologies that my future little - that my CryptoLink product will also use. I'm going to support all of this stuff in CryptoLink. And since it's possible for software to send SMS messages, that'll be one of the ways that you can set up CryptoLink to require you to authenticate yourself.

Leo: Excellent.

Steve: So zero cost there, too. And of course Perfect Paper Passwords. So you can just have the little list of goodies in your wallet and use that, as well.

Leo: That'd be another way.

Steve: Yeah.

Leo: Absolutely. Jared Burford wonders about maintaining a TPM-protected PC. He says: You've mentioned in the past about TPM, fingerprinting authorization and so forth. While I do agree this is great security - that's the security built into the processor and the laptop hardware security - I see a problem. What if your computer needs to go in for repairs? How will the technician have access to the computer if they can't get past TPM? If there's a way to disable it, then maybe, though I doubt this is possible. Even so, you still need to access BIOS setup in order to accomplish this. Your thoughts?

Steve: Well, there are a couple things. Jared should not worry that this is an unsolvable problem. It has been solved, and in a number of different ways. TPM, we did a whole episode on it before...

Leo: Trusted Platform Module?

Steve: Exactly. And it is essentially a little vault which is soldered non-removably to the motherboard of an increasing number of machines. Laptops typically have it now. Some desktops do. And the idea is that, in order for a system to be secure, there needs to be someplace that software can't access, no matter what you do. Like even if you use a boot CD in order to avoid all contact with the normal OS protections, there needs to be something fundamentally intrinsic that is, like, the last resort of protection. And that's what the Trusted Platform Module is. And in fact the way it's been designed is you can't - software cannot, cannot, no software can access its contents. You can merely ask it to verify things. So you put data in. And, for example, there isn't a way to get it back out.

You can only say this is the data I think you have in a secure way. And it can say, ah, you're right, I do.

So anyway, it's been really well designed. The UI, the user interface for this always, in every case I've ever seen, gives you a backup. So, for example, if your fingerprint won't scan, that is, for whatever reason, maybe you only registered one, you're able typically to register your whole - both hands, all fingers of both hands. So no matter how badly you damage yourself, you probably still have one. You might even be able to register the back of your knuckle, as some of our users have cleverly done when Disneyland was asking them for their fingerprint, and they said I don't think Dumbo needs my fingerprint. But the idea is, failing that, you always have the backup of manually entering your password. Which hopefully is a big gnarly long thing.

And so the idea is that the fingerprint is a shortcut for having to manually enter a big gnarly horrible password. So what you could do is either give that to the technician, which I'd be reluctant to. Obviously you would change it once you got the machine back and had it back under your control. But if you were going to change it, and you had, like, learned or written down the big gnarly long password, you might as well go into the BIOS and change it before to something simple, since you're inherently reducing the security while your machine is out of your control to a hopefully trusted technician. And so after you do use your fingerprint, or you manually type in your password, essentially that unlocks the machine, and you then have the ability to enter the BIOS the way you normally would by hitting F2 or Delete or F1 or whatever the BIOS entry keyboard sequence is.

So then you'd go in, and you'd have to give it your - in traditional change-your-password mode you'd give it your big gnarly long password, and then you'd give it "hijack" or something, something simple for the machine to have. And you just tell the technician don't worry about the fingerprint. When it's asking for the fingerprint, just type in this simple password, and you can get to my machine to fix it. And then of course when you recover your machine, you reverse that process and put back in your big gnarly long password, which is always your fallback in case the TPM-based biometric system, whatever it is, fingerprint or retina or who knows what, happens to fail.

Leo: And you would give that gnarly long password to the repair guy.

Steve: No, you would temporarily change your machine to a short password.

Leo: Give it to him, give him that password.

Steve: Exactly.

Leo: Yeah, okay. Got it. Got it.

Steve: He uses that. He never knows your gnarly long password. You don't have to change it afterwards because you've removed it, put in a simple one. Or maybe you just take it out completely, just shut down the password protection.

Leo: Right. You can do that, too.

Steve: Yeah, exactly.

Leo: Okay, okay. Yeah, I mean, it doesn't make sense. I mean, Lenovo uses these. I mean, these are laptops that are corporate laptops. Doesn't make sense that once you install the password, nobody would ever be able to access it again. That would be a bad thing.

Steve: There is a very good, intuitive, and sort of sane UI which, if you've implemented - basically, your fingerprint is a substitute for your having to manually enter the big gnarly long password. Which encourages you to use a big gnarly long password.

Leo: Absolutely. Absolutely. I just read a "10 Mistakes That Linux or Unix System Administrators Make." And one is not having a really gnarly long bad password, you know, tough, tough, tough password for the root. And never log in as root. You don't need it. They said it should be a password so tough that you have to insert the USB key, decrypt the USB key to get to the file where you've put the password and cut and paste it. That's how hard it should be. It shouldn't be something you can keep in your mind.

Steve: Right.

Leo: And that makes sense. Sam in Sweden wants to know how to really kill a hard drive. Hi, Steve and Leo. I have a quick question about how to properly scrap old hard drives. It's not so difficult when the drive is operational. Then he uses DBAN - we recommend that, Darik's Boot and Nuke - to wipe the disk clean. Google "DBAN," you'll find that, it's free. What if the drive has completely died? Hey, you know, I had somebody call the radio show with this question. So if you can't SpinRite it, you can't obviously boot and nuke it because you can't mount it. He says: I've in the past done all kinds of things such as soaking he drive in various not-so-healthy solutions, physically breaking the drive, done some damage to the drive platters by drilling holes and whatnot. Is there a quick fix? Quick way?

Somebody asked me, I'll tag this on, if they could take the drive - actually, you know who it was? It was Kris Kosach, used to work at TechTV. She used to work on the music show. She's married to Alex Wellen, the CyberCrime guy? She said, can I take my old drive, no longer operational, put it on the - they have a - you know this, Steve, because you've worked in audio - on the degausser that they use to erase audio and videotapes. It's a big magnet with a conveyor belt. Can I just put it on the degausser and degauss the drive? Will that work?

Steve: No.

Leo: Oh, baby.

Steve: Degaussing a drive from the outside won't. You really have to be in intimate contact with the platters.

Leo: So if you took the platters out and degaussed them, that would probably work.

Steve: Yes. I mean, if you take the platters out, scatter them to the four winds, or the four corners, and you're going to be safe. What I would do, to answer his question, you can always pull the board off the bottom. So that's the first thing. It's easy to do. I mean, you don't even have to be gentle with it. You're destroying the drive, so just take a screwdriver to it and pry it off, crack the board off, break it up in pieces. Now, that's not by itself enough because the boards are interchangeable among drives of the same make and model. The boards, however, are connected to the inside by a connector. And that is easy to destroy also. So destroying the connector, it just looks sort of like a dual inline set of pins. And you can just sort of scrape it off with the screwdriver. And that's going to further make it very difficult.

But if you really want to go one step further, and you have access to power tools, and I'm not kidding, simply drilling a hole through the drive, like an inch or two away from where you can tell the disks are spinning, that's game over. And it turns out that the metals are all pretty soft. They're aluminum, and so it's not like it's going to take some monster kind of drill. But if you'd simply drill a hole through, and while you're doing it you might drill a few more, it's going to go through the case. It's going to go through all the platters. And there is no way that the drive can mount and fly its head if it's got a quarter-inch hole running through the platters. And at that point it's pretty much game over.

Leo: Although somebody like the NSA might be able to use some sort of system to read the magnetic markings on the rest of the platter. But...

Steve: It's true that, I mean, if...

Leo: That's pretty hardcore.

Steve: ...somebody absolutely, desperately had to have it, they could go in, take it apart, fill the holes with something maybe, so that the head would fly across it. But, I mean...

Leo: Seems unlikely.

Steve: ...it would have to be so smooth. I mean, and again, the only way you're going to do it is you have to fly a head. Older drives, there were various things you could do where you could actually view the bits visually by subjecting the surface to various types of polarized lighting. But with contemporary vertically recorded drives, that's gone now. I mean, you literally have to fly a head with really good electronics over the surface. And if you've got some holes in the platters, there's just no way a head's going to fly over that.

Leo: I think I remember reading that in Britain probably the MI5 spec is they take apart the drive, degauss it, grind it up into a fine powder, and then they don't throw it out. Go ahead.

Steve: Returning it to its constituent atoms is pretty much a good thing to do.

Leo: And then, they don't even throw it out then. They put it in a box, and they store it in a safe in the basement, just in case. That's a little overkill.

Steve: Yeah.

Leo: Not necessary. That's good. Drill a hole in it. And that's an easy - it's actually just a few screws to open up the drive; right? It's not a big deal.

Steve: No, no, don't open it up. Drill right through it.

Leo: Oh, go through it.

Steve: I mean, that's why I like it, is in terms of, like, reasonable and seriously good, is just take a drill and go [manly drilling sounds] a few times. And it's over.

Leo: Now, you'd have to have a drill that could go through metal, obviously.

Steve: Yeah.

Leo: But you just go - so this is the drive platter is that round thing? And you just pick a place, a spot on the round thing, and you...

Steve: Yeah. Normally on any drive, certainly from the bottom, you can sort of see where the bearings are and that you're going to have the disk platters radiating out from there. And so just drill a few holes...

Leo: I love it.

Steve: ...an inch or two away. It feels good. And believe me, it seriously took care of that drive.

Leo: We opened up a drive on The Screensavers for a segment on this very purpose, and Patrick was going to show how to destroy the platters. So he opened it up. And

what he didn't know was that some drives - I didn't know this - sometimes have glass platters. So he hit the platter with a hammer and went [manly shattering glass sound], and shattered glass flew everywhere. We were very lucky we didn't get hurt.

Steve: Right.

Leo: So have you seen that, the glass platters? I thought that was weird.

Steve: Oh, yeah, sure. Yeah. And although typically they're aluminum. And if you wanted to, like, disassemble the drive and pull all the platters out, inside you'll find a hub with a whole series of either hex or, in some cases, torque screws. You take all those apart. And there might be eight of them. And then basically the platters will all come apart in your hands. And you could step on them, bend them, whatever. But really just...

Leo: You don't need to do that.

Steve: ...drilling a couple of holes through it, it's over.

Leo: Meanwhile, another Sam, hiding in an office cubicle somewhere, wonders what to do about his boss spying on me, spying on me. Sam is not, I hope, a pseudonym for Tony in the other room. Hello, Steve and Leo. I was wondering which anti-corporate-spyware program you recommend - anti-corporate spyware, hmm - as there is increasing spying going on, both in the office and at home. I feel my computer is not secure. I know for a fact some of the companies I work for on occasion - I am an IT consultant - use these kinds of programs - he references something called "Spector," which does sound kind of scary - to spy on their employees. I just have this uneasy feeling in my gut, and I want to be sure I'm not being spied on. Is there an easy way to check whether programs like Spector are installed on a computer? For instance, some program that checks or warns you if something is hooking into your keyboard and monitoring it?

Steve: The answer is, unfortunately, no.

Leo: Really.

Steve: We've talked a lot about rootkit technology. There are a couple bad viruses now that are modifying the boot sector and installing themselves before the rest of the system gets going. We've talked about Blue Pill, which was Joanna Rutkowska's sort of theoretical and continuing to evolve work on showing that it is impossible for programs to know if they are not operating on the actual native chip, but if they've been virtualized, because in theory there's no kind of test you could perform that you couldn't, as long as you encapsulated the environment appropriately, you know - and she calls it Blue Pill, of course, because it's very much like the famous sci-fi movie "The Matrix," where Neo and company did not know until they broke out of it that they were living in a simulated world.

So unfortunately there is nothing I could say except the following which would save Sam. And the following is that, if you were to boot one of these entirely bootable from CD or DVD environments, then that would never touch any resources on the machine. It would, as long as the BIOS is set up to boot first from a CD before it attempts to boot from the hard drive, and that's typically the default case, although you might have a corporate environment which is really bolted down where the CD and USB and floppy have been removed from the boot sequence, in which case you might be able to put that back in if you have the opportunity to get into the boot when it's, I mean, to get into the BIOS when it's booting. But my point is that the only way I know that you could safely use a machine is if, literally, you booted it yourself from a non-writable media that had preference priority, boot priority, over the hard drive.

And so many of these boot, you know, the pre-boot environment solutions, and they're becoming increasingly mature, I mean, you can boot up a very workable system with networking and browsing and email and everything that runs off of a CD or DVD. So, Sam, if there was something that you were doing that you needed to really know you were private about, and if the corporate environment wouldn't notice your machine had disappeared from the 'Net for a while, that's what you could do. But if you just leave it in its de facto booting, and you're not sure what's been installed, it's very possible for good technology to hide so well that it's undetectable by anything you could do as a user.

Leo: Geez, Louise. Well, there's a good reason to create a Linux boot CD.

Steve: Right.

Leo: ...and boot from that. Because even then they can watch your IP traffic. And they might, I mean, there's also you could put a camera over your shoulder, or we talked last week about Van Eck phreaking the keyboard.

Steve: Right.

Leo: It's pretty hard not to do that.

Steve: Yeah. So it's better just not to worry about being spied on, and don't do anything at the office that you would not want your boss to see. Maybe get your work done and, you know...

Leo: Well, courts have held for years - now, he's a contractor, so it's a little different. But if you're an employee, you have no rights. You're using the boss's equipment, the boss's premise, the boss's Internet access. You have zero rights.

Steve: Yup.

Leo: The boss can do anything he wants. Doesn't have to tell you. That's just the

way it is. I mean, I usually tell employers it's good to have a policy, written policy, make sure you explain what you watch, what you don't watch, what you allow, don't allow. But they don't have to.

Steve: Well, and when I've done similar sort of consulting, I've said to people, look, just bite the bullet, put a sign, like a notice glued to the top of everybody's LCD screen...

Leo: We're watching.

Steve: ...that makes it very clear. This is corporate property. Your use of this is at the discretion of the corporation. We reserve the right to log, track, monitor, filter, do anything to all of your use of any sort of this computer. Just put it there. And the advantage is everyone's notified, nobody's being discriminated against, and what it will do is it will tend to tamper down, tamp down, any of that kind of behavior that is inappropriate in the workplace anyway.

Leo: Now, of course, if you're bringing your computer in as a consultant, there's two concerns: one, that there may be something personal on that computer; but also that they could put something on that computer, and then you go to another business, and they could spy on that other business or even infect that other business. So it behooves you to have a sanitized computer every time you go to a new client; right? You want to sanitize it.

Steve: Depending upon what you do with the machine while you're there, yeah.

Leo: Would Steady State or Deep Freeze, Faronics Deep Freeze, one of those programs that kind of, you know, you reboot and everything goes back to the way it was, would that effectively eliminate that kind of problem?

Steve: Yes.

Leo: Okay. So that might be another way to go.

Steve: Although, I mean, depending upon, I mean, it should. But I don't want to say, like...

Leo: Well, they may not wipe the boot sector; you know? They may...

Steve: I was just going to say, if you boot something that someone gives you, you've lost control potentially before Windows and Steady State gets itself going. So...

Leo: Right, yeah.

Steve: ...it can be bad.

Leo: An interesting challenge for somebody who goes from office to office with his own system. Hmm. We'll have to think about that one. John D - I'm sorry. Let's go to Patrick in Des Moines, then we'll get to John D. Patrick in Des Moines needs some clarification about hard drive passwords: Quick question about hard drive passwords, the issue that was discussed on 171, a couple episodes back. I'm understanding - am I understanding this correctly? The only way to thwart a hard drive password - and we were talking about the old IDE passwords, not TPM and that kind of stuff, or the built-in hardware - is via the manufacturer's intervention via a subpoena from a government agency? If that's the case, wouldn't the contents of any government computer's hard drive - the Social Security Administration, for example - be relatively secure when that computer is lost, stolen, misplaced, or reutilized? Here's the part I'm unclear about. If the hard drive that is set to require a password is transferred into another computer that does not support this function, would the drive function, or would it remain locked? I'm hoping it would not function. And if that's the case, it seems like this is a simple solution to protect data. Thanks for a great netcast. I learn a tremendous amount from every single episode. That's a good point. Why doesn't the government lock its hard drives?

Steve: Okay. The issue comes up. And so I thought, okay, what I should do is to explain what the mechanism is. And then our listeners, who are certainly smart and alert, can extrapolate for themselves. The mechanism is that in the API, the application programming interface to the drive itself, you are able to say to the drive, hi there, I'm the BIOS, and here's a password that I'm giving you. And henceforth I want you to refuse access if you've been reset or power cycled unless I give you this same password again. So that's what it is. It's not - there's no encryption going on. It's simply at the interface to the drive something says, hi there, here's a password. Henceforth I'm asking you to require this password before you will accept any instructions except some basic ones like, you know, the drive ID instruction where it gives you its serial number, make and model, and so forth. There are a couple simple sort of background characteristic reads that don't contain any user data at all, which the drive is still allowed to respond to. But other than that, it will simply fail any request it receives with an error that says, sorry, you need to give me the password before I'm going to do anything for you.

So that's what it is. So to give a drive the password, you typically need BIOS support. That is, so the BIOS needs to be able to give the drive the password. But so, for example, understanding that model, Patrick asks, if I were to move that drive to a different BIOS, would the drive remain locked? Well, clearly it's the drive that's locked. So the lockedness goes with the drive. It follows the drive. And a different BIOS might use a different algorithm to translate the user-provided passphrase into the actual code, the actual digital password that's given to the drive. So, because there's no real standard for that, that is, the mapping of a passphrase into whatever pattern of binary is given to the drive as its password. So it might be that, if the algorithm were the same, for example, same make and model of laptop, then you could expect to relocate a locked drive to a different laptop of the same make and model and be able to have the BIOS unlock it when it powers up. But it's equally likely, and in fact very likely, if you went to an entirely different type of laptop, that that drive would remain locked, and nothing you could put into it that you could figure out to put into it would result in the BIOS unlocking

the drive. So the drive is inert.

Now, and relative to subpoenas and things, what this really means is that the drive at the interface, at the connector point essentially, has been instructed, do not do anything unless we give you the matching password. That password is written in a maintenance area on the drive. And so that does mean that the drive gods, typically the drive manufacturer, but also third-party data recovery companies also, have the ability to go in and remove that password from the drive with, you know, with direct access to the drive. And it's very likely that there are undocumented commands, not publicly known commands that make it easy for, like, the manufacturer to provide some super secret code that would tell the drive, okay, we know you're locked, and somebody gave you a password, but ignore that please.

So I want to explain the difference in security between this and whole drive encryption. Whole drive encryption means that there is nothing stored on the drive except noise. And it doesn't matter, you don't even need the drive lock anymore. It doesn't matter what anyone does. No force on earth can determine what that data is if it's been whole drive encrypted, given that it's been done correctly and that the passphrase is not known. So what's there is just noise. But if it's just locked with a standard BIOS-level hard drive lock, then that lock is carried by the drive, but it can also - the drive can be instructed to ignore the lock, not only by the drive manufacturer, but as I said, by third-party data recovery companies have the ability to do that, as well.

Leo: They could probably just pull the platters, put it in a new mechanism, and be able to read it. The data is not encrypted.

Steve: Exactly. The data is not encrypted. So I don't know whether pulling, I mean, when we say "pulling the platters" glibly, it's very difficult to do that. I mean, it's...

Leo: Well, but that's what people like DriveSavers actually do. They have duplicate mechanisms, and they have a clean room, bunny suit environment, and they can take the platters and put them in a new drive.

Steve: What they would need would be they would need different microcode for the drive. Because if they pulled the platters and put them in a different drive, that drive would obey the lock which is carried by the platters.

Leo: So it is on the platters, stored on the platters.

Steve: Yes.

Leo: Okay. All right.

Steve: Yes. And so...

Leo: So that still wouldn't solve that.

Steve: ...for that reason, for example, an easier thing would be just to switch the electronic boards. If the lock were memorized on the electronic boards, you could just swap it, which is very easy to do, and then have it unlock. No, it's actually on the platters. That's what they're using for their permanent storage. So but you could certainly have microcode which ignores the lock. And this could come from the manufacturer, or it could be reverse engineered. So there are ways around this. But it's sort of at the level of, for example, it's like protecting WiFi with a MAC address lock. I mean, it would prevent people from casually reading the data, but it's not as good as full crypto, which prevents anyone from ever being able to read the data.

Leo: I've got a way for you to be even more famous. You should create the Gibson Scale of Crackability. And it'd have to be, you know, it's kind of hard because there's things like MAC address versus decrypting a password, I mean, it's apples and oranges. But maybe you could make it in the number of hours it would take a reasonably sophisticated hacker to crack that, something like that, and create a Gibson scale. And you could say this is an eight on the Gibson scale, or a two on the Gibson scale. Would you do that?

Steve: No.

Leo: You'd be famous.

Steve: I don't want the responsibility.

Leo: You'd be like Richter. Nobody remembers who Richter was, but they know his name.

Steve: They do. And they unfortunately remember him at times of severe trouble.

Leo: True, maybe Richter's not the best example.

Steve: Yeah.

Leo: Let's see. John D in Chicago, IL, poses a great question about cracking decryption: Hi, Steve and Leo. With the recent discussions about various encryption/decryption scenarios, I have a general question about knowing when a possible decryption method works. I think it's fairly obvious to figure out when something like a password crack works by being able to use the password to gain entry into something. That's pretty obvious.

Steve: Ooh, look, I'm in.

Leo: It worked. But considering both transmission decryption and file decryption, how does an attacker know when something he or she is trying actually has worked? For file decryption I assume it would deal with working with the file system and file metadata to determine what something is. In other words, it'll say, oh, this is the right form for a PDF file or a text file. For transmission decryption I assume it would be something like being able to determine the encapsulated frame packet, et cetera, header information, whether it's a viable transmission medium. These thoughts popped into my head while doing an encrypted file copy using SCP over a Hamachi connection over an encrypted wireless network. Thanks for the great shows. This is actually a real problem, and it probably is an academic problem. How do you know when you've got the cleartext?

Steve: Yeah. I mean, and that's what I loved about the question. I don't think we've had it asked before. And remember that the process of decryption, all the processes that we know of that don't involve a badly broken cipher, that is, where the cipher is intact, you have plaintext. And one way or another you are brute-forcing, you are guessing one after another after another key, applying the key and the decryption algorithm on the cipher against the cipher text in order to get the plaintext. So the question is, how do you know when you've got it? Because essentially it's not like the cipher algorithm says bingo, I've correctly deciphered it. No, it doesn't work that way.

All the algorithms that we've been talking about, you put - it's sort of like garbage in, garbage out. You put something in, and you're going to get something out. The algorithm doesn't know anything about whether what you put in is readable or not. It just says, oh, look, this is binary data, and I'm an algorithm that transforms it into a different binary data. And so it's entirely up to the decrypter, to, that is, the person who is involved in this, or the system, whatever the architecture of this cracking project is. It's up to something or someone to look at the data that comes out of the cipher and say, did we guess correctly? And so, you know, if it were just unknown data coming in, and you got unknown data coming out, it might be the right unknown data, but you have no way of recognizing it.

So John suggested some things, for example, if this was a packet that was going through. And he talks about the headers, for example. And there he's completely correct because many forms of data have pretty well-defined structure. He talked about file system metadata where, you know, you've got in certain locations of the file system, like in the old FAT file system we had the file allocation table and the boot sector, for example, at the very beginning, and the directory at a given location, and various structures which are known. And in the case of packets we have a known formal definition for the layout of a packet where you've got the source IP, the destination IP, on the outside are the source and destination MAC addresses and so forth. So, and relatively well-formed structure.

So in something where you have a sense for what you're expecting, like if you're trying to decrypt a packet that you snatched out of the air, you could apply some heuristics, some rules of thumb to look at the output of each of your guesses of a key after it comes out of the cipher to say could this possibly be a valid packet? And there's enough structure that one chance out of many, many, many billions would you get something that looks valid but really isn't. And so that's enough to, like, for the system to kick out, here's the key I used to get this. This looks - this meets my first pass criteria for having properly decrypted it. I'm going to keep working while the humans come along and apply a higher level of judgment to whether this thing works or not.

And, similarly, say that it was just ASCII text. It was completely known ASCII text. But you had, in a block of text, say maybe you were decrypting a sector, 4K bytes. Well, even in 4K there's many things about ASCII text, even if you don't know what it says, that are tip-offs. For example, in ASCII, typically ASCII compresses highly because, for example, the high bit, the eighth bit is always off. Most of the alphabet fits, typically fits within the first 0 to 128, or 127, rather, characters. So your high bit is off. Well, in a block of 4,000 characters you're going to have - well, actually 4,096 in a 4K block - you're going to have 4,096 high bits. So if any decryption of something that you thought was just ASCII happened to suddenly have the majority of all of its high bits off, that's suddenly a very good chance that you've decrypted it correctly because encrypted text, which we know is going to be highly random, encrypted anything is going to, on average, have 2,048. That is to say, 50 percent of the high bits in the bytes are going to be on; the other 50 percent are going to be off. And it's going to be extremely regular. And so if suddenly you get a decryption of even something you don't know at all, except here again we know something about, we've made some assumptions about what it is that we're decrypting. We're saying we think this is ASCII. And when we get right, suddenly most or all of the high bits are off. The chance of it being a wrong decryption where all of that is true is just astronomical.

Leo: Although I suppose somebody could throw in a decoy where - this would be pretty tricky - where it seems to appear to be a decoding, but in fact it's a middle step to another decode. Is that possible?

Steve: Well, in fact it was one of the things I said a long time ago that actually generated a surprising amount of controversy when I talked about double-encrypting something.

Leo: Right. Oh, I remember that. Oh, we got so much email on that one.

Steve: I know. People just - they were pulling their hair out on that.

Leo: In other words, is double encrypting it making it harder to get to cleartext.

Steve: Right, because I was taking the position that, if you double encrypted using different keys, and technically you could probably even use the same key, although you'd like to use different keys for more. But if you didn't, if someone did not know that it was double encrypted, they'd only be testing it after a single decryption. They would never know to do it twice.

Leo: They'd never see the cleartext.

Steve: Much better. Exactly. You'd never get cleartext. You'd just get more random stuff. And they don't know that they ever guessed one of the keys correctly. So, yes, the more you obscure it and prevent somebody from being able to test - because this is the point of John's whole question is in every case it's a test you apply to what comes out of the decrypter, whether you got it right. The decrypter itself doesn't know. It just says you gave me something; I gave you something else.

Leo: It's what I think. Right.

Steve: It doesn't know.

Leo: Right. A human is needed.

Steve: Right.

Leo: I mean, it's easy when you unlock a door, it's unlocked. But it's not so easy with decrypting. Thomas in Stockholm, Sweden, wants to stay with the tried and true. But he wonders what can be done, if anything, to safely run older software with known security problems - like, say, Windows 95/98/ME would be a good example - but which is no longer supported by the manufacturer. His example is he has ACDSee, which is picture editing software, really a great program. But he'd have to buy a new copy because the old copy is no longer supported. Is there a - could you Sandboxie it?

Steve: That is exactly the answer to the question.

Leo: I'm not so dumb as I look.

Steve: It's why we keep you around, Leo.

Leo: I'm paying attention.

Steve: That's exactly what I would do. I would say to Thomas in Stockholm that, yes, Sandboxie would be a perfect solution. It's unlikely that he's going to get attacked through picture editing. I guess he would be opening a picture like a JPG where Sandboxie's JPG decoder had a known buffer overflow problem.

Leo: Which was in fact the case, yeah.

Steve: Those have been known to exist. But so that he's opening a picture that is a malicious JPG that tries to do something. Well, this is exactly what Sandboxie is designed for and why it's so cool that it's a general purpose sandbox that's able to sandbox not just Internet applications, but anything that you're doing. And so, yeah. And the cool thing is I've now been using Sandboxie for several weeks. I see no overhead, no core code or RAM bloat consumption. I see - nothing seems to be slowed down at all. I'm using it for Firefox and Eudora, my email client. And they're all just running perfectly. So I'm really pleased with it. And I'm sure you could run ACDSee in there and just have the confidence that, if by chance you did open a malicious JPG, it wouldn't be able to do any writing to your system. Sandboxie would protect you from it.

Leo: Very cool. John Pitt in Melbourne, Australia has discovered his ZoneAlarm is sneaky and leaky. He says: Hi, Steve and Leo. I recently heard on one of your Security Now! episodes a recommendation to get a program called Wireshark. What is that? That's the old - they renamed it.

Steve: That used to be Ethereal.

Leo: Ethereal, that's right. I did this. Thanks. And in addition to discovering the packets I was looking for, I discovered that ZoneAlarm was constantly sending DNS requests and packets to its own server. I don't like that. Even though I have another firewall running concurrently with no permission for this ZoneAlarm behavior, the other firewall does stop ZoneAlarm from contacting Zone Labs for the phone home purposes, but it doesn't see these DNS requests at all. How are they doing this? What are they doing, and why? Please talk about why ZoneAlarm is so sneaky. I love to listen to your show every week, and I hope you continue to educate people about the ugly world of Internet security. Hey, I'm a little disappointed in ZoneAlarm. What are they up to?

Steve: Well, when I saw this I thought, oh, yes. This is something that Gregor Freund and I once discussed as a sneaky way of getting data out of a PC that firewalls would not block. Now, I'm still getting myself in trouble for having once, many, many, many, many years ago, recommended ZoneAlarm.

Leo: No, it was a good program. You introduced me to it.

Steve: Version 2.6 something was a great program, back when Gregor and Conrad and those guys, the original founders were there. I liked them; I liked it. Now the company bears no resemblance, nor does the product bear any resemblance...

Leo: Oh, so they're not there anymore.

Steve: No. They've sold out long ago.

Leo: Oh, okay. And literally.

Steve: And the thing's just become a bloated nightmare program.

Leo: We no longer recommend it. You know what, you recommended it, but we recommended it on The Screensavers. I recommended it on the radio for years. It used to be a great program.

Steve: Yup. And that is seriously in the past tense. What ZoneAlarm is doing is it is deliberately using DNS to bypass firewalls so that it is able to contact ZoneAlarm, and

ZoneAlarm's end-users are unable to block it. The reason this happens is that DNS is one of those many things that uses the service host. So what happens in, for example, in XP, in one of the services running is called the DNS client service. And so you must give permission, any firewall must give permission to the DNS client service, or your system can't do DNS lookups. And we know from having talked a lot about DNS that email and web surfing, I mean, everything needs to use the DNS system.

So what happens is programs do not themselves use the IP system, the UDP/IP or TCP/IP system, to form and send DNS requests. Instead, this is a service offered by the operating system. So the program asks the operating system to please look up the IP of this domain name. Then the operating system does that on behalf of the application. Well, that means that the operating system has to be permitted to do DNS. And so what John has described is, literally, it's ZoneAlarm deliberately being sneaky, ZoneAlarm doing something for the sole purpose of bypassing the user's control over its phoning home. And he saw this with Wireshark. He says that they're sending DNS requests to ZoneAlarm's own servers. Well, ZoneAlarm, the only reason ZoneAlarm would have servers is to receive stealthful DNS packets from its customer base.

Leo: What's the content they could send? They could send lots of information.

Steve: They could be sending anything they want.

Leo: And it gets through because it's a DNS request, and so it just bypasses...

Steve: Yes. It's because the DNS request goes to the operating system. And all firewalls, either you...

Leo: If you don't allow it, you can't surf.

Steve: Well, exactly. Either the firewall, the first thing that happened when he installed it was it said I need you to give me permission for DNS, and you have no choice. You say, okay, fine. Unfortunately, ZoneAlarm is deliberately using that permission in order to send its own packets containing lord knows what, I mean, anything it wants, over which you have no control.

Leo: Oh, that sucks. That's terrible.

Steve: It's really bad.

Leo: I'm devastated.

Steve: I'm unfortunately not surprised.

Leo: And a very good reason why you might want to download the free Wireshark. I mean, you have to be somewhat sophisticated to use this. It's a packet sniffer. But boy, that's - and it's a protocol analyzer because it can...

Steve: Or don't download it. Just get rid of ZoneAlarm. Just say unh-unh.

Leo: Well, clearly we now know. Wow. Assuming that this guy is accurate in what he's saying. But it's interesting because you had that conversation with Gregor.

Steve: Yes. We were talking about this a decade ago, I mean, or whenever it was. When ZoneAlarm was very young we were talking about different ways of sneaking data out. And it was on my list of LeakTest enhancements for years. And it's a known way of getting data out. I'm disappointed to hear, but not surprised, that ZoneAlarm is doing this.

Leo: Well, just to be fair to ZoneAlarm, this is what a listener tells us. We haven't independently verified that they're doing that.

Steve: No. And given the facts, this is my explanation for what it is he's seeing. And it's interesting, too, that this other firewall is blocking the overt phoning home, but it's not catching this backdoor leakage through DNS. And there's no reason ZoneAlarm should be sending DNS packets to ZoneAlarm. I mean, to Zone Labs. That's nuts.

Leo: Right. Right. Unless they wanted to send out information of some kind. But again, I just want to make sure that you understand we haven't verified this independently. This comes from a listener.

Steve: Yup. And I'm not installing ZoneAlarm on anything I've got.

Leo: Yeah. Wouldn't surprise us. But there are other reasons not to install ZoneAlarm anyway.

Steve: Right.

Leo: Wow. Chris in Detroit needs a bailout. Yeah, join the club. Get in line.

Steve: I couldn't resist using that word.

Leo: Get in line. For his friend's once-infected PC. Hi, Steve. I love the Security Now! podcast. I've been a long-time listener. I'm having a few problems on a friend's computer. First, the machine was the unlucky host to a virus. My friend installed

somehow Comcast's Internet Security. I can't - they relabel somebody, McAfee or Norton.

Steve: Yeah, of course.

Leo: Which found the virus and diagnosed it as Apple HEBI. It is now removed. But the problem is that whenever you try to visit Google, the computer forwards you to what looks like a Microsoft page saying you have spyware - oh, boy - download our new spyware removal, but you know it's not. Any other Google site just ends up with a 404 error. Access to the rest of the Internet works fine, though. I was concerned - yeah, really? No. I was concerned that possibly the virus had changed his default DNS server's IP. I corrected that by sending him to OpenDNS, and I looked up the IPs of the servers on my own computer, so if his was possibly infected it wouldn't send me to a fake OpenDNS.

Steve: Very sharp.

Leo: So I set up his DNS servers manually, went to Google.com, still no avail. How could this possibly happen? The problem has to be within the computer itself, and I'm at a loss for a solution. He's got Antivirus 2009, is what he's got.

Steve: Well, what he also said, I liked it, in the subject of his note he called it "Pre-DNS?" with a question mark. And so what this could be is nothing more than this thing made a modification to his hosts file. Because the hosts file, as we've often said, is where the PC goes before it goes out onto the Internet. And so if something said Google.com and then gave an IP to this bogus Google.com, any attempt to access Google.com would instead be redirected to this foreign server. And it also follows that other things that are Google.com, gmail.google.com or google.com on other pages, well, those wouldn't be supported by the foreign spoofed server, so they'd come up as 404s. But you have more specific knowledge about something that does this, Leo?

Leo: Yeah. And I'll say a couple of things. First of all, there is a removal tool called Malwarebytes, B-y-t-e-s, lot of people recommend. But you and I both agree on this one, Steve, that what he tried to do, he got a virus, and he tried to manually remove it.

Steve: All bets are off.

Leo: All bets are off. And oh, lo and behold, something's still wrong. You don't know what's wrong. You might have a rootkit. You might have...

Steve: Can't know what's wrong.

Leo: You can't. You've got 20 other things going on. I say this on the radio show. I want you to just tell me if I'm completely full of it. Once you've been infected, really the only safe bet is backup your data, format the drive, and reinstall Windows from a known good source. That means an install disk or a system recovery disk.

Steve: Yeah, I was just going to say that the advantage is, his system is running. So with a running machine, I mean, my point is that many people have, for example, their hard drives die. And then they're like, ooh, I wish I had a current backup. It's like, yeah, I bet you wish. Go get a copy of SpinRite and pray a little bit. But here he's got a working system. So he has access to his data. As you said, get the data off, all the things you care about, your programs, your My Documents folder, all that stuff, get it off. And then the only thing you can really do to be sure is rebuild the system from scratch and then restore your data.

Leo: From a known good source that can't be corrupted.

Steve: Yes.

Leo: Like an install CD. There are a lot of cocky guys. Every time I say this I get guys in the chatroom say, oh, no, no, I can remove that. And there are a lot of guys out there, and gals, who say, oh, no, I can get rid of anything. I have the toolkit. And, yeah, you can get rid of things. In this case you didn't. But let's say you did, and everything seems to be all right. These guys don't want you to know they're on there. There's no - you have no way of knowing that you're completely cured. Once a bad guy gets on your system, he modifies all sorts, he can modify all sorts of stuff, hide things everywhere, and you can never detect it.

Steve: Well, and we are seeing a gradual escalation of the amount of - because it's evolution - of the amount of technology and effort being employed by the bad guys to make their stuff harder to remove. I mean, in the good old days it used to be you'd look in autoexec and win.ini and the Startup folder, and you'd take some things out of there, and it would be game over, you know, nothing is running anymore that you don't know about. But oh, those days are so far gone. I mean, system files can be replaced with things that, I mean, I've had friends who were cocky like this, who have spent untold weeks fighting, just because they're so stubborn. They've spent so much more time than if they had just pulled the data off and...

Leo: Formatted the sucker.

Steve: ...reinstalled. And then they call me, and they say, Steve, I don't know what the problem is, what's wrong. This thing keeps coming back. And I say, well, I know where the problem is. It's that you've spent three weeks on this so far, and you've called me four times, and every time my advice is give up, this is really beyond you now. It is very possible that this is beyond the casual, oh, I used to know what every file is on my computer, and I kind of still do. No.

Leo: The program Apple HEBI is a rootkit, by the way. So...

Steve: Ooh, goodness.

Leo: Yeah. We know it's bad. We know you're in trouble.

Steve: And what that means, just to remind our listeners, is it means you can trust nothing that you see. The rootkit means it's down in the operating system. It is filtering the operating system's own use of itself. So when you do a directory, when you bring up a listing, nothing that you see can be trusted. And, well, because the rootkit is in there making the OS lie to you about its own condition, about the files it has. You can no longer trust it. The only thing you could do would be to boot something, as we said earlier in this show, boot something off of a CD that allows you to inspect the hard drive and the file system so that you're not running the OS itself, you're running a different OS coming from a read-only medium. And then you've got a chance of seeing reality. But again, scraping this thing off - and in this case you want to make sure that your boot sector is cleaned, as well, and the first track of the hard drive.

Leo: Mess, just a mess.

Steve: A mess.

Leo: I, you know, I was that cocky guy many years ago. And I learned my lesson.

Steve: Well, and we all were because once upon a time...

Leo: Sure. Used to be able to do it.

Steve: Once upon a time it was feasible. It is no longer feasible.

Leo: That's really the bottom line. All right, our last question, Steve Gibson, and it's a good one. It's actually one I think I wanted to ask you last week. Ken Harrington, Herndon, VA, says, well, if you use Sandboxie, and it's so good, why do you need NoScript? Steve, thanks for your recent show on Sandboxie. It looks like a great product. The question is, doesn't it make NoScript redundant? If I understand it correctly, even a nasty JavaScript can't do any damage if it's trapped inside the sandbox. Can I just uninstall NoScript?

Steve: Well, this being the last question, it is a bit of a segue into next week's topic, which is the limitations of the use of virtual machines and any sort of sandboxing program like Sandboxie. But to answer Ken's question on this exact topic, you can make a differentiation between, and arguably these terms are a little similar, but security and privacy. I would say that Sandboxie is not an enforcer of privacy because things running

in the sandbox have full read access to your system unless you deliberately block them. There is the ability, and one of the other features of Sandboxie is you can, if you've got, like, you don't want anything to access your My Documents folder, you can easily put that in Block Access, and then something running in the sandbox will not be able to access anything under your My Documents folder and the whole tree of things below it. So that's yet again another cool feature of Sandboxie that we've never talked about before, the ability to deliberately blind things in the sandbox to some aspects, as many as you want, of your machine.

But where I would say that is a security benefit, Sandboxie is not providing a privacy benefit. And that's one of the things that not allowing scripts to run, a browser with no scripting and no other known security vulnerabilities that are being exploited, it is a read-only device, too. It's showing you static pages, which are not able to run any code, coming from a remote location. So there's nothing, there's no way for information to get out of your system and back out. But if you're running scripting, then there's the potential, at least, for the scripting itself, or for the scripting to invoke other modules in the system that have discovered vulnerabilities and use that to get data out of your system. And so, and notice that if something has access to your system, for example, it could look at the other pages that are loaded in your browser, maybe one page in your browser has the ability to watch you enter username and password into a different page in your browser. So even though all that activity is constrained within the sandbox, and nothing is making a permanent change to your system, it could still do some damage. So I really see both NoScript and Sandboxie as companions to each other, and neither one rendering the other obsolete.

Leo: There you go. And why not use both? After all, you got it, it's free.

Steve: Yeah, exactly. Unless you're you, Leo.

Leo: I don't use either. But that's okay. That's okay. I'm okay.

Steve: But now here I am Mr. Born-Again Firefox User, it's like, oh, what's wrong with everybody?

Leo: I do use Firefox. I do use Firefox. And I, you know, absolutely, I'd probably use NoScript at home. But I don't feel too much at risk anyway. We'll see. Hey...

Steve: You and I are both safe, and we don't go in strange, dangerous locations.

Leo: Yeah, I'm pretty careful about what I do. Hey, we have come to the end of our show, but not the end of Security Now!. You know, if you go to GRC.com you could find 175, something like that, 173 issues. Go back in time and listen to them all if you haven't. There are 16KB versions to make your downloads quick, if you don't mind the kind of reduced audio quality. Of course the full 64KB version, as well. And transcripts, if you like to read along while you listen. Some people like to absorb information through their eyeballs. You can find that, too. GRC is also the home of SpinRite, the world's best hard drive maintenance and recovery utility. We didn't do

a SpinRite letter, did we, today?

Steve: I just had that thought. You're right, we didn't.

Leo: You want to do one?

Steve: I've got one.

Leo: Yeah, let's hear it. SpinRite is such a great tool. We use it all the time. I mean, I'll give you a testimonial. We use it all the time around here.

Steve: I found a really fun one from a guy in New Zealand whose name is Hamish. He said, "Hi, Steve. As a loyal Security Now! listener from the very beginning, I've been aware of SpinRite and its stories for miraculous recovery for some time now. I've owned a copy for a while and used it occasionally on misbehaving hard drives over the years. Recently I got" - oh. And the subject, that's what - it was "SpinRite Saves Lunch."

Leo: I like it.

Steve: And I thought, okay, I've got to read this one. I've got to find out how SpinRite saves lunch. So he says, "Recently I got a call from the school cafeteria where I work, telling me their computer which runs their cash register software would not boot up. This was about half an hour before the cafeteria was due to open and be filled with hundreds of hungry students. I raced over to the cafeteria with my SpinRite CD in hand, popped it in, crossed my fingers, and made offerings to several gods in the hope that my students would not go hungry. After about 20 minutes - it's a small drive, only 4GB - SpinRite said it had finished and repaired a couple of problems. With bated breath I rebooted the computer, and up it came, loading the register software and allowing students to buy their lunches. Ten minutes later students filled the room and ordered their lunches, unaware of how this wonderful hard drive recovery utility saved them from an afternoon of grumbling stomachs. Thanks, Steve, for such a wonderful utility."

Leo: I do like that. SpinRite Saves Lunch. SpinRite to the rescue. GRC.com, you get your SpinRite there, you get your free software utilities, and of course ShieldsUP! and all that great stuff. It's a great site, highly recommend it. Steve will be back next week, and next week we're going to talk about virtual machines, what they can and cannot do to protect you.

Don't forget you can listen to Security Now! and not only download it from TWiT.tv and GRC.com, you could subscribe to it in iTunes. If you want to hear it every week, that's the best way to do it. Just do a search for Security Now! in the iTunes store, absolutely free. And/or you can watch it live. If you're really a glutton for punishment, watch us do the show live in video. You could see Steve's shining face every - we do it every Wednesday at 1:00 p.m. Eastern time. I'm sorry, 2:00 p.m. Eastern time. It's 11:00 a.m. Pacific at live.TWiT.tv. Steve, thanks so much. We'll

see you next time.

Steve: Right-o.