SECURITY NOW!

Transcript of Episode #188

Listener Feedback #62

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-188.mp3
Quarter size (16 kbps) mp3 audio file URL: http://media.GRC.com/sn/sn-188-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 188 for March 19, 2009: Listener Feedback #62. 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! with Steve Gibson, the show that talks about privacy, security, everything you need to know to stay safe online, and does it with no apology whatsoever, in the absolutely most geeky way possible. Steve Gibson's here.

Steve Gibson: We do have - I see as I'm running through the Q&A sometimes that people will say, well, I had to read the transcripts on that one three times before I really knew what you were talking about. But now I do, so.

Leo: Well, I think that there's plenty of places you can read kind of fluff and generalized stuff about this stuff. But there's not that many places where you can get the real goods, especially not that many people like you who can really dig into it, understand it, and then deliver it. So...

Steve: And, yeah, I don't really have a choice. I mean, I truly love this stuff, and I get excited about it. And it's the details that are really interesting, so.

Leo: "It's just the way I'm made," says Steve Gibson. "I don't have a choice." Well, what are we talking about today?

Steve: This is a Q&A episode. Yet there's so much other stuff that I need to talk about that I only put together 10 questions rather than our normal dozen. And we may not even get to all of those. So I put the good ones at the front because...

Leo: So in other words, this show is going to go down, straight downhill. Listen now, because the rest of it's...

Steve: Actually it was hard because there were lots of really good questions. So let's just try to move through it quickly and so we can get to everything. But there's a lot that I want to share with our listeners that happened this week, in the intervening week since we last spoke to our listeners. And we've got great Q&A stuff, too. Some, well, PayPal seems to be a never-ending theme, and...

Leo: I see we have the PayPal Horror Story of the Week.

Steve: Oh, this one, Leo, I'm just - you're not even - in fact, well, you ought to read it first when I'm talking about other stuff so you can prepare yourself for...

Leo: It's that bad my jaw is going to drop?

Steve: Oh, it's over-the-top horror.

Leo: Oh, I like that, though. I like being surprised because you can hear me catch my breath, go [gasping], "What? Oh, my god."

Steve: Oh, it's - this is just unbelievable.

Leo: Also a battery breakthrough from MIT. I'm very excited to hear about this. I don't know what you're going to say because I haven't - just before we started, Steve said, "Have you heard?" And I said no. So we're going to find out about that. Or let's start with the tech stories because there's a lot of them, tech news, errata, that kind of thing.

Steve: Well, we have, yeah, we do have, we have security news and a bunch of stuff.

Leo: Right.

Steve: So we were talking - we've mentioned several times in the past about the problems with security vulnerabilities in Adobe Reader. And some people indicated that, well, just switch over to Foxit.

Leo: Yeah.

Steve: Oops.

Leo: What?

Steve: Turns out...

Leo: No?

Steve: ...that Foxit has - the Foxit Reader has multiple security vulnerabilities, which they have just acknowledged and fixed.

Leo: Well, now, last week we talked about this, and you mentioned there was a patch. Is this over and above what you mentioned last week?

Steve: Well, what I had heard last week was just sort of some rumblings that maybe switching to Foxit wasn't going to fix you up any better, that either the same problems or different problems might be present. It turns out that these are different.

Leo: Right.

Steve: But these are remote execution code problems that involve specially crafted PDFs. And it turns out Foxit has a huge following.

Leo: Oh, yeah. I love it.

Steve: I think it's 20 million copies are in use now? So it represents enough of a target that you could imagine bad guys saying, well, let's send some PDF files, small ones, off in spam mailings, and we're going to catch some people. So, and essentially the exploits are known. Proof-of-concept code is out on the Internet and available. So it's probably not even a matter of time before this thing gets exploited. So I wanted just to notify any Foxit users to go back to the Foxit mothership and update their copies, or make sure that they are currently using the most recent version because there has been, just in the last week, an update to fix a number of remote code execution exploits, stack-based buffer overflow. I mean, the traditional problem that really all software really seems to have when it isn't written with just an absolute focus on security as one of the main things.

And you can imagine that the Foxit guys, like all other programmers, are more focused on getting it to work and getting their particular thing going in it than they are in absolutely thinking about every possible way it can be exploited. And as a developer myself, I can vouch for the fact that it's very difficult to get your head into that mode. I mean, it's coming at your software from a direction that you fundamentally don't - that you're fundamentally resistant to. So, I mean, the bad guys have an advantage because they have no stake in this thing operating correctly. Their stake is in finding where it doesn't. And so it's just - it's very hard to make code bulletproof, which is why week after week we keep running across one program or another that's got a problem.

Leo: I'm looking. I think this is the update that I mentioned last week, which was Foxit 3, Build 1506.

Steve: Yes.

Leo: So if you get the latest Foxit, don't freak out. There's not been another. These are the three flaws, the buffer overflow flaws - yeah, okay.

Steve: Yup, exactly.

Leo: Yeah, yeah.

Steve: You mentioned that last week?

Leo: I did. But it was in peripheral - it was peripherally that they had - remember I said they had problems, they fixed those problems with Build 3, they say it's not the same, what was it, big JPEG [JBIG2] issue that the Adobe people had.

Steve: Right, it's not the same, although they did have a big JPEG [JBIG2] issue, but it was different from the...

Leo: Right, different one. Because they use their own libraries. They don't use the Microsoft or the Adobe libraries.

Steve: Meanwhile in security news, the BBC did a rather controversial thing that upset a lot of security analysts and specialists. A guy named Spencer Kelly that does security-related work for the BBC went onto an Internet chatroom and purchased on behalf of the BBC a 22,000-PC botnet.

Leo: Oh, my goodness.

Steve: For the purpose of using the botnet to actually do bad things.

Leo: What?

Steve: Well, I mean, not against people. For example, they set up two email accounts, a Hotmail and a Gmail account. And he used the botnet to send spam through the botnet, from the botnet to those two email accounts.

Leo: Oh, just as a test.

Steve: Yes, they were spamming themselves. He also used the botnet to aim a denial of service attack at a cooperating security firm that confirmed that, yes, 22,000 machines it felt like were attacking them. Then, when all this was done, they recorded it, made the TV show and everything. They changed the screensaver of those 22,000 machines to alert the owners...

Leo: Hello, this is the BBC. You have got a problem. What a surprise. Oh, my goodness.

Steve: They changed the screensaver to turning it into an alert that would let them know that their machine was infected with a malicious agent. Which, I mean, I didn't see what the alert said. But presumably they were saying, "This is the BBC. We don't know where you are on the planet" - because of course they don't. These are 22,000 machines located anywhere, that they had control of as a consequence of purchasing this botnet from a chatroom.

Leo: Did they say how much they paid for the thing?

Steve: I didn't see a price.

Leo: Probably didn't want to mention it.

Steve: But they called it an "inexpensive botnet" because...

Leo: Well, it's not that big, yeah.

Steve: Aw, 22,000 machines, what can you do with that?

Leo: Oh, geez, Louise.

Steve: And then, after changing everybody's screensaver to this warning notice, they shut the botnet down. They used the botnet to remove itself, essentially, well, to shut it down, having taken control of it. And this notice helped, aimed people at some help that would tell them how to disinfect their machines from this. The problem is that it violates all kinds of laws.

Leo: It's illegal to do what they did.

Steve: Yes. It's illegal, I mean, they modified, without permission, the settings and operation of these 22,000 machines. Even though their intentions were good, hopefully the result was good, you cannot do that. Even if you're Robin Hood. It's just not okay. So it's caused a big kerfuffle. And they say, well we talked to our attorneys first. We would never have done this if we hadn't talked to our attorneys first. But people who are in the know of such things in terms of where this falls says, eh, not so good. So...

Leo: Well, I mean, it's pretty obvious it's not so good. You'd think they would, I mean, forget the attorneys, what about the ethics of it? I mean, you're modifying somebody's computer. Now, admittedly, it needs to be modified. But we've seen this happen before. We've seen kids write viruses that delete other viruses.

Steve: Like antivirus viruses, yes.

Leo: Yeah. And that's never a good idea.

Steve: Well, it's just not okay. It's the old "two wrongs don't make a right," even if your wrong is intended to be for good. I mean, what if it did something bad? What if, by mistake, they had hurt these? So, I mean, you really are opening yourself to some liability. And I wouldn't be at all surprised if, given the grayness of this, if somewhere among those 22,000 people, somebody was really upset by what the BBC did to their machine without their permission. I mean, they would feel the BBC was in my machine, gained access to my machine and made some change. It would almost have made more sense, frankly, for them not to acknowledge that they had used a bot in anyone's machine to do this, if they were determined to do it. It really does seem to me like, well, again, their heart was in the right place, but ooh, boy, are you asking for trouble. And I'll keep an eye on the story to see whether in fact anything further is done. It would be sad if the BBC were hurt, but that's the nature of litigation in the world today.

Leo: The other side of this is everybody knows these exist. Everybody knows how they operate. They have - it really is almost a little sensationalistic. It's not - you don't need to do that. I guess to prove the point that it's so easy to buy them.

Steve: Well, that's a very good point. What did they achieve by demonstrating that they got - maybe they demonstrated that anyone in a chatroom can buy one of these and do the following things. So...

Leo: I guess that's not such a bad thing to let people know.

Steve: Today I'm announcing the result, I don't know how much of the email dialogue you saw back and forth with myself and the other judges of the YubiKing Award. That was in the last couple days. You may remember that Yubico, the makers of the YubiKey, in preparation for the RSA conference coming up next month in April of 2009, they wanted to create an awards process to encourage people to do things with the YubiKey, and that all the people in their wiki, and there were a couple of hundred people in the wiki, and then a panel of three judges, of whom I am one, or which I am one, would also have a vote. Well, as it turns out, the top two wiki-voted entries were both by companies who had purchased a whole bunch of YubiKeys. And they used their YubiKey inventory to vote for themselves.

Leo: Oh. Oh.

Steve: So there were, like, it was ridiculously lopsided. I mean, the only solutions that got, like, a whole ton of votes were the two companies that had purchased a gob of YubiKeys. And the way the voting worked, you had a - the YubiKey authenticated you, so there's no way to vote twice unless you had...

Leo: A whole gob of YubiKeys.

Steve: A whole gob of YubiKeys, and just stuck them in one after the other and touched the little button and voted for yourself. So not surprisingly, those companies thought that their applications of the YubiKey were better than anybody else's, and they were there to stand behind that belief...

Leo: Oh, that's so funny.

Steve: ...by inserting all their YubiKeys and voting for themselves. So we discounted all of that, of course, because that really wasn't representative of what seemed good. There are three winners, sort of in different categories. And one of them is so cool. I mean, if nothing else came out of this other than all of our listeners finding out about this one, then it's been worth everything.

So the first is a really interesting sort of cloud-based app or system called Maventa, M-a-v-e-n-t-a. What they've done is they're working to essentially come up with a secure electronic invoicing system. They make the point that still today in this day and age, despite the fact that we have the 'Net, and everybody's plugged in and chatting and talking and sending email all over the place, the process of invoicing is still paper. That is, invoicers print out their payables and put them in envelopes and lick them and put them in the mail, and they trundle across the world wherever they're going, and then they're received, and they're opened up and logged into payables systems where then these things get batched, and checks are written, and that sent back.

And so this company says, okay, this is dumb now that we're not doing e-invoicing. And there is of course lots of problems. I mean, you need to do this securely. One of the things you need is to be able to sign these invoices, and you really need good authentication. So the way Maventa is using the YubiKey is that a company is able to submit electronic invoices to them, and they go there in a batch. Then you visit their website and view all of the invoices that you're sending out and agree that that's all correct. And then you use the YubiKey to - you are the authenticated invoicer for your organization, so you use the YubiKey for that one batch in order to sign them, authenticating them. And then Maventa turns around and forwards these electronically to the recipients of the invoices.

So anyway, it's a typical, state-of-the-art, this is how, this is like what the future is going to be where we use multifactor authentication, strong authentication, provided in this case by the YubiKey, an inexpensive, lightweight, easy-to-use, deployable token to provide that kind of required authentication that a system like this needs because you certainly don't want any major breaches in a system like this, and authentication of these transactions is required.

The second winner is a company called Collective Software, who has essentially come up with a multi-armed authentication solution sort of for enterprise-class Windows networks. And there's, like, a bazillion of them. So any company, this thing integrates with Active Directory for companies that are using Active Directory. It's called AuthLite, A-u-t-h-L-i-t-e. And if anyone - and they grabbed that domain, which redirects to the proper AuthLite page on their software site. So if you just put www.AuthLite.com, that'll bounce you over to Collective Software's page. And I've looked through it. It looks very nice. I mean, essentially they've got the whole Windows logon, Windows authentication, remote network VPN authentication, the whole package associated with Windows enterprise-class networks, implemented with the YubiKey as the multifactor token for logon. And so it's a soup-to-nuts solution, a company that wanted to increase - that wanted to use the YubiKey for Windows authentication. They can do that now with this AuthLite solution.

Okay. And the final winner of the YubiKing Award is a neat guy in Switzerland who - it turns out the company he works for was part of - is in the same building, coincidentally, and Stina didn't even know this until she looked him up after the panel, the judges, including myself, yesterday went back and forth about a bunch of these password sort of solutions and decided this was the guy. I mean, his was - there's a bunch of honorable mentions I'm going to run through real quickly about other password solutions that people may already be using which are now YubiKey enabled. But this was such a cool solution that I lobbied hard for this.

The product is - well, it's not even a product. It's free. He calls it KeyGenius. And again, you put "keygenius" into Google, and you'll find it. The way it works is slick. You create an add-on for your browser, so it uses a browser add-on, something that is running in your browser, watching you log into sites. And you essentially, you go to his server, and you create a - you tell his server what the password for any given site is that you want to log into. You don't tell - you don't have to create an account. You don't have to create an identity. You don't tell it what your username is, only what the password is for a given site. And you authenticate that then with the YubiKey.

So, and then he does a real-time - and this is in one-time password mode, the strongest mode of using the YubiKey. Then, anytime in the future, when you're online, and you are at that site where you're being prompted for the password, you just touch your YubiKey. So what goes into the password field is not the password for the site, but the one-time password that will never be repeated by the YubiKey. That's typed into the password field. The add-in that is running in your browser notices, it sees you entering a YubiKey-looking password. And so without you doing anything, it goes, it connects to his server and uses what you just entered to authenticate you because you've got that YubiKey. It looks up the matching password that you have logged into his site before and, on the fly, swaps it with your YubiKey entry in the password for the proper password, and then hits Enter.

So essentially what this means is that you can take your YubiKey with you anywhere.

This is secure login, for example, even in an insecure caf mode because, even if there were a keystroke logger, it would be logging the YubiKey keystrokes, which is just fine because it's never valid again, thanks to the YubiKey being one-time password. It would never see, the keystroke logger would not see the actual password because that's never typed at your keyboard. The add-on swaps it behind the scenes and then submits the login form. So it's just a very clever, nice solution. It's free.

The caveats are that this is just, at this point, it's just some random guy that loved the YubiKey and wrote this app because, you know, and set up a server just because he wanted to. So it's not clear that you can count on this always being up. I'm hoping that bringing some attention to him and this solution will - maybe someone will come along and buy it from him and turn it into an industrial-strength solution. I mean, it's just a - it's a clever, simple, nice way of using the YubiKey in a function which is inherently mobile, that is, you don't have to have it, you don't have to carry anything with you.

He's got a nice FAQ where he asks himself all the questions that people who are worried about this in all the different ways you might be, like, well, wait a minute, I'm giving you all my passwords. And he says, well, true, except you're not giving me any of your userIDs. I don't know anything about you. I don't know your email address. I don't know your name. I know nothing. All I know is this one aspect of authentication, not the other. So there is no way that he's able to take advantage of that.

Everything is encrypted. If you read the FAQ, it's clear that this guy understands security. He establishes an encrypted connection between your browser and his server, so all of that is secured. All of the passwords are stored encrypted at his end. And even if someone did get them, it doesn't help them because they don't know anything about what they're associated with. There's no knowledge of - there's no association to you or to the sites where they would be used. So - but they're encrypted anyway, so it wouldn't - they wouldn't be able to do anything with the information if they were able to get it. Anyway, it's neat. I played with it, as did the other judges. And we agreed that this guy deserves the YubiKing Award for having just created a nice, interesting, new approach for using the key.

I also wanted to mention that the product SoManyPasswords.com now is YubiKey enabled. LastPass is YubiKey enabled. KeePass, which is a very popular password management utility, is YubiKey enabled, as is Password Safe. So all of those were also entries because they are YubiKey capable. They just, I mean, they did sort of the normal thing you would expect with a YubiKey. So we wanted to acknowledge them, make sure people knew that they were able to use their YubiKeys with them. But didn't quite reach the level of something, a new, cool application that we thought was really clever.

I also wanted to mention that Joomla!, Drupal, and Enano are all now also YubiKey enabled, those content management systems. WordPress blog has a plug-in. There's an Apache module now for using the YubiKey for Apache server authentication, a Mac OS log-in, Google Apps you can now use the YubiKey for authenticating yourself, and just a bunch of other things. So overall the YubiKey is continuing to gain traction and is finding homes all over the place.

Leo: You deserve a lot of credit for that. I think if it weren't for you running into Stina at RSA a year ago, I don't know how YubiKey would be doing at this point. Have they made the move to the U.S. now? Are they in the...

Steve: I don't know actually where they are. I have not kept track. There's one last thing

I wanted to - two last things I want to mention. One is that there was one really fun entry that I just wanted to mention because it was just great. The guy called it Yubihome, as in "you be home." But he uses the YubiKey as his door key. He didn't win YubiKing Award...

Leo: As his door key.

Steve: As his door key. It's over on the other side of the door from where the handle is. He's just got, like, a USB extension cable, the head of a USB extension cable poking out through the...

Leo: Oh, that's hysterical.

Steve: And so he plugs his YubiKey in, and the little ring lights up, and he touches it. And he's also got a speaker overhead, and it says "Welcome Howard," or whoever his name. It says, "The door is now unlocked, and you have three new messages."

Leo: So he somehow tied his lock to a computer.

Steve: Well, yeah, he has a motorized lock. He got a motorized door lock. And those are available freely. Companies like Safe House sell those. So those are for whatever, like security or automated house sort of applications. And so there is a computer that is at the other end of the USB extension cable, which runs some software that authenticates his YubiKey and unlocks his front door. So it's clever. But it's not the sort of thing that everyone's going to go run out and do. So we wanted to give him an honorable mention as a submission. And he submitted a video that was really fun, too.

Leo: Oh, how clever. That's so...

Steve: Now the best news of all.

Leo: Yes.

Steve: They will be announcing next month something that everybody who is using a YubiKey wants, and that is a split personality YubiKey, where a single key can be both one-time password and static password.

Leo: Oh. I do want that, very much, yeah.

Steve: Yes. Because so, I mean, it just makes so much sense. And actually it's more than that. It's actually a dual personality key where either personality can be any configuration of the YubiKey. And so the idea would be you touch the circle for one second, and that engages the first personality. If you touch and hold it for three seconds, it engages the second personality. And so you could have two different static keys, two different one-time passwords keys, with different secret keys inside. So for whatever purpose, I'm not sure why you'd want to do that, but, well, for example, one reason is if you were using a third-party server for doing one-time password authentication, yet you still wanted the public side for, like, using the Yubico one-time password server. Then you would need two different one-time password solutions in a single key. So they are going to be announcing doubling the functionality essentially of the YubiKey next month.

Leo: Very cool.

Steve: Okay. So the big news from MIT in the last week appeared in the Letters section of Nature magazine, announcing MIT's, well, two materials scientists, some chemists at MIT, have come up with a major battery technology breakthrough. What they essentially did was they have come up with a technology for changing the surface crystallization in an otherwise standard lithium ion cell, where the surface is specially prepared to create a much more, effectively a porous to lithium ion surface. The upshot of this is you can take existing lithium ion chemistry, which is well understood and well developed. You switch it to using this particular electrode preparation. And you can now fully charge and discharge a lithium ion battery in a matter of seconds.

Leo: What? Charge and discharge in - this is like those ultracapacitors that we were talking about.

Steve: Well, exactly. In fact, in their paper they show the ways in which this technology is similar to ultracapacitors, even though it's entirely different. I mean, for example, as we know, the ultracapacitor owes its potential for high energy storage by using really high operating voltages. Now, that's the controversial aspect of an ultracapacitor, and actually it's a problem with its application. That is, if you were talking about, what was it, 3,500, 35,000 volts of charge, so you need to step up your available charging source up to that level, and you need to step - when you're using the capacitor's stored charge, you need to step the voltage back down to five volts if you're going to be using this technology in a laptop. So the beauty of using existing lithium ion battery technology is that we understand it, it's mature, fabrication's in place, and the charging and discharging, it's operating at natural use voltages instead of something exotic.

Leo: Well, how does it charge in nine seconds if it's the same voltage?

Steve: Well, voltage and current are different. So voltage is pressure, and current is flow.

Leo: Flow. It would need higher current; right?

Steve: Well, and so these guys - oh, yes. And in fact, the current, the available current is the limiting factor. For example, you could not - you cannot charge your plug-in hybrid vehicle in 10 seconds because you need too many kilowatt hours of energy. So a vehicle with this battery technology could technically charge itself up in a few minutes, but you'd have to give it way more than household current. So what you can imagine is, you can imagine the equivalent of a gas station, but now it's an electron station, where you literally...

Leo: A tank, you need a tank to fill.

Steve: Well, you literally drive your car up when it's near empty. You have some serious industrial-type connector which looks like some megawatt plug. You plug it in, and this thing dumps a huge amperage of current into your car. And in a matter of a minute, just like you're filling your tank now, this thing could recharge your car's next-generation lithium ion battery.

Leo: That's really amazing. Now, what about a laptop? Could you use - I guess you couldn't - could you use it on a laptop?

Steve: Absolutely. I mean, now, we're probably two or three years away from this getting out into the market.

Leo: Good, because I just bought a laptop. I don't want to buy a new one.

Steve: Yeah. We're probably two or three years away because, I mean, now, two companies, two producers have already licensed the technology from MIT. So, I mean, everyone gets it that this is a breakthrough. I mean, the days of charging up your cell phone or your PDA or your Kindle overnight, that's going to be gone in a few years. And I can imagine somebody four years from now listening to this podcast, it's like, what? You had what?

Leo: All night?

Steve: You guys used to have to do that? That's crazy.

Leo: This could be a huge breakthrough. And what I love about this, as opposed to ultracapacitors, is it works with existing battery technology.

Steve: Yes, yes.

Leo: How much of a change is it? Do they change how they manufacture them?

Steve: Well, yes. Again, this is all in the lab. And these guys, they talk in this paper, none of the - this is in the press a lot this week. But all the stuff in the press is just sort of your top-level surface junk, and it didn't really talk about how this works. So I bought a PDF from Nature of their paper, which is deep in chemistry and material science. And it talks about how they - what they make this of, that this is a lithium iron phosphate electrode, which they heat to 600 degrees for some length of time, then they raise it to

900 degrees, and they do this and that. And they understand, being materials guys, that what they're doing is they're changing the surface, the crystalline surface structure at the nano level so that it is far more permeable to ions. And it's the ionic permeability of the electrodes which have traditionally limited the rate at which you can charge and discharge lithium ion cells.

And they've got charts and diagrams. And they show, for example, they state in their paper that the typical power rate, okay, so that's not the total amount of energy, but the power rate, the rate at which you're able to take power out of a lithium ion cell, the traditional lithium ion cell, is between 0.5 and 2 kilowatts per kilogram. So think of it, between half and 2 kilowatts per kilogram. In their test cells, using their modified lithium ion phosphate electrode, they're able to get 170 kilowatts per kilogram. So from 2 to 170.

Leo: Wow.

Steve: So it's orders of magnitude. And that was a full discharge of the battery. They charged the battery up, topped it off, just like you do any lithium ion battery, although much more quickly, and they discharged it fully in nine seconds. So they dumped all of the battery power out in nine seconds. So, I mean, what this means, as you said, you asked for laptops. We're back again to plugging it in and counting maybe to 10, or maybe to 100. But, I mean...

Leo: That's great.

Steve: ...no more hours required to charge. See, right now...

Leo: But, now, we wouldn't need a special charging station, though; right? I mean, again, we need extra current to flow that much - or maybe not. Is a battery, a laptop battery that much current?

Steve: And that's my point exactly, is that we're not talking about filling up a car battery.

Leo: Right, right.

Steve: We're filling up a laptop battery, so...

Leo: You could do it on your standard, whatever it is, circuit.

Steve: Well, it will be different charging technology. So, I mean, it's not like we're going to be able to get new batteries and stick them in our old laptops because that won't happen. It'll be the next generation of laptop. It will work only with these next-generation batteries. And so when you plug your laptop adapter into the wall, okay, the house lights will dim a little bit.

Leo: That's not good.

Steve: But only for 30 seconds.

Leo: For nine, nine seconds, yeah.

Steve: It'd be like running your microwave, where you can sort of, ooh, wow, this is sucking some power out of that. But in 30 seconds your hamburger is hot. And in this case...

Leo: Now, this would also increase the capacity; right? We should be able to get much longer life out of these; right?

Steve: I don't think that's the case.

Leo: It's not, okay.

Steve: Because it's still using - and they don't directly address this in their paper. And if it did increase the capacity, they certainly would have addressed it. Because it is using standard lithium ion technology, they just solved the rate at which you can charge and discharge. Now, the other reason that's important is that, well, first of all, it means that you solved the problem of recharging, given that we actually would create electrical recharging stations the way we have gas stations now. But say that we stayed with a hybrid model. The problem with traditional hybrid technology, where you've got a gas engine, an internal combustion engine, is that there are better ways to convert fuel, gasoline, to electricity than an internal combustion engine hooked to a generator. What you really want is an external combustion engine. And that's called a turbine.

Leo: Ahhh.

Steve: It turns out that 17 years ago Ben Rosen, who is a very famous venture capitalist - he was the seed money behind Lotus and Compaq and a number of other tech startups back in that era - he decided he wanted to create a hybrid power train for cars. So he needed a way of converting - he needed two things, just like we do in hybrids now: a way of converting gasoline to electricity, and a way of storing the electricity. His solution for converting gasoline to electricity was far more efficient than ours is today because he used a microturbine generator, which had one single moving part, which was suspended with an air bearing and spun at 96,000 rpm. And so the beauty of this is that you're able - you spin this turbine up. You feed it gasoline or kerosene or who knows what it takes. But, you know, something "sene." Out comes energy with an extremely efficient conversion, much more so than we get with an internal combustion engine driving a generator.

Now, the problem is you've got far more energy than a battery can accept because - I mean traditional batteries, than lead acid or lithium ion. So you've got too much current.

So he said, okay, we can't use batteries because the idea is we're going to run this turbine on a short duty cycle to recharge something that is able to accept that much energy in a short time. Then we shut the turbine down. So instead of running, like, a traditional engine for a long period of time at a relatively low efficiency, the idea is you run it for a much shorter period of time, so your gas mileage is much higher.

What this guy Ben and his team came up with for storing the energy turned out itself to be controversial because he was using a magnetically suspended flywheel which spun, friction free, in a vacuum at 55,000 rpm. Now, it was cool because you could spin this flywheel up. That is, a flywheel would accept the energy from the microturbine during the short time it was running, spin it up, essentially giving you storage of energy in a mechanical form. And then it would work as a generator, dumping its energy into the wheel motors, just as we have with contemporary electric hybrid vehicles. The problem is, you don't ever want to be in a car accident where you've got a flywheel spinning at 55,000 rpm, a few inches away from your legs.

Leo: No.

Steve: Or the small of your back, or anywhere near you. And so, I mean, they understood that. They went to great lengths to make this thing safe, to wrap this thing in shock harnesses and, I mean, literally I'm sure there's like a heavy firewall between you and the flywheel. But that was the means of storing energy. They ended up obviously not producing cars. They made one. They made a prototype, and the darn thing worked, just like they thought. I mean, they magnetically suspended flywheels.

That ended up spinning off into a separate company which today spins magnetically levitated flywheels as a replacement for lead acid traditional UPS, Uninterruptible Power Supply systems for datacenters. And that turbine technology, there's a company called Capstone which they spun off which makes microturbines for all kinds of applications. So these things never went together in that way. But now that we've got this kind of lithium ion technology, which can be recharged in, literally, I would say, in a matter of seconds, but we can't source that much energy to it. But now that we've got it, something like a microturbine-based hybrid suddenly makes a lot of sense because the turbine spins up, does a much more efficient conversion of gasoline to electricity, brings the lithium ion battery back to full charge, and then shuts itself down.

Leo: That is so cool. You know, we just bought a hybrid car, Jennifer and I, because she wanted a big car, and the only way I could justify it, it's a Toyota Highlander, is with a hybrid. But so those are using regenerative technologies. Do they not use flywheels anymore?

Steve: Which?

Leo: The hybrids, current hybrids out there?

Steve: Oh, never did use flywheels.

Leo: Oh, I thought it was, okay.

Steve: No, it was just an idea that Ben had. And in fact, that's, again, one of the other...

Leo: It is a good way to store energy, of course.

Steve: Oh, yeah. I mean, and we've talked about electrostatic in the form of an auto-based supercapacitor. And we've talked about chemical in the form of a battery. And now mechanical in the form of a flywheel. But it looks like, I mean, this is really interesting because the battery runs at useful voltages. Apparently some of the electric cars are not very quick off the line because it is difficult to pull the kind of energy out of an existing lithium ion that you would like to in order to really accelerate well. Well, that problem is gone now, too. So we're talking about cars with tremendous acceleration which are able to dump their mechanical energy back into the battery as quickly as they need to when they're braking, so they'll use regenerative braking; and that are able to accept power, either from a highly efficient turbine using external combustion technology, or from - maybe you could charge it in your garage, if you can provide enough power for that to make sense. Otherwise, there'll be some sort of universal, high-current power plug, and we may see electric charging stations in the future.

Leo: Very, very cool.

Steve: But it's big news that I wanted to share with everybody.

Leo: Well, really timely, I mean, energy news. That's so exciting, yeah.

Steve: Yup. And just in the nick of time. And it's interesting, too, how - back in 2002 I found a paper, they were talking about lithium ion phosphate as having its conductivity dramatically increased, and literally how this could really make sense for batteries and be a breakthrough. It took seven years from that knowledge being in place, and guys in the materials labs working out, okay, how do we actually solve the problems? And so it - oh, the other thing I forgot to mention is it looks like this also solves the cycle life problem. Lithium ion suffers from both a shelf-storage life limitation, where literally the chemistry gets stale, even if you're not using it it's aging; and then also a maximum cycle life of on the order of 350 to 500 cycles. I know that Apple is bragging about a new technology they have in their latest laptops that's supposed to be a thousand cycles.

Leo: Yeah, yeah, yeah, right.

Steve: That demonstrates that, yeah...

Leo: We'll see.

Steve: That demonstrates that there's some awareness of that. It looks like in their initial work that this solves the cycle life problem, too. So, I mean, we're basically talking three or four years from now I expect this era of having to charge our little portable things for almost as long as it takes us to use them, that's gone. We're going to really change that duty cycle.

Leo: Well, remember you talked about that screwdriver that charges in 30 seconds. I finally got it, and it does indeed work. It does charge in 30 seconds.

Steve: And that's a supercapacitor-based screwdriver.

Leo: Yeah. It's very cool. So...

Steve: That's neat.

Leo: Yeah. It's not the greatest screwdriver. I don't know if it's because of the supercapacitor, but the barrel is kind of big. The ergonomics are not great. But it charges in 30 seconds. And it does, and it has a good lifetime, lots of torque. It's a nice design.

Steve: Well, it's got an infinite lifetime, technically.

Leo: Yeah. I mean, we'll see.

Steve: Yeah, there's nothing to die electrochemically in there. So it's neat, I mean, so we are seeing supercapacitors happening. It looks like the supercapacitor now is going to have a run for its money with this next generation of lithium ion technology.

Leo: That's awesome.

Steve: I do not have a SpinRite testimonial.

Leo: What the heck?

Steve: I have an amazingly cool tip.

Leo: Oh, good.

Steve: Dave Jones in Birmingham, Alabama says he got SpinRite to boot over PXE.

Leo: PXE, okay, I have...

Steve: Which is network boot.

Leo: Yeah.

Steve: And it's built into all BIOSes now and has been for some time. So he wrote, he said, "Steve, I thought you might get a kick out of this. I just bought the four licenses to get a corporate site license of SpinRite for our firm. We've now successfully gotten SpinRite to work over a PXE network boot. It involved taking the fdboot.img floppy image from FreeDOS and merging spinrite.exe into it. We then placed a call to it in FreeDOS's autoexec and put the .img file in our PXE menu. To my great pleasure, it booted straight away."

Leo: That's cool.

Steve: "We can now run SpinRite on any computer in our firm, almost a hundred machines, directly from the network, without lugging any disks or CDs around the office. I hope to put up a full tutorial on how to get this working on my blog, and I'll send you a link when I do. Thanks, as always, for such a great product. Being an Assembly programmer myself, I truly appreciate the hundreds of hours of careful programming that SpinRite represents. Regards, Dave Jones, IT Manager."

Leo: That is really neat.

Steve: Well, what's so neat about it is you've got a computer that won't boot. And so it's like, okay, cool. You reboot, you go into the boot menu and into the PXE menu and say, "Run SpinRite." And it provides - it boots FreeDOS that is bound into SpinRite and then runs SpinRite. So, I mean, it's like - and then SpinRite goes to town and fixes your disk. So it's just - it's such a cool idea to run a data recovery tool that will solve boot problems. And it also makes it easier to run in a preventative maintenance way. You just reboot your machine into SpinRite and let it run overnight, and come back the next morning, and go back to your normal work. So I just thought it was really cool. I wrote back to Dave, I said, please, please, please share the way you did this on your blog. And I will share it with our listeners when he gets his blog up.

Leo: I'm sure you'd love to have that be part of the SpinRite instructions. That's really cool.

Steve: Yeah.

Leo: All right, Steve. I'm ready to read...

Steve: [Groaning] Buckle your seatbelts.

Leo: This comes from Francis in London, the Horrifying PayPal Revelation of the Week: A brilliant show - it does seem like it's weekly these days. A brilliant show, he says, keep it up. I'll try and keep this as brief as I can because you get so many of these. I needed to reset the password of a client's PayPal account. I used the forgotten password link on PayPal, received an email, clicked it - this means that he's on the client's account because the email went to him.

Steve: Right.

Leo: And expected to be asked to verify my identity. PayPal said the link had expired, even though I had just requested the PayPal reset, so I tried again, same thing. So I called customer service. They asked me to quote the last four digits of the bank account number. It's a client's corporate PayPal account, and I didn't know it, so I said so. So the customer service guy said, "Why don't you guess?"

Steve: He asks him to guess the number.

Leo: Why don't you guess? Well, it's four digits. That means there's, what, 10,000 possibilities.

Steve: Uh-huh.

Leo: I said there was no way I could guess. He said, "Well, guess the first digit." I said I couldn't. He said, "Well, give it a try." So I guessed five. He said, "Higher." So I guessed eight. He said, "Lower." I guessed seven. He replied, "Correct, it's six. Now guess the second number." Strangely, I managed to guess all four numbers. He then sent me an email which, when I clicked, allowed me to change the password and all the basic security questions, et cetera. What?

Steve: Oh, my goodness.

Leo: What? Boy. To make things worse, the company used their main reception email account that more or less everyone has access to. And finally, thanks for the amazing SpinRite. I used it a couple of times when I was in tech support. It really did get the company out of a couple of tight scrapes. Sigh.

Steve: Is that just too...

Leo: Guess. High. High, you're high.

Steve: Low, low. Oh, I don't know the number. Oh, that's okay, just guess it.

Leo: That's helpful customer support.

Steve: Well, and you know, I guess, what would you do otherwise? It would probably cause this poor customer service representative more trouble if he just had to say no, I'm sorry. Or maybe he doesn't want you screaming at him, or who knows what, what it would take...

Leo: Bizarre.

Steve: ...to end up doing this. And this also indicates that the support guy is seeing the number. It would be nicer if the system were designed so that he enters the number you give, and then it either says yes or no to him. So obviously there isn't that aspect of privacy being maintained. He has the last four digits, so he's able to say, uh, guess a little bit higher.

Leo: Little higher, little lower. You're close.

Steve: Oh. Oh, goodness.

Leo: Guess. I don't understand how that can happen. It sounds like a renegade customer service representative.

Steve: I would love to have an audio file of that conversation. That would be a keeper.

Leo: Guess. Ian Cummings [laughing].

Steve: I know.

Leo: Ian Cummings reports from Newbury, UK, another UK listener, that PayPal may not be a lost cause: Hi, Steve and Leo. Love the shows. I'm behind and catching up, but listened the other day to the story about only needing the last four digits of your credit card to get through PayPal's security. This was another story, by the way.

Steve: And remember we mentioned there that the problem is that the last four digits of your credit card is so commonly used that it's exposed everywhere.

Leo: It's on the slip. It's the one thing they give you. Well, it looks like they realized that wasn't a great idea, and I just saw this, quote, "Secure Web PIN for Customer Services: We're always looking for ways to improve your service and security, which is why, after March 31, 2009" - this might be, by the way, British PayPal. It may not be the same everywhere.

Steve: Good point.

Leo: "We will only discuss your account when you provide a Secure Web PIN. The last four digits of your bank account or card number can no longer be used to identify yourselves. When you want to call us, just log on to PayPal, go to the 'Contact Us' page and click on the 'Call Us' link. On the next page you'll find a six-digit PIN code valid for one hour. Quote that when you call us." It's a small step, but in the right direction. Keep up the good work, guys. P.S.: The reason I'm behind and catching up is Leo does so many good podcasts, and has gotten me into Audible, and my journey to work isn't long enough. He has not enough time to listen to everything. Hey, but the problem with this is it wouldn't have helped our previous guy because he couldn't log in.

Steve: Correct, he was unable to log in. So this is a customer support authentication loop where you log in, you get a six-digit PIN code, then you use that to talk to a human being.

Leo: Perfect.

Steve: If you're able to log in. And so what that's doing is that's solving the social engineering problem of pretending to be somebody that you're not. And they're now saying after March 31st you're not going to be able to use your credit card or your bank account digits. You're going to have to log in properly first and then use the PIN code which is a one-time password sort of one-hour expiration code. So, again, it's like somebody's listening, so that's good.

Leo: That's an improvement, but I don't know what you do if you don't - you can't log in. But that's...

Steve: Oh, you just guess. I don't know my PIN code.

Leo: Maybe that's why they have to guess. Well, we know this doesn't work if you can't log in. So...

Steve: [Sighing]

Leo: It's bizarre. Mark McSweeney, Concord, New Hampshire shares his Poor Man's VPN solution with us. He says: Steve, I'm a regular listener of Security Now!. I have a question about the security of what I'm calling my "poor man's VPN." I have a Linux firewall router set up at home. It has a SSH server running in it on an alternate port, not the standard port 22. What I do when I'm away from my home network is to use an SSH client to log onto my SSH server using a public/private key pair for authentication and set up a tunnel dynamically forwarding port 1080. I then configure my browser to use a SOCKS proxy to use localhost:1080. I've checked the IP addresses being reported using WhatIsMyIP.com, and the remote address is displayed. Is this method as secure as an IPSEC or SSL VPN? Your comments and feedback would be appreciated.

Keep up the great work with Security Now!. I immensely enjoy the show and learn a little bit every week, although some of the topics such as the recent show dealing with HMAC do sometimes make my hair hurt, so I need to listen to them a couple of times to really grasp what is being discussed. Hey, this sounds like a very clever - an SSH tunnel, very clever way to do this.

Steve: Yes. And it is - I would say it's every bit as secure as an SSL VPN. It's using a pre-arranged public/private key pair, so the client which is out roaming around is authenticating itself in a very secure fashion to the server. I loved it that Mark is not running his SSH server on the default SSH listening port 22...

Leo: Does that make a difference? Because I figure, if somebody's trying to find it, they just ping until they get an SSH response. They try all the ports.

Steve: Yeah, but that's...

Leo: That's more work.

Steve: Well, and the typical attack is not scanning a single IP across all of its 65,535 ports. The typical attack is scanning all the IPs for something listening on port 22.

Leo: Right, right.

Steve: So the idea, I mean, to answer your question, absolutely yes.

Leo: Okay.

Steve: The fact that he moved to an alternate port, you know, he didn't tell us, we don't want to know. But it's something between 1 and 65,535. So his server is listening there, and only he knows where that is. It's true, somebody could find it. But you really - it's just better not to be answering, not to be running the service on the default port. So then what he's doing is, the way his SSH client works, is that, when he runs it, it opens and listens on port 1080 of, for example, his laptop, out roaming around. So he's got the client side. He makes a connection to his SSH server at home. The client side opens and listens for any connections to the local machine, that is, the laptop's port 1080.

He then configures his browser. Instead of going out over the Internet, he sets up the SOCKS proxy in his browser to instead use localhost, meaning the own machine's IP, port 1080. So the browser connects to port 1080, which the SSH client is sitting in there listening for connections on. And of course only the browser there is able to connect. So that gives the browser a connection across SSH, which has been securely authenticated and is encrypted using standard TLS, which is SSL technology. That gets him to his remote location, where the traffic then comes out of the SSH server and goes out over the Internet. So he has come up with, you know, it's a little clunky to configure. It's not a general purpose, complete solution because it's only useful for things that you can get to proxy through SSH. Browsers are perfect for it, so surfing the web is perfect. But it's absolutely secure, and it'll work well.

Leo: Yeah, that's something called SSH tunneling. And you can get other things to use the SSH tunnel, too. I mean, the browser's a little easier than maybe other things. But this is actually a fairly - he didn't invent this.

Steve: No, no, no.

Leo: It's a fairly well-known way of doing this. And, yeah, it's very secure.

Steve: Yup, I just wanted to describe in detail to our listeners how SSH tunneling works.

Leo: Yeah, it's a great idea. I keep thinking I'm going to set this up. I never get around to it, though.

Steve: Yeah.

Leo: Jeff Harmon says he can't get his 16-bit software to go anymore: Thanks for the great security podcast. I'm a software architect responsible for the Internet banking application of a midsized bank, and it keeps me up to date on the current security threats and vectors of attack. That's great, Jeff. We're glad you listen. I consider myself to be pretty technical, and I love it when you dive deep into the bits and bytes. Occasionally I have the netcast playing while driving with my wife in the car. She laughs at me as I am finishing Steve's sentences or answering the questions Leo asks. She thinks we're all nuts. Keep it coming.

My question has nothing to do with Internet security, however, more of a problem that just kills me I haven't been able to solve so far. I've been having trouble with this for months now, ever since installing Service Pack 3 on my XP machines. I have a number of old 16-bit children's games that worked fine prior to SP3, but no longer load or install. The installation starts but hangs partway through with the WOW.exe process taking up a significant amount of CPU. I've tried letting that process run overnight to see if it ever finishes, but none of that seems to matter.

I believe the WOW.exe process is what makes it so that a 16-bit program can run on 32-bit XP. Is there something that changed with SP3 that made it so that the 16-bit games can no longer run? Is there some kind of security issue? Is there a setting that can be changed to allow this? I wouldn't do that on all of my machines, only on the machines for my children. But he would like to run those games. Do you know about that? What is that WOW.exe?

Steve: Well, WOW stands for Windows On Windows. And it is Microsoft's technology for essentially hosting an incompatible version of Windows on a different version of

Windows. There was something like it back in the old days. Remember when we had Windows 3.1, and then there was Win32s? That was sort of a - that was an introduction to the 32-bit API that Microsoft was in the process of bringing us over on NT. And so it was sort of a straddling technology. This WOW.exe process is, essentially, it's the 16-bit Windows API that is then hosted on our 32-bit platforms. I checked when I saw his note, because I've still got some 16-bit stuff, and I was able to run it on a machine both with SP2 because, like, the one I'm sitting in front of right now, I tried to give it SP3, and it choked on it. So I backed out of that quickly. But I do have a SP3 machine that in my case does run the 16-bit software.

The reason I brought up the question, though, is that what we're seeing is, we're seeing Microsoft finally beginning to say, okay, we're no longer going to support this really, really old stuff. As I remember, and you and Paul may have discussed this, but doesn't Windows 7 formally drop support for 16-bit software?

Leo: You know, I don't know. That would make sense. They still support 32, of course...

Steve: Yeah. I think I remember...

Leo: ...seamlessly. But I wouldn't be surprised, yeah.

Steve: I think I remember seeing that somewhere. So this is a place where virtual machines are your friend because I think we're going to be seeing more and more virtual machine technology coming to bear where it's necessary to run, either different platforms on existing platforms, or in some cases really old software. I was just yesterday, I got a copy of Fusion, VMware Fusion, running on my Mac notebook because I really want to use my DOS, my 16-bit DOS editor, BRIEF, over on my Mac. And I need to use the Mac because that's where that really cool PDP-8 minicomputer simulator is, is on the Mac. And but I don't want to use the Mac editor, I want to use mine. So I've got Fusion running, and I literally have the DOS from Windows 95, or 98, four point - that thing that ended in .2222, which was sort of the classic last version of DOS.

Leo: 6.22, I think, wasn't it?

Steve: Yup. Well, there was DOS 6.22. But then there was - they carried it on. There was Windows 95 and 98 and then Millennium. And but there is the DOS underneath Windows 98 because remember, those still actually booted DOS and then ran an environment on top of the operating system. So you could either boot to MS-DOS, or you could boot from them back into MS-DOS. Anyway, the point is that I'm using virtual machine technology to host a foreign operating system.

And so it doesn't sound like this is a big problem for Jeff. I mean, it sounds like he's wanting to figure out what's gone wrong and why. I couldn't find anything on the 'Net that said that SP3 killed 16-bit software. And I did verify that at in at least one case I'm still able to run 16-bit software under SP3, or that would be a real problem for me because I'm a dinosaur still running DOS apps. But virtualization technology is something to keep in mind because it's not expensive, and it's really become very robust and reliable.

Leo: Yeah, yeah. I think it's probably something corrupted when he installed SP3 because I haven't heard the same thing, that you can't - I haven't heard that problem.

Steve: Right, right.

Leo: But you're right, this is where virtual machines are wonderful, wonderful solutions.

Steve: They'll certainly solve that problem for him because he could easily install a VM and then put whatever version of Windows he wants to on that, just for the kids' games.

Leo: Yeah. Larry in Minnesota's scratching his head about open hotspot WiFi security. He says: I hate to beat a dead horse, but I'm a little confused about the whole cybercaf scenario. In an early episode, if my memory serves me, you had indicated that the personal HTTPS, secure HTTP transactions like banking in the workplace could be subject to a variant of an MIM, man-in-the-middle, attack by your employer since they control traffic between your PC and their outward-facing IP. Couldn't the same be said of public WiFi access points? Couldn't Coffee 'R Us, or a disgruntled, out-of-work techie who just took the crumby coffee shop job to get his wife off his back, or an evildoer that realized they're using default admin settings on the router, do the same thing, a man-in-the-middle attack? How could HTTPS, VPN, or for that matter any reasonable solution protect you from this? Have I missed something?

Steve: Well, it's a great question.

Leo: Yeah.

Steve: Because we have talked about - we've sort of talked about this in all kinds of different ways. So I just wanted to reassure Larry that, without explicit configuration of his client, that is, of his laptop, for example, in an open hotspot WiFi scenario, there is no way to perform a man-in-the-middle attack on SSL. I mean, that's the whole point of SSL. And this gets blurred, and the reason for this confusion is in a corporate environment there are companies that want to perform content filtering on all traffic, including SSL.

And I remember I had an extensive conversation with a company at last year's RSA conference who was offering this. And I said, so you're installing certificates in all of the machines in the enterprise. And he said yes, that's the technology we use. So what that means is that, instead of you accepting the remote server's SSL certificate, you're creating the SSL connection with your employer's or the corporate gateway. Then it's decrypting that little hop of security, doing content filtering, and then it's working with the regular SSL connection to the remote server.

So that is classic man-in-the-middle, quote, "attack," unquote. It's not an attack because this is something by policy that the corporation has established. And your system is cooperating because you have the corporate certificate which allows the corporation essentially access to your connection by virtue of it not - your connection is not going direct to the remote server. It's only going to the corporate gateway, where it's decrypted thanks to the fact that the corporate gateway has previously provided you, your client in the corporation, with that certificate.

So because that scenario is not the case when you're roaming around in an open hotspot, you will have a direct connection to the remote server, and SSL will protect you. It's only where, by corporate mandate, they want access to all, even encrypted connections, that your client gets a certificate that allows that to happen.

Leo: All right. Let's see, here. Paul Kucher from Ellicott City, Maryland offers a great example of hysteresis. This all started because I was reading the definition of hysteresis from this radio/television electronics dictionary that Dane gave me. And you came up with a definition having to do with a button press.

Steve: Yup.

Leo: He says: Hi, Steve. I've been listening to your latest explanation of hysteresis, the nature of a keyboard's snap action. I think I have an even simpler example with your home's thermostat. When the temperature rises beyond the set temperature of the thermostat, the heater turns off. When the heat begins to escape your home, and the temperature begins to fall, the heater does not come back on immediately after it reaches the set temperature. Otherwise it would be constantly switching on and off. Instead, the temperature decreases to a lower threshold, whereby it then begins to increase until it shuts off again. Hysteresis allows the heater to turn of and on at a minimal internal without constantly switching off and on while it tries to converge on the target temperature. Of course it has...

Steve: I thought that was...

Leo: That's good.

Steve: Yeah. I thought that was another great example. I don't know if any of us, if any of our listeners are as curious about mechanical things as I. But I was fascinated when I took the cover off of many different thermostats - it's not that I go around taking the covers off of thermostats, but for some reason I seem to have seen a lot of thermostats in my time. And maybe you've seen this, Leo. In some models of thermostat there's a coiled spring and a mercury switch, that is, a glass capsule with liquid metal mercury in it, and two electrodes at one end so that, when the capsule is tilted toward the electrodes, the circuit is completed by the mercury. When it's tilted toward the other end, the mercury leaves the electrodes and opens the circuit.

And what's clever about this is it implements hysteresis using the movement of the mercury. That is, as the temperature increases, that coiled spring is actually a bimetallic spring. It's two dissimilar metals with different coefficients of expansion on the inner and the outer sides of that coil. So as the temperature changes, the outer one, for example, expanding more, will cause this coil to wind itself tighter. So the outside begins to pull the mercury switch over center. And at some point it's just enough the mercury rolls to the other side.

Well, in the process it changes the balance. Now it's like the thing has switched itself on, and it's going to stay there. It's got to go much further back now in order to compensate for the fact that the mercury is over on one side to roll up back on the other. And so the result is hysteresis, with a really simple, very sort of physically obvious technology. I just thought it was very clever.

Leo: We got a letter from Allyn Malventano, who is a computer security guy in the Navy, who has done some stuff for us. He actually reviews solid-state drives for PC Perspective. He says - he has another definition. He says hysteresis - now, this is kind of more the radio definition. "Hysteresis is the tendency of a ferrous metal to retain some of its magnetism. It comes into play in electronics and radio gear, mostly by the use of transformers. Hysteresis is the major contributor to any power loss or signal loss to those passing through a transformer because of the residual magnetism of the core is always lagging behind the field induced by the transformer's primary winding. The constant realignment of the particles within the material causes the lost power to be given off as heat. That's hysteresis loss." So there's another...

Steve: Cool.

Leo: ...much less intuitive explanation.

Steve: Thank you for that, Leo.

Leo: He says it takes work for the metal to change poles, and that work eats up some of the signal efficiency of the system.

Steve: Yup.

Leo: And that's what makes unused wall warts warm to the touch. Did you know that?

Steve: Yeah, actually, yeah. They've got transformers in them, and the transformers are sitting there transforming, whether you need them or not.

Leo: Yup. That's why we unplug from the wall now, even if we're not charging. Glenn Edward in Nottingham, Maryland - actually especially when we're not charging. Is even AES and PGP secure? I wonder, queries Glenn. After reading stories like the one referenced below, I have to wonder if any data encryption algorithm that's allowed to see the light of day hasn't been compromised. What do you think after reading this? Some of this was covered in my local newspaper a few years ago: mediafilter.org/caq/cryptogate.

Steve: Well, his link refers to a story that you may have heard about. It was in the news, as he mentioned, a few years ago, where somebody was grabbed by, I think it was officials in Iran, and accused of - it was a Swiss official accused of planting deliberately compromised with a backdoor crypto machines that came from a highly reputable company. Crypto AG was the name of the company. And I don't remember now one way or the other whether it was ever proven. But the accusation was that the NSA was working secretly with Crypto AG to install cryptographic backdoors in their equipment, and that allowed the NSA to obviously decrypt otherwise unbreakable codes.

The reason I bring this up is that this is a perfect example of the difference between technology and implementation. That is, in Glenn's citing this example, asking is even AES and PGP secure, well, this is not - it never was the fault of the underlying cryptographic algorithm that was the problem. It was implementation details. It was the fact that the algorithm was being misused or abused or something about the system that employed the algorithm was deliberately being made insecure, if this is even true. I don't know whether this is urban legend or true or not. But that's not really the issue. I wanted to draw a clear distinction between the idea of the algorithmic strength versus the application strength.

And actually, Leo, this sort of goes to your point of liking open source software because it helps to bring visibility to the implementation, and it makes it much easier for people to look at the code and go, oh, this was done right. So we know we have both a really strong algorithm and a really strong implementation of the algorithm. And those are two very different things.

Leo: Yes. As we've learned. Tazz in Nova Scotia has some relief from PayPal concerns, just in the nick of time. He says: Steve and Leo, I'm a little behind on listening to Security Now! due to being busy at work. I'm catching up. I have listened to SN-182, and I have a little info for the PayPal football and security question dilemma. If you log onto PayPal and go to My Account, then Edit Profile, down at the bottom of the account information column there is something called Identification Preferences.

The web page the link points to says: "When you call customer service, we will ask you to confirm your identity by providing your primary phone number and one other piece of information. Please let us know what you would like to use for your other form of ID." The choices are customer service PIN, your Social Security number, the last four digits of the primary bank account number, the last four digits of the primary credit card number. Choice one lets you create your own six-digit, numbers-only, PIN. The default setting, I guess, it certainly was mine, it was the bank account number. If I only had a credit card registered, I'm guessing it would have been the default. It seems to me, just guessing here, that if Brian in Raleigh, the guy that wrote in at the end of 182, had set up his own unique PIN, the customer service would ask him for that instead of the last four numbers of his credit card. If that's the case, well, everybody should be going there and fixing that.

Steve: And Leo, it is. I didn't know about it. I went there, and it's there.

Leo: I'm going there right now.

Steve: Yes. It's exactly where he said. You log into PayPal. And if you can't you just guess, and they'll let you in anyway. Go to My Account, and then there's a little Edit Profile link. You choose that, and there's three columns of data. The left-hand column is called Account Information. The bottom of that is Identification Preferences. And sure enough, I mean, I had never been there before. And mine was defaulted, like everybody else's is, to I think it was the last four digits of my bank account number. Either that or it was my credit card, I don't remember which. And the option for customer service PIN was grayed out, non-selectable. But over on the right was a link to create one. And it literally - and so I went there, and it says - there's two fields that look like password fields. And it says, make up six digits. And so I did that, and put them in twice, and it said good. We got your own, self-assigned PIN. And then I went back to the prior page and selected it as my identifier. So I will no longer be using or asked for, hopefully, my bank account number and credit card number. And I wanted to provide this "Thank you, Tazz in Nova Scotia." This is a fantastic tip for all of our listeners who are PayPal users.

Leo: I'm logging in right now. That's great.

Steve: Yeah, this is just - it's an absolute way of increasing PayPal security. And lord knows, after what we've heard today, we need all the PayPal security we can get.

Leo: Just guess [laughing].

Steve: Just guess. No, a little too high. Guess lower. Oh.

Leo: Oh, sorry. Renee Ann from Birmingham, Alabama describes herself as one of our few female listeners. We don't know that. But I think you might be right. Dear Steve, I'm so glad to hear about the hands-on, how-to episodes you're planning. I'm embarrassed to ask what is probably a really dumb question, but here goes. In passing, you and Leo often make mention of making an ISO image of your computer. While I back up my data regularly, I don't have a clue as to how to make an ISO image. Could you discuss this, either in listener feedback or as part of a how-to episode, what's the best software to use to do this? And please talk about the smallest details of how to do this. Perhaps you could give us a recommended checklist. Delete old files, defrag, whatever. Profit. Thank you. A devoted female listener, Renee Ann in Birmingham, Alabama.

Steve: Well, my favorite imaging tool has changed.

Leo: Oh. I've been telling everybody Drive Snapshot.

Steve: Yup. And it had been until I was updating my copy of my favorite boot manager, which is BootIt NG. This is made by a guy named Paul Terrell, who I think is in Nevada, if I remember right. And he's got something called Image for Windows. And I like it better because it will image to NTFS partitions. Whereas Drive Snapshot is a DOS-based tool, this thing is no OS based. So you actually boot it itself. He will create a boot floppy or a boot CD for you, sort of very much the way I do with SpinRite, where you don't need to provide it with an operating system, it solves that problem for you. And it understands natively Firewire and USB, so you're able to use those; whereas, if you boot DOS, you have to have additional drivers, unless your BIOS provides recognition for those. Anyway, it's a little more flexible. It's not as easy to use. The UI is sort of text screen based, whereas Drive Snapshot is a little bit easier over on the Windows side. But I do like it. And I've been using it and recommending it to friends that ask me the question.

Leo: How much is BootIt NG?

Steve: BootIt NG is $34.95.

Leo: Okay, so it's about the same price, but you get this boot manager, too.

Steve: Well, actually they are separate. BootIt NG is $34.95.

Leo: Oh, okay.

Steve: The Image for Windows is $38.94. I don't know where that number came from, but...

Leo: They have Image for DOS and Linux, too. Okay, that's cool.

Steve: Yeah. I mean, it's really nice stuff. It works well. The guy knows his way around imaging and low-level stuff. BootIt NG is well known among our listeners, I'm sure, because it's been my favorite boot manager for quite a while. And it itself does some imaging, but not as flexibly as his standalone imaging tool. But to answer Renee Ann's question, those don't make ISO. When we talk about a drive image, we're not talking about a CD or DVD image, which are technically and typically ISO images. That's a sort of a different kind of image. That's an image of a CD or a DVD. In this case it's a drive image, which is a file, or multiple files. Sometimes you break them up into smaller pieces. If you're, like, using a FAT file system that can't manage a multi-gig file, it'll create a series of smaller chunks.

And I do find myself - actually I was setting up a new machine to try using Skype under Windows for this episode of Security Now! with Leo, and after I got it all set up I made an image. And so I do the things that you would do if you're about to sort of like make something you care about, like take a snapshot of your drive, which is delete all your temporary files from your browser. Under Windows there's always a tool under System Cleanup called Cleanup. And you can check, normally check all the boxes for the types of things you want to clean up. So I run a Cleanup. And it'll find and discard a whole bunch of stuff, and also offer to compress files that haven't been used for a long time to sort of keep them in a smaller storage form.

Then after all that, that is, after deleting all this stuff - oh, also empty your trash, if you use trash and have a trash can. Then basically get rid of all the junk that you really don't need to keep around. Maybe look at your desktop and delete, take that opportunity to do a little bit of spring cleaning. And then, finally, do a defrag to sort of get everything in nice shape. And then take an image of that. Which is the - that's sort of the routine I go through whenever I'm doing an image. Sort of get things shipshape before you take a snapshot. And use a program like Drive Snapshot, which I really recommend for less high-end users, or this Image for Windows, DOS, or Linux from Paul Terrell. He's at TeraByte, by the way, TeraByte. And I recommend his stuff without hesitation.

Leo: Good. Well, now I have another imaging program to buy.

Steve: It's a good one.

Leo: I've been using Drive Snapshot. I'm very happy with it. But this looks like it might be a little easier to restore from. I mean, the big issue is creating this boot disk to restore from.

Steve: Yes. And...

Leo: And this seems like it might be easier to do.

Steve: That's exactly the case, Leo.

Leo: Our last question. Hard to believe. Rafael Mediavilla in San Juan, Puerto Rico says: Steve, one simple question. Why aren't you on Twitter? Man, I thought we were going to have one show in this whole darn network that we don't mention Twitter [laughing]. No.

Steve: Well, I'm going to sort of help you with that, Leo.

Leo: Oh, good.

Steve: What is Twitter?

Leo: I have no idea. I can't imagine what he'd be talking about.

Steve: I'm Assembly code, and it is fundamentally incompatible with Twitter. I don't know what Twitter is. I don't want to know what Twitter is.

Leo: Good. I'm not going to tell you.

Steve: I don't care what Twitter is. I know that I'm the only one left on the planet who doesn't know what it is. But that's just fine with me.

Leo: Twitter grew in the last year at a rate of 1,320 percent.

Steve: And now we have tweeting, whatever tweeting is. Is tweeting the act of twittering? Or is that something different?

Leo: Yes. Yeah, instead of saying "twittering" you say "tweeting." And your post is a "tweet."

Steve: Yeah, and don't you say, like, things, oh, I just walked in the front door, or gee, you know...

Leo: Well, you can say a variety - I don't want to get in this discussion.

Steve: I saw the strangest looking cloud just now.

Leo: Yeah, you could tweet that.

Steve: But who cares?

Leo: But you don't have to tweet that. You could also say, I've learned of a new technique for cracking AES. See this link. I mean, who you follow really determines what kind of stuff you're going to read. You don't have to follow people who say I looked at the sky.

Steve: Just push away from the computer, Leo. Step back slowly and carefully.

Leo: I've tweeted a lot less lately. I'm kind of - I'm on the...

Steve: Is there an exponential decay curve in tweeting, some people tweet...

Leo: No. It's an exponential explosion, exponential explosion curve, unfortunately.

Steve: So maybe this is what everyone's doing when I'm seeing them all doing something in the restaurant.

Leo: Yeah. You see...

Steve: Everybody in the restaurant is, like...

Leo: They're tweeting. They're twittering.

Steve: ...bent over their little PDA, pushing buttons. I'm thinking, okay, what the heck are you doing? Oh, they're tweeting.

Leo: Yeah, that's almost certainly what they're doing.

Steve: There's this weird guy across the restaurant staring at me.

Leo: Yup.

Steve: That's me, yeah.

Leo: Rickster just said, "Steve Gibson just became my favorite person on the TWiT Network."

Steve: Thank you, no. I don't have Facebook. I don't have MySpace. I don't have any of that teenager stuff.

Leo: You don't have time for that stuff.

Steve: Don't have any twit, tweet, or anything else.

Leo: All right.

Steve: Exactly I don't. I'm happy to do Security Now!, and that's it. That's my exposure.

Leo: That's it. It wasn't easy getting you to do that.

Steve: No. When you first suggested it I was groaning, thinking, ohhhh. What is this now that Leo's come up with? Turns out the best thing that ever happened, Leo, so I'm glad.

Leo: Thank you. Well, we're very grateful to you because we learn a lot every single week. You could find, of course, the show notes and transcriptions and 16KB versions of all of this at GRC.com. And if you'd like to ask a question, we do these Q&A episodes every other show. You can go to GRC.com/feedback and leave a question.

Steve: Please do.

Leo: Yeah, Steve likes those questions. And of course, when you're at GRC, the Gibson Research Corporation, pick yourself up a copy of SpinRite, the world's best disk maintenance and recovery utility. It's a must-have.

Steve: Or get four, if you want a site license. Then you can use it on all...

Leo: Is that how that works?

Steve: Yup.

Leo: So if you buy as many as four, then you're done. You can install it now on all the systems in your office.

Steve: Yup, exactly. In a single site. We use 10 copies if you want a corporate-wide, multi-site capability; and 20 for global.

Leo: Perfect.

Steve: And I just liked it because it allows someone to try it and see if it does the thing for them. Then they don't have to say, hey, I already bought one copy, or I bought SpinRite, what's the site license plan? It's like, okay, wait a minute. Just if we make it incremental - and that way, similarly, when there's an increase, they can upgrade by upgrading that set of SpinRites. So it just sort of all scales properly. I just - I liked it. I thought it solved the problem nicely.

Leo: I think that's, yeah, it's very clever. Very clever. All right, Steve. We're out of time, my friend. But...

Steve: Oh, boy, are we.

Leo: Long episode, but full of good stuff.

Steve: Yup. Great to talk to you, Leo. And we'll do it next week.

Leo: On Security Now!.