SECURITY NOW!

Transcript of Episode #296

Listener Feedback #115

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-296.mp3
Quarter size (16 kbps) mp3 audio file URL: http://media.GRC.com/sn/sn-296-lq.mp3

Leo Laporte: This is Security Now! with Steve Gibson, Episode 296, recorded April 14, 2011: Your questions, Steve's answers, #115.

It's time for Security Now!. Yes, I'm back in the studio. Steve Gibson is back in his secure fortress, his secure undisclosed location. And we are going to discuss the latest security news, privacy news, and answer your questions, too. Good morning, Steve.

Steve Gibson: It's been such a busy week. We have lots of news to cover. And actually a new section added to our - in my own mind. Our listeners wouldn't know. But I do have an outline that I produce every week, and it's got major section headings. And normally we have Security Updates and Security News. I've decided, as a consequence of this week, and actually get a clue from what's been going on recently, we now need an Attacks and Breaches section.

Leo: Holy cow.

Steve: Because, I mean, there was a Texas-size screw-up in Texas by Texas. Adobe's got news. WordPress got hacked. And there was a third one, or a fourth one.

Leo: I love it.

Steve: Oh, and Barracuda Networks, a very large networking security company, got themselves hacked when they turned off their own firewall by mistake. So lots of interesting news this week. And of course it's a Q&A episode, so we've got some great feedback from our listeners to share.

Leo: Not to mention, and we're going to talk about it in just a moment, a record number of fixes from Microsoft on Patch Tuesday.

Steve: It is 64. I wonder if they tried to make it 2^6.

Leo: As long as it's not 128 next month and 256 the month after. That could really get out of hand.

Steve: Yeah, but they did fix some longstanding problems that we'll talk about. So, yes, it was a record-breaking Patch Tuesday.

Leo: Patch Tuesday, Steve.

Steve: Yes, yes, yes, yes. So this was, this most recent Tuesday, a record 'nother - we have been setting records. And in order to keep setting records, they have to keep getting worse and worse. And this one did.

Leo: It's not a good record to set.

Steve: No, it's really not what you want. 64 known security flaws were patched. The good news is, that persistent MHTML flaw, which we've talked about a number of times, which was being actively exploited in the wild, and which Microsoft, they took their time fixing this. I think we missed a prior Patch Tuesday, so it's been more - it's well over a month. And remember that this was the one where you could, if you saved a page that had scripting in it, then in a web page archive, which is an IE-only format, then this is the - MHTML stands for Mime HTML, which allows a single file to contain multiple assets of a web page, and this was where the problem was. So the idea was that...

Leo: It's the web archive; right?

Steve: Yeah, the web archive, which allows - it's Microsoft's way of bundling everything, all the different images and...

Leo: When you save a site, it says that. Save a site, you get a web archive.

Steve: Right. You're not just saving the text, though. You're saving all of the parts of the page in this single file. And so they had a mistake in that which they finally fixed. They also fixed the way that they were owned in the Pwn2Own competition. That got fixed. And we also referred to an SMB flaw, the Server Messaging Blocks flaw. SMB, most of us know it as file and printer sharing. And everyone made a lot of worry about that when this was unveiled.

I told our listeners, well, okay, it's a concern on a local network. But firstly, most ISPs are blocking those ports, the SMB port, which is 445, preemptively because of the longstanding history of problems before Windows had a firewall. And Windows now, for three versions, XP after SP2 and Vista and Win7 have all had firewalls that are there and on by default. XP's original firewall was there but not turned on, as Microsoft sort of creeps forward and adds security features to their OS. So that would have protected people. And being behind a router keeps you being safe from incoming, unsolicited attacks over that port.

So the real problem was local networking. If something got into a corporate or a personal network, it could use that to spread among the machines within that network. So that's also been fixed. In addition to 60 other things that are just your typical Windows and Office sorts of problems. We won't go into enumerating them all because people will fast-forward.

Leo: In the interest of fairness, I might point out that I just came in, and I have a security update from Apple, too, on all my OS X machines. Their second update of the year came out. They don't do it on a schedule. But this one fixes Safari, which I imagine has to do with that Pwn2Own contest, or maybe something more, and also a number of security fixes. Apple is not as forthright about what's going on as Microsoft is. But you should apply that update, as well.

Steve: Now, I almost have an update, if we had the hot tub time machine, Leo, I'd have an update which is expected tomorrow.

Leo: That's what we need, to go forward, not back.

Steve: That's right. We could prospectively warn listeners of coming problems.

Leo: Wouldn't that be good.

Steve: That would be a hot podcast, in that case. Not that this isn't. But...

Leo: Here's what's going to happen tomorrow.

Steve: The hot tub would heat us up further. So anyway, there will be, it's expected that Adobe will be releasing a fix for a new zero-day exploit. And I've got to take my hat off to Adobe. They may not be holding to this ridiculous quarterly update cycle - well, in fact they haven't from the moment they announced it. But they really do respond quickly. This zero-day flaw first came to surface early this week. And immediately there was a security notification. And just this morning they said they're working to get it fixed tomorrow.

So in our Attacks and Breaches category, this is where I put this, it affects all versions of Flash, or all current versions of Flash, across all platforms. And we are expecting a fix on tax day. Although I guess we get a deferral, don't we, until Monday because tax day's on a Friday, so you don't really have to have things postmarked until, what, till the 18th, I think. But otherwise, Adobe is supposed to be getting this thing fixed.

And Brian Krebs, a well-known security blogger and researcher and author whom we've spoken of often, wrote something that I thought was interesting in his own analysis. He said it's not clear how long attackers have been exploiting this newest Flash flaw. But its exploitation in such a similar manner as the last flaw suggests the attackers may have a ready supply of unknown - to everybody else - unpatched security holes in Flash at their disposal. And his point is that, essentially, this one just was slipped right in, I mean, slipstreamed, just moved right in and replaced the prior exploit which Adobe patched. And now there was another one. Which, and frankly, given the fact that so much code was written to create Flash prior to any real focus on security, I mean, I'm letting Adobe off the hook a little bit. Really they're big enough...

Leo: That's very kind of you, actually.

Steve: They're big enough, they really ought to have their act together more than they do because this is really causing problems, well, we know it's causing people problems because RSA lost their keys thanks to the prior one. So, yeah, not good.

Texas. Get a load of this. Get a load of this. Texas. The state of Texas itself revealed that the personal information of 3.5 million citizens, including their names, their addresses, and their Social Security numbers...

Leo: No.

Steve: ...and more has been exposed to the public. Ars Technica summarized some of the news. And Ars Technica wrote: "According to Texas State Comptroller Susan Combs, the data wasn't exposed by a hacker or a group of vigilante script kiddies. It ended up on a state-controlled public server after having been passed around between various state agencies."

Leo: Okay, that doesn't sound good, either. That's not good. There's a hard drive you might want to see.

Steve: Oh. "The data came from the Teacher Retirement System of Texas, the Texas Workforce Commission, and the Employees Retirement System of Texas, all of whom transferred the unencrypted data, against state policy" - but, oh, well, data is data - "between January and May of 2010."

Leo: But now here's the best part, kids.

Steve: "The information was only discovered on the public server on March 31, 2011, meaning it has been available for almost a year."

Leo: Almost a year. Ding ding ding ding ding. You're a winner. The world's longest security breach.

Steve: Oh. Combs said that other data, like, get this, date of birth and driver's license numbers, had also been exposed to varying degrees. And I did go back and look at the original source documents. And she said, well, but they were all kind of run together in a big stream. It's like they...

Leo: Oh, yeah, computers can't figure out.

Steve: Exactly. That's not difficult to parse out...

Leo: Nitwits. Nitwits.

Steve: ...that information. She said, finally, "I want to reassure people" - uh-huh, good, yeah - "that the information was sealed off from any public access immediately" - after a year, no - "after the mistake was discovered and was then moved to a secure location." Well, big deal. Combs said in a statement, "We take information security very seriously, and this type of exposure will not happen again."

Leo: Well, okay, thanks. That's reassuring.

Steve: Until it does.

Leo: Until it does. Geez. So, I mean, if a bad guy has your Social, your address, they don't really need your DOB, but now they've got that, too.

Steve: Oh, they've got everything. Your name, your Social Security number, your date of birth, your address...

Leo: Well, here's the message. If you were in those databases, the Teacher Retirement System, the Texas Workforce Commission, the Employee Retirement System of Texas, in other words, if you work for the state of Texas, you should be very - you should be on fraud alert right now for your identify theft.

Steve: Yes. I would say that's exactly the right takeaway from this is this is all the information required to perpetrate identify theft. So people impersonating you who are, like, for example, applying for credit cards, applying for various types of accounts, the good news is those applications are generally caught by the major credit screening bureaus. And so it would be worth...

Leo: Well...

Steve: Well, but, you know, it's one thing to check is to see whether there have been applications or queries against those that were not yours, or that were not people that you were applying for accounts and credit from.

Leo: I mean, given the incredible breach from Texas, I wouldn't count on this, but here's what Texas should do is notify all their employees and help them to protect themselves by putting fraud alerts on all of their accounts on Experian and TransMed, all three of the...

Steve: TransUnion...

Leo: TransUnion, Experian, I can't remember the third one. But anyway, no, this happened, Tech TV, they weren't sure, but they thought maybe some data had been released from the payroll system. And they, on our behalf, went to those three credit unions and said, this is what's happened. You have to have an excuse because credit unions don't like to do this because then they can't issue credit cards randomly. So you have to have an excuse. But if you say, look, this has happened, if Texas goes to those three and says that this has happened, please put fraud alerts on these accounts, they will.

Steve: And the nice...

Leo: It's the least they could do.

Steve: And the nice thing about this, Leo, is they have all the data. They've got the database of all the people's names and addresses and so forth. So they can just hand this over to the credit bureau agents and say, okay, we have 3.5 million people we'd like to put fraud alerts up for.

Leo: Experian, TransUnion, and Equifax.

Steve: Ah, Equifax.

Leo: Equifax. That's something anybody, everybody should know about. You can go to all three of them and say, I want a fraud alert placed on my account. Makes it very hard for you to get a loan because the people who are trying to get the loan on your behalf can't just run a credit report on you. But it also makes it very hard for a third party with that information to get a fraudulently issued credit card. So, very important.

Steve: Yeah, and I think you're able to ask for a free report at least once a year.

Leo: Once a year.

Steve: Once a year, yes.

Leo: Reporting bureaus, not credit unions, credit reporting bureaus, thank you.

Steve: Right.

Leo: TransUnion, Equifax, and Experian, I think.

Steve: Yeah, and I did - something happened to mine a while ago where someone - it wasn't identity theft. I can't remember, it was a credit - I lost my credit card, it got loose on the Internet through someone I purchased from who consequently had my name and address and so forth. And someone had something sent to a different address, like to them, and that's what raised the flag. It ended up all getting resolved and everything. But I remember then pulling reports from the credit reporting agencies at that time, just to make sure that there had been nothing else going on. So as a wake-up call, you can do it once a year, and it's worth just going through and seeing if there's anything that looks strange, that isn't your own activity.

Leo: There's a website called FightIdentityTheft.com that has an explanation of how to do fraud alerts. I am a subscriber to a company called LifeLock, which used to do this. And the problem is that Equifax, Experian, and TransUnion don't want to do this because what they really make their money on is selling your information to companies to issue you credit cards, which they no longer can do if you have a fraud alert. So they actually blocked LifeLock from doing this, which is unfortunate because LifeLock was just turning it on every 90 days because it expires in 90 days.

Steve: Ah, nice.

Leo: And it was a great feature. And yes, it meant when I wanted to refinance I had to jump through some extra hoops so that the loan office could get my information. I did not mind that.

Steve: Yeah, but so what, compared to the extra security.

Leo: So what, yeah. I wish I could have a permanent fraud alert because I don't want credit card offers in the mail anyway.

Steve: Yeah, that ought to be done.

Leo: Terrible. Yeah, it should be automatic.

Steve: So, WordPress, on our list of a very busy hack week. The WordPress hosting company, which is - I think it's Automattic with two T's. The first time I saw it, I thought that was a typo.

Leo: Because it's Matt Mullenweg's company. He's Matt, get it, Automattic.

Steve: Yup. And in fact I have a quote from him, in Matt's own blog, the founder of WordPress. He said: "Tough note to communicate today: Automattic" - with two T's - "had a low-level (root) break-in..."

Leo: Whoops. Whoa.

Steve: Yeah, "...to several of our servers, and potentially anything on those servers could have been revealed. We have been diligently reviewing logs and records about the break-in to determine the extent of the information exposed, and re-securing avenues used to gain access. We presume our source code was exposed and copied. While much of our code is Open Source, there are sensitive bits of our and our partners' code. Beyond that, however, it appears information disclosed was limited.

"Based on what we've found, we don't have any specific suggestions for our users beyond reiterating these security fundamentals," which are coming from Matt, but they will be very familiar to our listeners. "Use a strong password, meaning something random with numbers and punctuation. Use different passwords for different sites. If you have used the same password on different sites, switch it to something more secure. Tools like 1Password, LastPass, and KeePass make it easy to keep track of different unique logins.

"Our investigation into this matter is ongoing and will take time to complete. As I said [before], we've taken

comprehensive steps to prevent an incident like this from occurring again. If you have any questions or concerns, please leave a comment below or contact our support." So it is, I mean, the refrain we always hear is, oh, well...

Leo: Never happen again.

Steve: Yeah, we're really sorry, but we learned our lesson, and it's never going to happen again. It's like, well, yeah, okay. I hope that's true.

Leo: Wouldn't it be nice if companies, states, government agencies would learn the lesson from the other guy instead of letting it have to happen to them.

Steve: To them, yes. Yes.

Leo: Geez.

Steve: And finally, Barracuda Networks, which may be a name not known to our listeners, it's not like Cisco or Netgear or D-Link, because it's a big iron company. These are the guys, they produce - and sort of the focus of this is something called a WAF, a Web Application Firewall, which is used in datacenters and by companies, larger companies that are able to afford a very expensive, multiple thousands of dollars, like $15,000 are what these things typically cost, so-called "big iron" machine. A blog posting from Michael Perone, whose title is EVP and CMO - well, now, EVP's got to be executive vice president. What's a CMO?

Leo: Probably, usually chief marketing officer, which always makes me nervous.

Steve: Okay. I was hoping for something a little more technical than that.

Leo: Chief marketing officer means the PR machine is in full force.

Steve: Yeah, okay. So what happened was, what was compromised was names and email addresses, much as was the case with the hack last week.

Leo: Oh, last week.

Steve: Yeah, that was...

Leo: Who was that?

Steve: Somebody, I'm blanking on the name.

Leo: Epsilon.

Steve: Epsilon, yes. He wrote that their databases contained just one-way cryptographic hashes of salted passwords, which we'll be coming back to later in this podcast. All active passwords for applications remain secure. And so, quoting him, he said: "So, the bad news is that we made a mistake. The Barracuda Web Application Firewall" - I mean, and so what's a little bizarre is that the equipment they invented and produce and sell is supposed to stop what happened. This was an SQL injection. So "The Barracuda Web Application Firewall..."

Leo: This is really embarrassing, then.

Steve: It is, "...in front of the Barracuda Networks website was unintentionally placed in passive monitoring mode..."

Leo: Oh, somebody's in trouble.

Steve: "...and was offline through a maintenance window that started Friday night (April 8, 2011) after close of business Pacific time. Starting Saturday night at approximately 5:00 p.m. Pacific time, an automated script began crawling our website [searching for] unvalidated parameters."

Leo: That just shows you, it doesn't take long.

Steve: And this is what - he has some nice conclusions here. "After approximately two hours of nonstop attempts, the script [crawling our site] discovered an SQL injection vulnerability in a simple PHP script that serves up customer reference case studies by vertical market." So this was an obscure little script somewhere that wasn't, like, mainstream. It wasn't the big deal. It was just off on the sidelines. But of course that gave it a foothold.

"As with many ancillary scripts common to websites, this customer case study database shared the SQL database used for marketing programs..."

Leo: And there's the problem.

Steve: "...which contained names and email addresses of leads, channel partners, and some Barracuda Networks employees. The attack utilized one IP address initially to do reconnaissance and was joined by another IP address about three hours later. We have logs of all the attack activity, and we believe we now fully understand the scope of the attack. This latest incident brings home some key reminders for us, including that: One, you can't leave a Web site exposed nowadays for even a day (or less). Two, code vulnerabilities can happen in places far away from the data you're trying to protect. And, three, you can't be complacent about coding practices, operations, or even the lack of private data on your site, even when you have Web Application Firewall technology deployed."

Leo: That's amazing.

Steve: Yeah.

Leo: That is really - but, you know what, I think Barracuda responded quite well. As they should, they're a security company.

Steve: And they got kudos within the security community. Your point is exactly right, Leo. They got - some analysis said congratulations on the way you came forward. As opposed to, for example, RSA, who is also a high-profile security company, but they angered everyone for several weeks.

Leo: Right. They obfuscated it. They kind of were so - they felt so bad that they didn't respond appropriately.

Steve: Right.

Leo: And of course you feel bad. It's very embarrassing. But this also shows a couple of things that could be good lessons. One is that PHP scripts are all over the place, and if it gets run at any time, it's enough leverage for a hacker to get in. And I think, to their credit, it sounds like they had multiple databases, and their most secure databases - this was just the marketing database.

Steve: Right.

Leo: You want to separate databases, obviously. Because once you get access to a database, a MySQL database, you've got it.

Steve: Yes, you're able to tell it to enumerate its own tables, as we talked about back in our SQL injection episode. But in security news, I have great news.

Leo: Oh, thank you.

Steve: Apple is adding the do-not-track browser header to the OS, I can't remember, am I supposed to say 10 or X?

Leo: 10.

Steve: To OS X's Lion release.

Leo: So that'll be, oh, that's good, that's 10.7. Okay, good.

Steve: Yes. It's in the next - it's currently in the developer release, the pre-release betas. The do-not-track header option has been added, is being added to Safari. So that means IE 9, Firefox 4, and the next Safari will all have it.

Leo: Well, pressure's on Chrome, now, boy.

Steve: It really is. And I'm glad.

Leo: Well, they'll do it. They'll do it.

Steve: Yes, yeah. Okay. And we're going to talk about France a little bit, Leo, because you and I did talk about this during Sunday's This Week in Tech podcast that I joined you and John Dvorak and...

Leo: By the way, thank you for being on with Jerry and John, Jerry Pournelle. It was our gray-hair, our periodic gray-hair episode. You can't be on unless you have gray hair or no hair.

Steve: Or aging gray matter.

Leo: Or aging gray matter. And every time we do it, one or two people say, I don't want to hear old farts. But almost invariably, people are grateful because of the depth of knowledge, the context, the history that everybody shares on that show.

Steve: Oh, we were talking about ham radio and, I mean, all kinds of wacky things that, you know.

Leo: It's really fun to do that. And we do it every few months. And thank you for being a part of it because...

Steve: It was great.

Leo: ...it was really great. I really loved it.

Steve: So France has done something which I'm still holding back judgment. What they've said is that part of their data retention laws, which they're in the process of getting ready to start enacting and enforcing - and we know that data retention is already controversial because it puts probably an impossible burden on ISPs, the idea being that people who provide end-users connectivity to the Internet need to keep everything that their customers do. Well, that's easy for some legislator to drop the gavel and say, yeah, that's a good idea, that sounds good, let's do that. But, I mean, look at the bandwidth we have now and the amount of data that would have to be aggregated across - at that bandwidth across some length of time. I mean, it's just, I mean, there is just no technology for it today.

I mean, you can imagine at some point in the future this happens. There will be data aggregating service providers that will create big RAID arrays that timestamp when things happen. The problem ultimately could be solved, although at an expense that would never be low. It just seems like a bad idea. You could step back from it and say, okay, let's just then record the IP addresses, or the email headers but not the content, or there are things you could do to back away from it that make it substantially more feasible to do this.

But what France has said is they want the usernames and passwords and basically login information to be made available by people who provide services, like Google and eBay and so forth. And those companies are fighting back. And the issue that is interesting is that, as our listeners know, no responsible providers store passwords any longer. That's old school. If you see something that says, oh, your password is limited to 16 characters, as we've discussed, that's a warning flag that they've got a 16-character field in their SQL database, and there's spiders trying to crawl around their site, trying to get into it right now.

You don't want your password saved. You want a hash, and technically, yes, a salted hash of the password. That is to say, you want a one-way cryptographic function between what you enter and what get stored permanently so that, when you are asked to log in, you provide that again, and all your provider, Google or eBay or whomever, all they're doing is they're doing the same thing to the new password you put in that they did to the last password you put in, the valid one, and seeing if the result of the hashes matches up. And if so, that's cryptographically sure that you entered the same thing. In fact, because a hash is a lossy operation, there are other things that technically could hash to the same thing. But the hash is so large, typically minimum of 128 bits and often now 256, that the chance of a collision is just vanishingly small.

So anyway, I'm hoping that the French legislation said usernames and passwords just because that's the way they think of it from the user end. And once the legislators get some additional education about what all this means, they'll go, oh, well, I guess what we really just need is - and I'm not sure what it is they wanted. I mean, getting usernames and passwords potentially allows people with that information to impersonate those users, which is nothing that you should allow data retention laws to facilitate.

Leo: But that's what they wanted. Remember, it's law enforcement that's asking for this. What would they like? Well, they'd like to be able to log in as Tony Soprano, into his email system, until he has the brains to change his password, i.e., forever. I think that's what they wanted. I wouldn't hold out hope for them.

Steve: My mind wasn't even believing that that's what they could want.

Leo: Of course that's what they want. They want as much as they can get. Because remember, law enforcement's always thinking, well, it's bad guys we're going to do this to. But as civil libertarians what we're thinking about is, well, who defines who the bad guys are? And today it's crooks, tomorrow it's political dissidents, somebody who doesn't agree with the government.

Steve: So I have on my note here, or on my news, the question of Dropbox authentication and whether it's insecure by design. There's a blog posting that was made by someone who was just doing his own little forensic analysis of what kind of debris is left behind by the use of these kinds of remote storage systems. I had it here, and I wanted to mention that I know about the blog post. I'm going to cover it in detail next week because it was one of the things that I was going to do if I had time, and I just ran out of time. So I will cover this. I want to let everyone who's been sending me email and tweets about it that I do know about this posting, and I will figure out what this guy did and what it means and tell all of our listeners next week.

Leo: Thank you.

Steve: In my miscellaneous section, I wanted to note that Amazon has done something interesting. They've dropped the price of the Kindle to $114, if you allow...

Leo: With ads, yes.

Steve: Exactly. But I saw - TechCrunch mentioned it, and then mentioned, well, that they didn't think that was cheap enough. And I sort of agree. If they could get...

Leo: It's 25 bucks. That's not a whole lot of savings.

Steve: Right. If they could get down under a hundred, if they could get into the $99...

Leo: $80. And you see ads. You don't see ads in the books. You only see ads on the front page and when you're browsing.

Steve: Well, I was just going to say that it's less intrusive than I thought. So I wanted to say that, as I understand it, there's some sort of a screensaver. Their screensaver I guess is customizable. And only on the home page are there ads. But you're not being accosted by them, like when you flip the pages of your book and suddenly there's an ad there.

Leo: That would be more not - that would be unacceptable.

Steve: Yeah, that would be.

Leo: This seems to be not so bad.

Steve: I think it's not so bad.

Leo: I wish it were more than 25. But maybe that's because it's not so bad. They can't really expect to make more than 25 bucks per user. But isn't, after all, isn't the profit in the Kindle in the books, not in the Kindle itself?

Steve: I really think so, Leo. When I was thinking about this, it occurred to me that, wow, when I look at all the money that they've sucked out of me because I have Kindles, it's like, wow, I mean, they're making much more money on me selling me bits than they are the pieces of plastic.

Leo: Just imagine if they could manage to - I don't mind these ads - if they could manage to subsidize to the point where they give it away. For instance, there's a lot of people with an iPad or an iPhone who say, well, I could read my books on my iPad and iPhone. But now they can get a Kindle, they can add that because there's times you want a Kindle and bright light and so forth. And they're going to buy that many more books. I mean, I'm sure Amazon's doing these calculations. They know their business.

Steve: Yup. So I just wanted to let our listeners know that, with a little bit of...

Leo: Tempting.

Steve: It is, $114.

Leo: All right, I'm going to order one. I'll tell you how bad it is. But the thing is, you've got to think it's going to be $85 in about six months.

Steve: Yeah. Also, I've mentioned to you, I think probably offline, Leo, that there is, in the current build of the iPad - oh, and by the way, there was just a new iOS, minutes ago, came out from Apple.

Leo: You're kidding.

Steve: 4.3.2 is now available for the Phone, the Pad, and the iPod Touch.

Leo: Must be a bug fix.

Steve: Yes, it's just bugs and security. It doesn't appear to have any new features.

Leo: You know what's interesting, Steve, though, is they're releasing these updates - this is the second update in a couple of months. They're releasing these updates much more quickly than they used to. Clearly these portable platforms are now in exactly the same position that the desktops have been for a long time, which they need to be updated frequently to prevent security flaws.

Steve: Yup. I think that's very much the case. I enabled something that I mentioned to you called, well, I wrote down "developer gestures." I don't think that...

Leo: Yes. Yeah, because...

Steve: But I don't think that would be, I mean, it is at the moment for developers. But what I wanted to mention was that I was experimenting with cloning one iPad to another, that is, making a backup of my main iPad after I got all the apps organized and arranged and set up in subfolders. And then I didn't want to go through all that again because it's a pain in the butt to do that. So I erased one iPad, the second one, and then told it it was the first one and restored the first one's image, essentially, to the second one, just to see that that all worked. And it does. So it's a very nice thing to do. You lose some passwords, like your WiFi passwords and things you have to reenter again.

Leo: That's appropriate. You also lose...

Steve: Makes sense.

Leo: I found the folders kind of disappear, so you have to kind of rearrange stuff.

Steve: Well, mine didn't. I was able to...

Leo: Oh, you were lucky, okay.

Steve: ...clone the entire folder tree, which is what - that was really my goal. What I lost was, temporarily, I got it back, those gestures. And so what - and so my point was that you know something is wonderful and, like, the right thing, when, after you lose it...

Leo: You miss it.

Steve: You can't live without it. The gesture that I love most is just you just put all your fingers on the pad and squeeze. You sort of - as if you're squeezing the current app, and that's a replacement for the home button. And...

Leo: It's so sweet. It's so sweet, yeah.

Steve: Oh. It is just - it's the right thing. And it's also nice you're able to put your fingers on and lift up in order to do the equivalent of double-clicking the home button in order to get to the little strip of things you used recently and also the brightness setting and so forth. But I just wanted to mention the presence of those gestures. You need to use the Xcode. And when I talked to you about it before, maybe it was during the podcast, I don't remember, I bought it, I bought Xcode 4.something or other, and it just got updated for $5. But it turns out that the free Xcode 3 works, too; works also.

Leo: Oh, that's good. If you have the SDK on it.

Steve: Yes.

Leo: You have to have the iOS SDK. So it's not free because it's 99 bucks because you have to have the iOS SDK. I think that's the case.

Steve: Okay. What I did learn was that Xcode v3, which you can apparently get for free...

Leo: Right, connect.apple.com. You have to have a developer account. But I think you have to also have an iOS - check me, chatroom, if I'm wrong on this. I thought you had to have an iOS developer's license, a $99 developer's license, to get the SDK. Oh, no, it does include the SDK. Never mind. You're right. Sorry. Forget I said anything. Zipping it up. Chatroom confirms.

Steve: So I don't have one, and it worked for me. And let me just say, oh, it is the best thing. Once you start just squeezing the apps to make them go away, when you want to do the equivalent of a home button, you can't live without it.

Leo: So you have this on your - if you have a Mac, you have Xcode 3 on your disk, on your install disk.

Steve: Yes, it is. You're right, it's an optional install when you install it, or you can add it later on. So yes, yes, yes. I just wanted to bring it to our listeners' attention. And while going through email today, this didn't quite rate a Q&A slot. Jared in Western Australia says he wondered about Leo and his iPad. Now, I didn't know what the context of this was. But he says, "You guys on MacBreak are nutters. Did Leo actually take his iPad 2 everywhere, even to the bathroom on a cruise ship? I couldn't believe what I heard. Next thing would be Leo taking his iPad to bed with him. Sheesh."

Leo: I do. Don't you take it to bed with you?

Steve: That's where mine - I have got one in bed.

Leo: What are you talking about? Of course you take it to bed with you.

Steve: "Now, that's an Apple fanatic. Maybe you never heard of something, it's called 'privacy.'" It's like, okay, well. So I wanted to just acknowledge that I sleep with my iPad.

Leo: I got in bed last night, after getting back from Las Vegas, Jennifer was in bed, and I had to slide her iPad over to make room. So even my wife goes to bed with her iPad.

Steve: And this morning, before I got out of bed I grabbed it and read some news for a while.

Leo: That's the whole point is you take it with you every- in fact, Steve Jobs is very famous, when he was first approached about making tablets, he said, "I don't want to make a bathroom device." But he did. 'Nuff said. We won't belabor the point.

Steve: And I had promised a review of iPad 2. And I will only say this. I agree with your appraisal, Leo, that anyone with an iPad 1 need not get an iPad 2. My takeaway is that the only thing they did, and it is brilliant, is they beveled the back side so that it comes to the flat front. And bizarre as that is, I mean, that's all they did. But when you hold it, you swear to god it is, like, a third of the thickness.

Leo: I know. It's tricksy. I know. They're very tricksy.

Steve: It's bizarre. And I, like, I've turned it over so that I'm holding it, looking at the back of it. And sure enough, you don't get that same feeling. Now it seems like it's the same thickness. But when you hold it with that bevel, just the way your fingers work on the edges, it's fantastic.

Leo: They did the same thing with their laptops, the MacBook Air, to give it the same apparent thinness. It's very clever. They are smart, aren't they.

Steve: Yeah. I mean, it was simple to do; and, oh, what a difference in holding it.

Leo: Well, it's funny because at NAB I saw a number of iPad 1s. And they looked so thick.

Steve: They look boxy, they do, they look boxy.

Leo: Now, my conspiracy theory is, because Apple did this with the MacBook Air years ago, like two or three years ago, so they knew about it. They intentionally released the iPad 1 boxy so that they'd have something to do with iPad 2. I'm sure of it. They knew how to do this. They could have done this.

Steve: Well, we had a listener, Evan Drosky, who said - this ties into backups - "SpinRite saves me from my backup wakeup call. Steve, I've been listening to Security Now! for a few years and appreciate the job you and Leo are doing and can't wait for Wednesday afternoon" - or in this case Thursday - "to roll around for the next never-too-long episode." We may be testing that today. He says: "Okay, truth time. I haven't been backing up my system. I usually transfer my stuff from one drive to a larger one every so often as I need space, and keep one step ahead of the inevitable disaster. I'm sure you know where this is going. This time I waited too long."

Leo: Oh, boy.

Steve: "It started when I was ripping an unprotected DVD I have and was shocked to see that Windows Media Player was having problems playing back the file I had just created. Luckily I already had SpinRite, so I rebooted into SpinRite, ran it at Level 2. It found a few bad sectors right away and fixed them. No other problems were detected with the rest of the drive. I rebooted into Windows and Chkdsk came up to fix the file system. Okay, fine. Everything looked okay, and I went about my normal computing again. A few days later I rebooted my system and Chkdsk came up again for the drive and fixed some problems again."

Leo: Bad sign.

Steve: Uh-huh. And he says: "Hmm, not good. I rebooted into SpinRite the next morning, set Level 2 on its way, and left for work. When I came home, I was shocked at what I saw. This time the first two lines of SpinRite's graphical status screen were completely filled with red U's. This was bad. The detailed log told me that every 50 to 100,000 sectors or so, there was a bad one that was only partially recovered.

"Four days later, and my 250GB drive was finished. Looking at the results, everywhere there was data on the drive, there were bad sectors only partially recovered. It looks like every file on there could have been affected. I restarted into Windows and started copying my recovered data onto a newly purchased 1TB drive. Post more use of SpinRite, there was only one file that Windows could not copy." Oh, I see. So he kept using SpinRite and working on getting his data back, sort of like bouncing back and forth between Windows and SpinRite. He says, "And I can easily recreate that one ISO file myself.

"A preliminary look at my data shows that SpinRite has recovered everything off that old drive that was trying everything it could not to give up my stuff. I want to thank you for such a well-written and tenacious program. It's well worth the price to me for recovering 10-plus years' worth of my digital life. My next project will be to build myself a file server, RAID and all, for image-based system backups - one step closer to the 3-2-1 process that Leo endorses. I can now say I'm a backup convert. Thank you again." And not a moment too soon, either.

Leo: No kidding. He was very lucky, yeah.

Steve: Yeah. And so I just, I did want to make a comment that hard drives should never show any bad sectors. So he should have taken more warning when he ran SpinRite that first time and saw some red U's. That says the drive is in trouble. He got four more days of apparent bonus life out of it before it really began collapsing on him. And then it's a good thing that he was able to get everything he could off. But I would consider that really pushing one's luck. So if you do see the red unrecoverable sectors, SpinRite is telling you, well, I've done what you told me to, but don't count on this drive for much longer.

Leo: So "unrecoverable" means I can't read it, and I can't save it. I can't save that sector.

Steve: Exactly. And SpinRite doesn't make any conclusions. It doesn't rewrite it. It leaves it alone because there's - and this is a good thing because clearly there are tricks like getting the drive cooler, sticking it in your refrigerator for a while and then trying it, that do, often, surprisingly, allow data to get recovered that couldn't be otherwise.

Leo: But you do that once, and then you get a new drive.

Steve: Yeah. It's a lot of work.

Leo: Yeah. And it's not going to work twice.

Steve: Right.

Leo: You should get the data and get out.

Steve: Exactly.

Leo: Thank your lucky stars if you do get it. All right, are you ready, Steve?

Steve: You betcha.

Leo: Q&A time. Of course did I close the questions? I probably did. You know what I do, as we're doing this show, I immediately get them into the wiki because I want to make sure that people can read the show notes and even read the questions as we go. In fact, I don't know if I mention this enough, but we have a really great wiki, wiki.twit.tv. Because it's a wiki, we use our community to keep it up to date, and people are always putting information in there. Many of the shows have their own kind of people doing the show notes for them.

But since you send me - you're the only host that does this, Steve - complete show notes with all the detail and links, I just, because I have them, I do a little grep on it to clean it up and wikify it. And then I put it into - let me turn this on. I put it into the wiki, wiki.twit.tv slash, well, it's a long URL. If you just go to wiki.twit.tv, click the Security Now! link, you'll see all the notes are in there as we go.

So Question #1, right from the wiki, it's Beau via Twitter, so it's a short question. Steve, is there a do-not-track option in Chrome?

Steve: There is not at the moment. And we mentioned this earlier in the show, that the good news is the notion of - okay, now, he says do-not-track option. And what we're talking about is a do-not-track header, where right now...

Leo: Chrome has an extension to do this.

Steve: Well, it has, and that's what I'll address in a second.

Leo: From Google, yeah.

Steve: Yeah, but it's different. So what Firefox does is, with v4, or if you're using 3, as I still am, waiting for 4 to kind of settle down a little bit, using NoScript, NoScript has converted its do-not-track header over to what has now become the standard, which is just DNT: 1, which is not the default at any of the browsers. But in IE9 and in Firefox 4, or 3 if you use NoScript, and we believe now in the next release of Safari, the Safari that'll be part of the OS X Lion release, that will be available. That adds this header which requests not to be tracked, which is different from other technology which is available, for example, turning off third-party cookies, blocking persistent objects, using IE's tracking protection lists. There are other technologies.

What Chrome has is an add-on which saves your preferences. So the idea is, you opt-out on a advertiser-by-advertiser basis, or there are sites that we've talked about where they'll use scripting in order to visit all the sites and set the opt-out cookies for you. Then what Chrome allows, what this Chrome add-on allows you to do is to freeze those, essentially to not worry about those requests, those individual requests not to be tracked. Don't have to worry about them being lose somehow or reset. So I'm not nearly as excited about that as I am about just having our browsers say "do not track."

And everyone comes back, well, not everyone, but they say, wait a minute, that's optional. I mean, people have to agree to abide by it. And it's like, well, today it is. The direct advertising groups, the societies are trying to decide whether - how they feel about this, whether they're going to adopt it or not. They haven't said yet one way or another. But there's a sense that, if they don't go along with this voluntarily, that it'll be imposed on them legislatively. So I just think, given that the major browsers are supporting it, we now have a standard, thankfully. It'd be crazy for Chrome not to add this. So it doesn't do it yet, in the same way that the other ones do; but I think we'll have it before long.

Leo: Yeah. And just by virtue of the fact that everybody but Chrome is doing it in the browser, it's just going to not only have it happen, but it'll also mean that the federal government can then pass a law, if they need to. I mean, there'll be a lot more support for the notion of do-not-track with the system working.

Steve: Yes, exactly.

Leo: So it's just a matter, I mean, I'm surprised - Google did introduce, may have been the first to introduce this plug-in and this concept. It was either Google or Firefox. Google followed very quickly on. But it's a plug-in, it's not in the browser.

Steve: Right.

Leo: Bruce Powers in New Jersey wonders about CAs, Certificate Authorities: I see that Comodo has another bad CA event. Do you remove authorities from your browser, Steve? What disadvantage is there from removing one I don't know or use?

Steve: Actually Comodo did have two other problems, which they disclosed at around the same time as the bad one that we discussed before, although those were breaches of two additional subsidiaries of theirs. The breach occurred, but certificates were not issued. So it didn't create as big an issue. So I just did want - I wanted to cover that. I don't remove CA certificates, certificate authority root certificates from my browser. But I kind of think that I should. I mean, I think it's worth doing, depending upon who you are. I would like to see better technology for managing our SSL connections. And we've discussed this from time to time.

For example, it would be nice if our browsers cached the SSL connection details that we make, for example, when we're really connected to Google or to Microsoft or wherever. That is to say, we've talked about how there's actually a certificate chain. And every certificate has a unique serial number. So even fraudulently obtained certificates will have a different serial number. The only way to spoof a serial number would be to get access to the root CA's private key, and so far that hasn't happened in the case of certificate authorities. We know that it happened, for example, it's one of the things that Stuxnet did in order to spoof its driver certificates. They did get the private keys of two companies that were hardware manufacturers that was being used to sign their drivers. So it's not impossible for private keys to get loose. It hasn't happened yet in the case of an SSL provider.

But the idea is that a fraudulent certificate could be used to impersonate a website, for example, Yahoo! was one that was targeted immediately by whoever it was believed to be in Iran who used the compromised third party to issue themselves certificates for these high-profile websites. But if your browser remembered the certificate path with serial numbers that it normally uses to go to Yahoo!, a change in that would really be a red flag. It's not necessarily the case that that represents a problem because, for example, when I renew my certificate, when GRC's certificate expires, as it does every two or three years, I'm forced to get a new certificate, which I install on my server before, hopefully before the expiration of the prior one. Well, that will have a different serial number. So that would raise a red flag if such a system as I'm describing existed.

But people could then check to see, like, does this look like it's still GRC.com? What IP am I connected to? Is there any - or look at the certificate chain carefully. Are there any unknown intermediate certificate authorities in there? I mean, like, essentially do a sanity check to see whether something has changed because change would be an indication of there being a problem, and it's not something that is spoofable. Alternatively, you remove certificate authorities that you just don't trust, like the unfortunately often maligned, unfairly maligned, I should say, Hong Kong Post Office. I hope it's unfair.

Leo: I'm sure it is. We bring it up a lot.

Steve: We do. They're my favorite whipping boy. So have I ever visited a website that was over SSL, using a certificate signed by the Hong Kong Post Office? I don't think I probably ever have in my life. I mean, I don't speak Chinese. If I go to a Chinese site, I can't read the page that comes up. And the flipside of being spoofed by a certificate of an English-language site signed by the Hong Kong Post Office seems higher than what I would lose if I didn't have that certificate. So I could see some advantage to removing certificates that just really seem like they're not bringing me more value than they are bringing me the possibility of them being used to exploit the trust that they imply.

Leo: I guess the problem is you don't know - removing a root certificate doesn't just remove the Hong Kong Post Office. It removes everything that uses that root CA to certify itself.

Steve: Precisely.

Leo: And you can't - is there a way you could tell where the - no, because it's just as it comes in.

Steve: Exactly. So that, yeah, there isn't any way.

Leo: So if, for instance, Google used the Hong Kong Post Office as its certificate - the only thing you would get is this certificate isn't legit; right?

Steve: Right, right. You would get the notice saying that this certificate is signed by a party whom you do not trust. That is, you haven't said you trust it because that chain is not anchored by a certificate that you've implicitly trusted because you haven't explicitly deleted it. So you get that notice. And then the behavior from that point varies. And it's been getting tighter, I've noticed. It used to be that you could click through those warnings and say, yes, okay, fine, I trust them anyway. That's one thing that's probably a good change we're seeing is browsers are really beginning to fight back more and saying, eh, we're not going to let you go there at all, if you don't trust these people.

Leo: Oh, so that's interesting. So you might not be able to get to a site. But at least you'd know, you'd probably have some idea why.

Steve: Oh, yeah. You could definitely look at the certificate and see that it was signed by the Hong Kong Post Office and go, oh, that's right, I deleted that.

Leo: Yeah, yeah. So go ahead.

Steve: I mean, that's what I think, too. I think we could use some nice certificate authority management tools. The tool that I mentioned, the EFF runs this thing, the EFF Observatory. I've got it on my notes to spend some time and look into that, probably do a podcast on it because the sense I get is that they may have an answer to that question, Leo, who's signed by the Hong Kong Post Office, which they would have obtained by watching Internet traffic over a long period of time. And I think what they're doing is building a record of certificate chains in use on the Internet.

Leo: Yeah, that's what it looks like, doesn't it. That's great. One more reason to love the EFF.

Steve: Yeah, they're doing a great job.

Leo: If you use this, donate to them. I donate, we make a regular monthly donation [indiscernible].

Steve: And they take bitcoin. They take bitcoin, Leo.

Leo: They take bitcoin, but I send them American greenbacks.

Steve: Oh, okay.

Leo: I'll send them bitcoin, too.

Steve: My machine just generates bitcoin for me, so I send it off.

Leo: Yeah, we know. A lot of it, apparently. You have friends at Bitcoin. No, just teasing.

Steve: There are no friends.

Leo: There are no friends because there is nobody home.

Steve: No one to love there.

Leo: No one to love. Moving along, Question #3, I think. Let me just make sure. Yes. Lisa Matias in San Jose says, why do you like Java on Android? I heard you, Steve, and Leo speaking favorably of Android, the Google operating system for handsets, and its openness. But it seems to me, as a third-party developer, Android OS is the most closed system since it restricts me to only develop "glorified web apps" - Java, JavaScript, Flash apps. Well, I'll speak to that in a second.

It also seems strange that whenever Google needs a new type of processor-intensive app, as the Android guardians, they create extensions to their Java VM to support it. This is not an option that third-party developers have. Android apps are restricted to the virtual machine. In contrast, only iOS allows me to create native binary apps using the same API, libraries, and SDK that Apple uses for their own native apps that come bundled with it. Apple apps are not restricted to any VM space since they run natively, but are restricted in the app store. Note that, like Android, you could still develop and install your "glorified web apps" - they're not. I'm sorry, I can't let her say that twice. A web app is JavaScript, HTML, and CSS. Java, I think people often confuse Java with JavaScript, is a full-blown programming language. You have full access...

Steve: Beautiful language.

Leo: It's a beautiful language. It is not a "glorified web app" by any means. It's run in native code, well, not native, but it's code running on your phone, just like a regular program, just like all of the other Google programs.

Steve could easily write his own iOS apps in assembler - which is true - and publish them in the - I don't - I guess it's true. You could certainly write assembler code in Xcode for a desktop. I don't know if they have an assembler for the ARM A5.

Steve: I doubt that I would pass their API tester, which only allows you to use special things.

Leo: Yeah. I do not think that you can, in fact, write A5 for assembler.

Steve: Right.

Leo: So anyway, Steve could easily write his own iOS apps in assembler and publish them in the iTunes app store, an option Steve does not have with Android. So could you guys please explain why you like Android OS when it seems to be nothing more than a glorified web browser, like Google's other Chrome OS. Sincerely, Lisa. Steve? Do you want me to yell, or should I? You're not going to yell. You're going to be very nice.

Steve: Well, it sounds a little maybe that...

Leo: I think she doesn't understand stuff.

Steve: Well, and maybe a little evangelical. I mean, she seems to be very pro iOS and very anti Android. And so I was thinking back, well, okay, I mean, I really don't have an opinion one way or the other that she sort of seems to think I have. I mean, I'm really sort of an iOS fan boy, if anything.

Leo: She's referring to me, really, more than you.

Steve: Well, and what I think it is, I think she got a little confused because what I was excited about was just the concept of Amazon's Android VM that allows you to test Android apps in the web browser. And that is something that we talked about in the last week or two, which I think was very cool, the idea that you could do a 30-minute test drive of an app, play with it, see how you like it. I mean, it's one of the things that really annoys me about iOS is there's no way to try out an app. They're not very expensive. But I've got a lot of them that I wish I hadn't bought. But I don't know that until I buy it.

Leo: Right. So let me just address the technical issues, which is that Java, the way Java works, and it's a very respected language, works very well, is it does, it's write once, run many, because every platform that you run a Java program on does in fact have a virtual machine. So Google...

Steve: Yes, Java compiles down to an intermediate language, which you then have an interpreter, essentially, to interpret Java bytecode into the native platform. Which gives you this tremendous portability.

Leo: Right. And that's - many phone systems, BlackBerry uses this, as well, many phone systems use a Java virtual machine and Java for several reasons. There is a security value to it because all the apps are running within the virtual machine, and they're protected from the hardware. It's the same reason Apple, I guarantee you, does not let you run assembly language code on the iPhone. You're always running, in a sense, you're always running within an environment...

Steve: Containment.

Leo: Containment to protect you on the iPhone or on the Android phone. So that is not the - when people say Android is not open, that is not where the discussion lies. These are not web apps. They're genuine first-class apps. You have access to the same tools that Google has and uses. And many developers do in fact like writing Java code for platforms. So I don't think that is an issue.

Steve: Yeah, and when Java - I guess it was on Sunday we were talking about Gosling.

Leo: He's now working for Google, the guy who wrote Java.

Steve: Yeah. I mean, it is a state-of-the-art, very modern, tremendous little language.

Leo: And if you feel like it, you can write Android code in other languages, including Python, C, or C++. They just have little plugs, little bridges that make it run. Still within the VM, which is the tool that Google promotes, is an amazing tool that does allow you to write in other languages and so forth. So I think that any complaint that it is a web app is not accurate. Nor does Apple have some sort of advantage by not using a Java VM. I don't think so.

Steve: Well, and market forces and popularity is showing us that Android's OS is doing just fine on similar platforms, with similar processor power and similar battery life.

Leo: Now, she says access to libraries. I don't know if Google writes libraries that they don't release. I don't believe that's the case, either. Android is not fully Google's. It is the Open Handset Alliance's. Google does write code that they then later release into the open source. But you can go to GitHub right now and get the full source code for the current version of Android, which is Gingerbread. Full source code, and I presume full libraries, or it wouldn't run. So I don't think they're secret libraries. Can I tell you something? Guarantee you Apple has secret libraries. There's no doubt, because Apple can do things on the iPhone that no one else can.

Steve: Oh, yeah, it's a completely locked-down DRM'd platform.

Leo: Yeah. Now, [indiscernible] asks an important question: Is performance on Java on Android comparable to native performance? Yes. Because it's hardware optimized for it. So I don't think that you're seeing a performance hit at all. Now, Lou, who works for Microsoft, says Android does have operating system APIs they don't publish, which is why some applications require root. But, see, then I wouldn't - I don't think so, Lou, because then you would be able to download Gingerbread. Well, maybe there are, so maybe Gmail and some of the proprietary apps have access to - no, because you have the source code. I don't...

Steve: They'd be open, yeah, there's no way to hide that.

Leo: If you have the code, there's no way to hide that. There's no way to hide that. Apple absolutely hides that. They have many undocumented libraries. So, sorry, Lisa, nice try. Question #4, Dan - we could keep this conversation going. It's so funny because I - here's the bottom line. Why be religious about operating systems?

Steve: Right.

Leo: Makes no sense. There are pros and cons to everything. They're just computer operating systems. You may like or not like a company, but the company couldn't care less about you. You can love the iPhone. It doesn't love you back. So why be dogmatic about it? Everything has advantages or disadvantages.

Steve: I'm pretty sure my iPad loves me, Leo.

Leo: See? See? Steve's smart. It's that thin little beveled edge. Dan Hummon - or Hummon - in Pennsylvania, on Question #4, offers a minor nitpick about our password discussion in 294: Steve, I just wanted to drop a quick note about how passwords are stored in databases. I totally agree with hashing and salting passwords - we just talked about that - but I think you left out an important final step. When choosing a hash algorithm, make sure it is one that has some significant computational load

associated with it. Oh, this is an interesting point.

The one I personally use is bcrypt. If there's a small, but real, computational cost to hashing one password, then if the database is compromised, brute force attacks against the entire stored database hashes are much more difficult to accomplish. I only mention this because I know many up-and-coming web programmers are listening, and I want them to have the best possible tools available to them. And he points us to a website, CodaHale.com, and the page there - Coda Hale is I guess a programmer, and his website has a page called "How to Safely Store a Password." Yeah, you wouldn't use rot13.

Steve: Well, it's more than that. And this is something I've been looking at sort of myself for different reasons. The idea is that the hash algorithms that we commonly use -MD5, SHA-1 and so forth, even the more recent stronger ones, SHA-256 for example - one of their benefits, one of the things, one of the reasons we chose them is that they're fast.

Leo: Wasn't SHA compromised?

Steve: Well, both MD5 and SHA-1 have had some sort of semi compromises, meaning that, as cryptanalysis has gone further with them, it's been possible to play some games with them. For example, you wouldn't use SHA-1 today in a state-of-the-art product. You would absolutely use SHA-256. But they were - one of the original design criteria was that they're fast, that is, that it's very quick to hash something because you might be hashing, and typically are, large blobs. So the algorithm itself needs to run very quickly.

Hashes are not, except for the purpose of obscuring a password, hashes are not used on something smaller than the hash itself. Remember, the hash produces 256 bits. You might be putting in a password actually smaller than that, and it expands it to 256 bits because anything that comes out of the hash is always the same size. So typically hashes were designed and chosen for speed. Well, what that means is the flipside is, from a brute-forcing standpoint, it is much easier to brute-force a cached password which is based on a fast hash because you can do many more of those brute-force tests per second when the hash, even though it's a cryptographic function, it just very quickly produces a result. So you can try another one and try another one and try another one and try another one.

So what Dan is saying is, and he's exactly right, state-of-the-art protection deliberately slows this down. Bcrypt is a solution which uses, I believe it's blowfish, it uses the blowfish key schedule, which is a time-consuming thing to set up. It uses that in order to produce a deliberately computationally intensive process of going from password to hash. So it's on purpose it's slow. And actually, what's cool about bcrypt is it's scalable. As processors get faster, you can turn the workload up to slow it down so that there's a relatively fixed amount of...

Leo: That's funny.

Steve: ...of cost, yes, that deals with the increase of speed in processors over time. Now, I'll take this one step further because I have been reading a lot about hashing lately, or passcode, or password, passcode protection. The problem you still get, then, is with FPGAs, Field Programmable Gate Arrays, basically hardware. The idea being that GPUs, Graphics Processing Units, in PCs are amazingly powerful. But people who are serious about cracking passwords have field programmable gate arrays, literally turning that process into hardware so that it is very fast. And interestingly, the way people on the leading edge of this have dealt with that problem is something called "memory hard" problems. One thing that field programmable gate arrays don't have is vast amounts of memory. And so what Dan was talking about is doing an algorithm which is computationally intensive.

But imagine an algorithm which is memory intensive, that is, where in order for it to function, and there's just there is no way around this, it has to be given a big, like a gig of memory, a big block of memory. And it has to have it all to itself for some length of time in order to do its job. And there just isn't - you can demonstrate cryptographically there is no way to do this without it having access to all of that memory. And those are called "memory hard" problems. And so the state of the art in protecting passwords involves, not just something that is computationally intensive, but is deliberately memory intensive. And what that prevents is it prevents people like the NSA from just doing this in hardware. You'd have to have vastly more memory than is practical currently. Which I think is a very cool insight.

Leo: That's a great idea. CodaHale.com to read about using bcrypt to hash your passwords. Yeah, that's kind of clever. I like that. Let me see here. Did I close your window again? As I surf around, I keep closing your window so I can't - let's see. There it is, twit wiki. I'm reading it out of the show notes these days. Dalvik, by the way, we were talking about the Java Virtual Machine on Android phones, it's called Dalvik, which is, I think, a Finnish fishing village. And it's open source, as well. The virtual machine itself is open source. I find that kind of intriguing.

Steve: That's really neat.

Leo: Yeah. Russ - oh, and LouMM, who works for Microsoft, says that there are secret, or let's not say secret, there are closed libraries used by some of Google's own closed apps on the Android phone that are dynamically loaded, they're downloaded, which is why they're not in the open source.

Steve: Ahhhh.

Leo: But of course if you could only do - the closed source apps would be the only apps I think that could use those closed libraries. So, yes, I guess to defend Lisa a little bit, it is true, it is the fact that Google, or it could be the fact that Google has proprietary libraries they don't disclose. But I don't think that gives them magical powers over the Android operating system.

Steve: Not when they're running on top of an open source OS.

Leo: Right, right. Glenn Edward in Nottingham, Maryland - nope, sorry. We'll go to Question #5, Russ in Austin, Texas, he's next, takes issue with your comments about Microsoft Windows being a toy operating system.

Steve: Boy, that really did generate some fur, some fur flying.

Leo: Oh, I bet you got some. But you knew it would.

Steve: Oh, yeah.

Leo: Them's fighting words. I think it's unfair to criticize Windows for having hundreds of files and modules, as well as distributed development teams. Are you saying every other "real" OS such as Linux, BSD and others are made by a single team of programmers that handle all aspects of the OS? I know this is not true, and I know there are tons of files as part of the distributions. I also think it's fair to differentiate a consumer OS such as Windows - unfair, or fair, he says fair - to differentiate a consumer OS such as Windows 7 and Windows 2008 and their roles. Oh, yeah, to be fair, these are consumer OSes, of course. Lack of proper configuration and maintenance of an OS will leave everyone vulnerable. IT professionals working at companies need to be responsible for their configuration regardless of OS. It's unfair to imply that BSD, Linux or others would be secure with no additional configuration or maintenance out of the box.

Steve: So all of the people who took issue with my use of the word "toy" brought me to some pause. It's like, okay, well, what did I mean by that? When I made the comment, it was the idea that RSA, their most secret crown jewels could be exposed by somebody opening a Microsoft Excel spreadsheet that had an embedded Flash movie in it, and that that let the person get into their network. So my frustration is that, and one can imagine this is no longer true at RSA, but it was true once, it was true the first time. So my annoyance is that RSA's original network architecture was such that there was no division, I mean, no absolute unequivocal division that prevented someone pulling a rogue piece of email out of their trash and opening it and allowing a bad guy to get in.

And I do, I'll defend my lack of respect for today's operating systems. These are consumer toys. I mean, it is possible, computers obey strict rules, it is possible to have an absolutely bug-free, bullet-proof system. It's very expensive. And we don't have any. And I didn't mean to imply that BSD and Linux were necessarily different. We're surrounded by toy operating systems. Unfortunately, that's all there is for us to use because it's just too expensive. People say, oh, all software has bugs. I cringe when I hear that. It's, yes, it's true, and absolutely unnecessary, yet it's true. Because it's too expensive to do a perfect job. There's no money in doing a perfect job.

Leo: Do you think it's even possible?

Steve: Yes, Leo. I mean, c'mon. It's math. It's processors. These things obey rules. There's absolute, I mean, I guess I feel so passionate about it because I live down in assembly language where nothing is hidden. I've been pulling my hair out for the last couple weeks in JavaScript land. I will soon have a JavaScript machine to show everyone, all of our listeners. But oh, my god, what a catastrophe JavaScript in the environment is.

Leo: Oh, it's a nightmare, yeah.

Steve: Yeah. So, I mean, everything I do I have to fight and struggle, and it's not compatible, and it runs over here but not over there. I even ran across where my own development environment, something was not working there, but it does when I'm not in the development environment. It's like, oh, my god. And nothing is the same between browsers. There was one place where every browser I tried interpreted something slightly differently. So, oh, goodness, it's just a - and I'm so unused to that because I'm down in assembler. And one thing that Intel has been good at - not perfect, but good at - is the Intel machine language is uniform across their chipset.

So of course, I mean, by definition it's possible for us to have an absolutely bug-free environment and not a bug in any apps. It'll never happen. But it's absolutely possible. And it bugs me, it annoys me that we're sitting here, some guy turns his web firewall off and a spider marches into his website and crawls around it for a few hours and finds an unchecked SQL phrase and leverages it. I mean, this sounds like science fiction. It's not science fiction. It's true. It's happening all the time. And does it seem it's happening more often, Leo?

Leo: Oh, yeah. You now have a section called Breaches and Vulnerabilities. We're making a special feature on the podcast just for that.

Steve: Wow.

Leo: Yeah, I mean, PHP is pretty notoriously bad for this, and so much of the web is written in PHP. But it's programmer education. It's things like validating your SQL queries.

Steve: To a huge, I mean, again, when I let Adobe off the hook, it's because I recognize that all of that code in Flash was written with no concern for security. I mean, just like Windows was originally written with no concern for security. That doesn't mean that lots of bad decisions or decisions made for expediency's sake haven't occurred since. They certainly have. But still we are dealing with a huge legacy that only very slowly gets moved forward.

Leo: Sanitize your inputs, folks. Silver lining from Glenn Edward in Nottingham, Maryland, an observation about this recent April 2011 Windows Patch Tuesday, the one we just talked about, with 64 fixes. One thing you can say about this month's Patch Tuesday is the majority of the vulnerabilities that are being patched exist in Windows 7. Okay, that's, wait a minute, I'm trying to figure out the logic here. That means either, one, no more faults exist in Windows XP [laughing]; or, two, Microsoft isn't bothering to fix Windows XP faults now; or, three, hackers are abandoning XP for the more exploitable playground of Windows 7 and Vista. A silver lining for those of us still using Windows XP, like you, Steve, is of course we may finally be slipping off the radar, hacker attack-wise. Even if it's not true, it feels kind of nice to believe it. Almost like running Linux and knowing no one's actively after your system. What do you say to that?

Steve: I think there's something to it.

Leo: Really?

Steve: I mean, I've - oh, yeah. I've talked before about how I've got some friends who are on Windows 98 because it does what they need, which is minimal, email and web surfing, and there is no - no one is attacking Windows 98 anymore. And it is the case that the target is a moving target upstream, and that we also know that new code is inherently more vulnerable than old code. And I think it's no coincidence that we will be and are beginning to clearly see a differentiation between attacks against XP and Windows 7 and Vista. It's the newer stuff, the newer browsers. IE9, for example, no, you can't have any IE9 problems in XP because IE9 won't run on XP.

Leo: That's true.

Steve: So as we move forward, we really do - it's not that we're leaving behind perfect code. We're leaving behind much more well-tested code, and no one is messing with it any longer. It has a chance to just sit there and not be changed all the time because change is the enemy of security.

Leo: I would just like to point out that Microsoft's end-of-life date for Windows XP was 2009. You can get extended support, but you have to be a business. The reason they're not pushing fixes is because they end-of-lifed that operating system. There's no way in the world that it is secure, that they got all the stuff fixed. And they're not pushing fixes because they said we're not going to push fixes anymore. That's why.

Steve: Yup.

Leo: I wonder, though, I mean, I have to say, if I were a hacker, I would look at what is the percentage distribution of the various versions of Windows. You certainly do go after the one that's in the majority.

Steve: But they're still fixing SP3. XP/SP3 is still being fixed.

Leo: You do have to have SP3. But I just think there - I think there are a ton of unpatched XPs running in closets in corporations all over the world.

Steve: Oh, god. You take an original XP [laughing]...

Leo: And no one's paying attention to those. So you hack them, and it's yours forever.

Steve: True.

Leo: I think, if I'm a hacker, I'd go after XP. But if you have SP3 you're all right.

Steve: If you took an original XP and stuck it on the Internet, it's hysterical to see how quickly it succumbs. I mean, there's still Code Red and Nimda are out there wandering around, making random pokes.

Leo: So you're saying, if it's SP3, you feel comfortable with it. That came out 2008. Support ends two months after the next service pack release, or at the end of the product support lifecycle, whichever comes first. 24 months. So I think that also ended, but I may be wrong. It's hard to tell. This Microsoft table is very obtuse.

Steve: Yeah, it is.

Leo: Review note, it says. Review the note. All right. Well, I'll move on. I hesitate to say, oh, we're safe now. Nobody's paying any attention to us. John O. in Argyle, Texas found debris in MRT: Steve, I enjoyed your nice discussion of Windows Malicious Software Removal Tool, MRT in Episode 293. You might add a note on the next show about where MRT logs are stored. It's in C:\Windows\Debug\mrt.log. Oh, that's good to know.

Steve: Actually it's very cool. I don't think I ever noticed that MRT was leaving a log. And I would commend our listeners to go look at it because one of the problems with MRT is that it's so quiet. I mean, you don't know that it's running all the time, except this log details every single time it has launched, and what it found, and what it did. It's just fantastic to have that. So it's just in your Windows directory. There's a debug subdirectory underneath Windows. And there's a collection of little logs there that Microsoft builds very quietly in the background. And there's an MRT, there's something else related to MRT. There's two different MRT-related logs and some other things. And I thought, wow, cool. And mine is huge, 660K was one of the logs because, I mean, literally every time it's run, it's left a little note behind. It's like, hey, I ran on this date, and everything was fine. It's like, wow, very cool.

Leo: Extremely useful.

Steve: Yeah.

Leo: Logs, as every security guru knows, are your best friend. And that's immediately what hackers modify. Soon as they get in there they...

Steve: To remove trails.

Leo: ...take the log, take down the log. Question #8, Craig in Chicago wants you to help put pressure on Yahoo!. Hi, Steve and Leo. Steve, I've been with you on

SpinRite since the mid-'80s. Yeah. And, yes, I've used it many, many times and have referred SpinRite to many over the years.

I need to ask you two a favor. I've been a Yahoo! user for way too many years, and I have for the last five years been sending them requests to go SSL for the entire session, as Google and now Facebook does. But apparently they just don't care. If you could talk about this and ask all who have Yahoo! accounts to demand they get their act together, there's no reason for them at this point to keep their customers at such a high level of risk. I do understand why they lost their lead. And if I weren't so entrenched, I would just move. But there's no easy way to move years of emails and other things. I do pay them for a premium service. What a joke that is. I also pay for premium Yahoo! mail. I guess I pay for no SSL.

So maybe if, with the quality and quantity of your listeners being what it is, they might finally get the message. But of course after they've been deaf to the world for the past 10 years, it's probably wishful thinking. You and Leo are the best. I've been listening since the very first podcast and can't thank you enough for all you do. Signed, Craig.

Steve: Well, so I never somehow got the Yahoo! fever. I don't know. I guess I was using my own email server and clients. And then Google came along with Gmail, and that seemed to be interesting enough to pull me in to experiment with something else. I have two observations. First, anecdotally, any time I have played with Firesheep, as I had occasion to a few weeks ago, remember, when Good Morning America wanted to find out what was going on. And I went over to the UCI-located Starbucks. By far and away the most intercepts that Firesheep finds is Yahoo!.

Leo: Oh, boy.

Steve: More than Facebook. I would have expected Facebook. But Yahoo! just pops up like crazy. And obviously it's non-SSL. That's the problem is that these credentials can be stolen because Yahoo! is not SSL. And as a consequence of Craig's note, I thought, well, okay, maybe I could recommend that he use one of the force HTTPS solutions. We know that our friends the EFF have HTTPS Everywhere, and there is something called Force HTTPS. Turns out Yahoo! resists that. It is specifically mentioned that, for example, HTTPS Everywhere from the EFF cannot do its magic on Yahoo!. Yahoo! fights it.

So presumably, I'm sure, you are secure during your login. But as we know, the whole deal with Firesheep is that it grabs your cookie, which is still being sent to keep you authenticated during a non-SSL post-login period, during which time it's possible to easily grab the session, because it's only represented by that cookie token, and impersonate the user.

Leo: So even if you have an SSL login, it doesn't matter.

Steve: Correct.

Leo: Because the person using Firesheep in the counter over there could just be you.

Steve: Right. And I did see some people posting in Yahoo!, I mean, Craig is not alone in recognizing that Yahoo! not offering SSL is really a problem. And so, as much voice as this podcast gives us, Leo, I'm glad to raise this because this needs to be fixed. Yahoo! should either go away or fix it.

Leo: Yahoo! is the - now I understand. Yahoo! is the No. 1 place people report getting their email account hijacked.

Steve: Ahhhh.

Leo: And that's the one where you get the email from somebody you thought was your friend with pertinent comments and information in the email that only your friend would know, saying, in my case, it was our gardener...

Steve: Oh, yeah, you told us about that.

Leo: ...I got robbed in England, and I lost my passport, credit cards. Please end me a thousand bucks so I can get home, and I'll pay you back the minute I get home. And that was a Yahoo! account that had been hijacked. And I've often wondered, well, is it a bad secret question? Is it a bad password? Now I'm starting to think it's Firesheep.

Steve: Yeah, or, well, I mean, Firesheep made this way more easy. But it has always been possible, I mean, any time you did packet sniffing in an open WiFi, you would see all of this Yahoo! nonsense going by.

Leo: But you wouldn't get the password because that is SSL.

Steve: True, but you do get the cookie that then allows you - if you just start using that cookie, it thinks you're the legitimate logged-in person.

Leo: So you're sitting in a coffee shop next to somebody. You use Firesheep. You're able to look at their email, read enough of it to get some pertinent details, and send an email to everybody in their address book. You don't need to be able to log in after that.

Steve: And you can send it as them. When you send it, you're sending it from their account.

Leo: Right. So now when somebody calls me, and that has happened to them, I will say I know exactly what happened. And I think it's almost certainly Firesheep.

Steve: Yeah. And I haven't looked recently at the download count, but it was 1.3 million copies of Firesheep, 1.3 million copies. And again, when I turned it on in an active location, I see more Yahoo! people than anything else.

Leo: Right. Somebody's saying there's phish, it could also be phish. That's true, you could get a phishing email that says, this is Yahoo!, somebody's been accessing your account, please log in and change your password, which would be also another way to capture it. Usually, my experience with Firesheep is you can impersonate the person. But changing the password almost, I don't know about Yahoo!, but I would guess, almost everybody says give me your password before I change the password. So Firesheep users I don't think, in most cases anyway, I mean, can change your password. They just can see, they can just be you for a while.

Steve: Yeah, and that's bad enough.

Leo: Bad enough. Question #9. You can see the whole article at cs.unc.edu/~fabian. He's talking about this VoIP cracking thing, which we just loved it because it so clever.

Steve: Yes.

Leo: So he's talking, he's referring to the original research paper. What you're failing to note is that the system as-is has 50 percent accuracy for the words and phrases in its list. You said that. This is not the same as the ability to discern 50 percent of the conversation. Ah, that's a distinction.

Steve: Yes, and a very important distinction.

Leo: Right. That's hype. So he's referring to a figure in the article. I'll put a link in the show notes if you want to see it, wiki.twit.tv; I'm sure you'll have a link in your show notes, Steve, at GRC.com. Figure 11 is titled "Performance on selected phrases." All this setup can do is look for select phrases and words. Even the authors' "evil scenario" means the villain has to create a rainbow table of words. One can't use a dictionary pronunciation to guide because people don't speak right. I know we security folks are pushed to be paranoid in order to balance our society's lack of logic, but I think you've taken this to the hype level, which I'm defining as past what the data supports. Love the show and my SpinRite license. David in Seattle. That's a very good point.

Steve: Absolutely. And David, thank you. You're right. Remember that we referred to this again when somebody was concerned that it was possible then to eavesdrop on his VoIP system that he'd set up for his company a long time ago. And so it's, I mean, and in retrospect it's like, duh, I mean, clearly this isn't, if all you've got is compression rates, and you're basing your discrimination on how much compression different audio phrases get, then what they said was they are able to determine with 50 percent accuracy of the phrases they know, what those phrases are after they've been compressed, based on how much they were compressed. Which is entirely different from saying they can transcribe an arbitrary conversation through VoIP. So I did want everyone to relax a lot. This was interesting technology, very clever, but it has a long way to go before it compromises anybody's encrypted privacy.

Leo: Very good to know. Our last question, Steve, and it's some really cool news. There's an app for that. In case you haven't seen it, says an anonymous listener, your show, Security Now!, has an app on the iTunes store - I didn't know this - called "Security Now Catalog." It came out last Friday. No, I have nothing to do with it. I just figured you'd be amused and flattered to know this. It's in the education section. Actually, Tom Chisholm - I don't know if he wrote this or wrote the app.

Steve: Well, no. He wrote the app.

Leo: He wrote the app.

Steve: And it is, if you - I just Googled iTunes Security Now Catalog, and it brought me to the page. So what it is, is...

Leo: It's a buck ninety-nine, but I don't mind he's making a little money on that. That's great.

Steve: Yep, it's two bucks, and it is every - it automatically updates itself so it'll be current, it'll stay current weekly. And it is apparently direct access to all the podcasts, all 296 podcasts, and transcripts. He's got - it links in transcripts. I don't know if he searches transcripts. That'd be very cool if it, like, had compressed transcripts that it could search. Probably not. But he does have a search feature. And it's both the iPhone and the iPad. I'm constantly receiving requests from people, and I know you are, too, Leo, for, like, could I get the first hundred podcasts? The iTunes feed or the RSS feed only gives me the last few. I want to go back further. Of course I have them all on my site.

Leo: As we do, too.

Steve: As you do.

Leo: You always have access to it. But the RSS feed would get mighty big with 300-odd entries in it.

Steve: Yes. And so this is - it looks, I mean, it's very simple, but it's very clean. It's just it's a listing of all the podcasts, which you can search by name. And presumably you click on it, and it downloads it. So it's just a way to - it's the Security Now Catalog app for iOS.

Leo: Wow.

Steve: So thank you, Tom. And thank you, anonymous listener for letting me know because I don't know, I mean, we would have found out about it sooner or later. But it was just last Friday, so it came right to our attention. Which is cool. And I hope that Tom sells some because that would be great. And maybe he'll make it better.

Leo: And Tom, email me. Because if you want to write an app for every one of our shows, I'd love to help you do that. Wouldn't that be great? Every show should have its own app.

Steve: Yeah.

Leo: Why not?

Steve: Yeah.

Leo: And if you, I mean, of course you can subscribe to any podcast. But I love this idea, that every one of the shows had its own app. If you were a really big fan of that show, like Security Now!, you'd always - I think this is great.

Steve: It's got a nice little icon, which he obviously grabbed off of...

Leo: It's our album art, but that's fine. And it says "catalog" instead of audio or video on the right. Tom, you've got my wholehearted thanks and support. And I would love to talk to you. So please - I presume he listens, or he wouldn't have done this.

Steve: Good point.

Leo: Email leo@twit.tv. And I want to commission you to write one of these for every one of our shows. I imagine, once you've written the first one, it's all the same. I mean, it's very similar naming and so forth. Steve, I have one more question from Leo to you. I've got to send you two emails I received - and I just can't figure out if it's a spoof or not - I received last week, and everybody said, well, you ought to ask Steve, I received an email from the president of one of the world's largest advertising companies, Publicis, last week, saying we would like you to come to the EG8 Summit in Paris - by the way, this is May 25th and 26th or something like that, it's coming - 26/27. It's coming up, like, in a month.

Steve: Wow, yeah.

Leo: As the guest of President Sarkozy because we're putting together a panel of people who are Internet luminaries to guide the G8 summit, which is a summit of the top, the leaders of the top eight nations in the world, as to information and communications technologies, the Internet as a force in economic growth and so forth. And I thought, yeah, right. He said, you'll get an invitation from President Sarkozy soon. Well, I've got the invitation. But it was emailed. And this makes me suspicious. Wouldn't the president mail it to me? I'm a little worried about that. So I want to send you these messages. Maybe you can look at the headers.

Steve: I'd love to look at the headers. I'll tell you what I find, sure.

Leo: It just doesn't feel right to me.

Steve: I mean, clearly this is not a come on that would work for pretty much anybody but you. I mean, it's just not...

Leo: No, and there is going to be this forum. I just feel like maybe somebody's pulling my leg. I don't know.

Steve: Wow.

Leo: The G8...

Steve: Clearly you need to talk to somebody...

Leo: I've got to call them before I buy a ticket.

Steve: ...before you go make your plane reservations, yeah.

Leo: President Sarkozy is the president of the group of eight countries: Canada, France, Germany, Italy, Japan, Russia, the U.K., and the U.S. And they're convening an extraordinary, invitation-only meeting of the - I'm almost embarrassed to say this - the best and the brightest technology leaders from the G8 and the rest of the world. It's an EG8 forum. It'll be in Paris, preceding the G8 Summit in Deauville.

Steve: Wow.

Leo: Actually I guess the G8 Summit's the 26th and 27th, so it's a few days - the 24th and 25th at the Tuilerie. Invitation-only. I think I should go, if it's real.

Steve: Oh, my god, of course you have to go.

Leo: I wouldn't have any choice.

Steve: No.

Leo: But I just - I can't believe it's real. I'll send you the emails. You could tell me if the headers make any sense.

Steve: I'd love to. I'll track them down. I'll see what they say.

Leo: You know why I believe it? As far as I could tell, the header that came from Maurice Levy, the president of Publicis Group, was sent from a Lotus Notes account. And I don't think any hacker uses Lotus Notes.

Steve: Good point. Good point.

Leo: Doesn't seem right. But maybe somebody got these and intercepted them and reformatted them and sent them - I don't know. I guess I should call. Maybe I'll call - I'll call Nic. The President of France, I'll call him. Hey, Nic baby, comment ca va?

Steve: Yeah, he'll probably be going off to a fancy wedding here before long.

Leo: What's happenin'? What's happenin'? Did you really mean to invite to invite me to that party? They don't want me. Steve, so great to talk to you. Thank you so much, as always. If you want to find out more about this show, get the show notes, the transcriptions, the actual audio including 16KB for the bandwidth impaired. We're getting very sensitive to that, I think we're going to have to start doing that for all shows as these new bandwidth caps are implemented. Go to GRC.com. That's a great place to ask your future questions. We'll do this again in a couple of episodes. GRC.com/feedback. And by the way, when you're at GRC, might as well just plunk down, what is it, 80, 90 bucks for SpinRite?

Steve: Yup.

Leo: Well worth it. If you've got a hard drive, you need SpinRite, the world's finest hard drive maintenance and recovery utility, and of course Steve's bread and butter. GRC.com, the Gibson Research Corporation. Steve is on Twitter, @SGgrc.

Steve: Yup, I'm enjoying notifying people on short notice of things that happen. And, boy, it's becoming a really fantastic resource for me. Our listeners know that they can just drop me a short little blurb when something happens to make sure that I know. And it's just great. I accumulate things during the week and then deal with them during our podcast.

Leo: Well, thank you so much, Steve. This is, obviously, if you follow security at all, and you follow Twitter feeds from security experts, this is a must-follow: SGgrc. Steve, always a pleasure. We'll see you next week, our usual time, by the way, Wednesday, 11:00 a.m. Pacific, 2:00 p.m. Eastern, at live.twit.tv.

Steve: Thanks, Leo. Talk to you then.

Leo: Take care.