Listener Feedback #151

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-371.mp3
Quarter size (16 kbps) mp3 audio file URL: http://media.GRC.com/sn/sn-371-lq.mp3

Leo Laporte: This is Security Now! with Steve Gibson, Episode 371, recorded September 26th, 2012: Your questions, Steve's answers, #151.

It's time for Security Now!, the show that protects you and your loved ones and your privacy and all that jazz online with this guy right here, our Explainer in Chief, our security guru, Mr. Steve Gibson of GRC.com. Steve was of course the guy who discovered spyware in the first place, coined the term "spyware," wrote the first antispyware app. He's also the author of SpinRite, the world's best hard drive maintenance utility, so he's an expert on hard drives. Joins us every week. I don't really need to introduce you except there may be some new people in here listening. We welcome you all.

Steve Gibson: I know that we do get new people from time to time.

Leo: Oh, yeah.

Steve: I think probably we have - certainly we have people who are dedicated long-time listeners. But there's also some churn. There are people who get busy and sort of drop off the roles, and new people coming along. There was a question that I selected today about SSL authentication and man-in-the-middle stuff. And I thought, well, we've discussed it, but it keeps coming up. And so that says to me it's an important issue, and it's worth giving it a little bit of time. And I always try, even for the people who believe they know this as well as I do, and very well may, try to come up with some new information, even when I readdress things that we've talked about before. So I think I have that. And, boy, we had a busy week. Some weeks, not so busy. This week, lots of fun stuff to talk about.

Of course, two days after we recorded last week's episode with Mark Russinovich - which we got a great bunch of great feedback about, by the way, everyone really enjoyed having Mark on. So that was neat. Two days after that, Microsoft did release a formal patch for Internet Explorer, which they pushed out through - this is an out-of-cycle patch they pushed out through their Windows Update facility. So we were talking about this bad vulnerability that IE had. And at the time there was only the little Microsoft Fixit deal that would shut that down. And, wait, no, I'm confusing myself. That was different. This one they were recommending the enhanced experience security, that EMET deal, which it turns out didn't really close it down very well. So the only real advice was don't use IE.

So as a consequence of this and the fact that this was being actively exploited in the wild, they pushed out an out-of-cycle patch. It fixed not only that, but four other privately reported vulnerabilities. They wrote that the security update resolves - I'll paraphrase. The security update resolves one publicly disclosed, which is the one we've been talking about, and four privately reported vulnerabilities in Internet Explorer. The most severe vulnerabilities could allow remote code execution if a user views a specially crafted web page using Internet Explorer. An attacker who successfully exploited any of these vulnerabilities could gain the same user rights as the current user, blah blah blah, we've heard that all before.

This update is rated "critical" for IE6, 7, 8, and 9 on Windows clients and "moderate" for IE6, 7, 8, and 9 on Windows servers. And that's because in Windows servers the Internet Explorer runs automatically within a much more constrained environment, so it can do less damage there. And under no circumstances is IE10 affected. Actually, IE10 is not affecting many people's lives at this point.

Leo: It's not out yet.

Steve: Exactly. So that's been taken care of pretty quickly. However, we will in a second be discussing, well, let's jump to it right now. I'll do this a little bit out of sequence. We have another massive Java exploit.

Leo: What? Already?

Steve: Yeah, well, Leo...

Leo: That was last week.

Steve: This is a new week.

Leo: So this is a new one, not one you've referred to before.

Steve: Never referred to this before.

Leo: Geez, Louise.

Steve: This is from Adam Gowdiak of Security Explorations. And this is either their 50th or 51st exploit that they have found.

Leo: Wow, they should have a cake.

Steve: Well, they ought to - Oracle should be paying them. It's like, come on, let's get this fixed. Okay. So this one is really significant because it affects all Java versions since 5, which is the last eight years' worth of Java, which is all of 5, all of 6, and what we have so far of 7. And due to the back reach of this problem, they're estimating, when you update or install Java, it brags about how it's in three billion devices, and then it goes and tells you all the things. It's, like, in your shoes and all clients everywhere. So the estimate is, due to the back reach and the depth of Java's reach, maybe a billion users.

Leo: Well, and people don't update their shoes very often, so the chances are you're running the old Java in there.

Steve: Exactly. And you might get tripped up.

Leo: [Laughing] Nice.

Steve: So Adam wrote that, "The impact of this issue is critical. We were able to successfully exploit it and achieve a complete Java security sandbox bypass in the environments of Java SE 5, 6, and 7." Adam wrote that Security Explorations, his group, successfully pulled off the exploit on a fully patched Windows 7 32-bit computer in Firefox, Chrome, IE, Opera, and Safari. Although testing was limited to Windows 7 32-bit, Gowdiak told Computerworld that the flaw would be exploitable on any machine with Java 5, 6, or 7 enabled.

Leo: Now, presumably on those browsers like Chrome and Safari, it said, "I would like to run Java now. Is that okay?" Right? Did it bypass that?

Steve: Well, if your browser asks you, then yes. Now, this is - note also they were messing with it in Windows 7 32-bit, but also 64-bit Windows, Mac OS X, Linux, and Solaris. So this is also cross-platform. This is a core vulnerability in - actually it's an exploitation of Java's type management, not like...

Leo: It's write once, exploit everywhere, as we said last time.

Steve: Right.

Leo: The beauty of Java.

Steve: So last time they advised Oracle of a problem, Oracle ignored them for four months. And then we had that really bad zero-day Java exploit as a consequence of Oracle sitting on their, whatever they were sitting on for those four months. They've advised Oracle. They've given them technical details. They've given them proof-of-concept code. And so we'll see how long Oracle takes to respond. It's not clear whether what little has been divulged is enough for bad guys to go duplicate it. But we do know that it's in Java's type management. We know that it's been there for eight years. So there's some clues.

We'll see now who's first, the bad guys exploiting it - well, the other thing that can happen, we're seeing evidence, as we talked about last week, of there being inventories of known vulnerabilities in the bad guys' toolkits. And they're keeping these vulnerabilities offline and doling them out as necessary. There's a chance that the bad guys already know about this, but haven't had an opportunity or a need for another Java zero-day vulnerability. The news of this hitting, though, means for them that there's a window. And so that would mean that they would immediately bring out the implementation of this in exploits, knowing that now that Oracle knows about it, there's a time limit on how long they're going to be able to use it.

That's the reality, from everything we've seen of today's cybercrime world, is that all of these web exploits are potentially known, and the evidence is that bad guys have an inventory, and they bring them out as they need them. But learning about this, publicizing it this way, means they would be induced to use it now because they know pretty soon it's going to get shut down. So don't...

Leo: Has that changed, now that they do these exploit kits? In other words, this model that you're talking about is some elite hackers who are saving this stuff, and they have a reserve and a reservoir. But it seems like many of these exploits, especially exploits like this that take place over the web, are just being sold into kits. And so as soon as they're discovered, they're added into a kit which bad guys buy.

Steve: Right. Now, normally what'll happen is, for example, Rapid7 is the group that is managing the Metasploit kit. And so once the nature of the vulnerability is public, then Rapid7 will immediately stick it into, I mean, in a matter of hours, stick it into a new module in Metasploit, and then it's available widely. So that's sort of a different aspect of this. Right now, as far as we know, no one except the Security Explorations guys, well, and Oracle, know the details of what they have found. We have a few little clues, but presumably, well, presumably this is obscure. We don't know yet whether this exists in someone's inventory.

But the Rapid7 Metasploit folks probably, unless they were really motivated to go after this, they'll wait until it appears in a zero-day vulnerability. Once that happens, then it becomes public. So essentially their role is mass availability of something that was previously limited in availability. And that's not good, either. I mean, it's bad because, as you said, Leo, it turns it from something you might have to have some expertise to use into a drop-in toolkit, where it's like...

[Talking simultaneously.

Steve: Exactly. And so the problem here is that eight years of this problem's being present means, for example, it's in all of our DVD players. Now, that's not maybe a problem. But our DVD players are now on our networks. And so it's like, okay. DVD players all have Java in them now. It's one of the things that Oracle brags about. And I notice the sticker on the box and the back panel of mine.

So there are problems with Java itself having been vulnerable this long. So it's not necessarily just drive-by web attacks, but because of the pervasiveness of Java in devices which are not just mainstream smartphones and laptops and desktops, if bad guys had a really focused targeted desire to get in somewhere, very much like we saw with Stuxnet, where that entire enterprise was focused on one specific target, we want to mess up the nuclear enrichment program in Iran. Similarly, there's Java all over the place. A billion is a big number. So this is unfortunate that a problem like this has been found which has such long legs.

Leo: So what, I mean, do we know, is it in cars? I mean, what kinds of devices do you think it's Java 5 or later?

Steve: I'll bet it's in cars. The problem is it has been a very popular implementation language for quite a while. I mean, Java is in my BlackBerry. My BlackBerry uses Java. I mean, like as a core runtime for the BlackBerry. So that's a problem. And dishwashers and microwaves and just - it really is pervasive. It is in devices of all size and makeup. Now, if bad guys can't get to the device by its nature, then that's not such a problem. Or if the device isn't able to do any damage if it became exploited, then that's not a problem.

Leo: Right, yeah. So your car's not online. On the other hand, having it hacked would be a high-risk enterprise.

Steve: Right.

Leo: Oy oy oy.

Steve: I think we probably haven't seen the last of...

Leo: Heard the last, yeah, yeah.

Steve: Yeah, of this one. As it turns out, the guess that I had, which was actually not a big deal because it was pretty clear from the meaning of the acronym, we talked last week about this year's follow-on to last year's BEAST exploit, this one called CRIME, from our buddies who hang out on the beach sipping umbrella drinks. That was CRIME, which was unveiled at the Ekoparty late last week. The acronym stands for Compression Ratio Information-Leak Mass Exploitation. And it works...

Leo: Nice acronym.

Steve: Yeah. It's one of those, as you said, you reverse-engineer the acronym. You say, well, let's call it CRIME.

Leo: Yeah, retronym, yeah.

Steve: We have compression ratio, CR. Ooh, information, CRI. What ends with CRI? Oh, ME. Now, what can that stand for? Oh, mass exploitation.

Leo: Perfect. Easy.

Steve: Yeah. Works. So this essentially creates a serious problem for Internet on-the-fly compression because, well, and so when we talked about this last week, I talked about how there had been mitigation already put in place. Well, what that turned out to be is Chrome and Firefox both immediately shutting down compression. Chrome and Firefox were the only two browsers which were standards-compliant enough, or you might say advanced enough, to be supporting the compression, the SSL/TLS compression that is in the spec and has been, I think since '04. It's been there for a long time, the idea being that there is a specification actually in the SSL specification for how the endpoints can negotiate and perform compression.

So that's at the connection level as opposed to what you'd call the "application level." The application level, the server, the browser can say, independent of the connection, I'm able to decompress using Deflate and Gzip, for example. And then the server says, oh, good. We're going to save bandwidth by compressing those things before we send them to you since you've said you know how to decompress them. So that's at the application level, or at the HTTP protocol level. It's also possible to do compression at a lower level, down at the SSL link level, essentially.

So only Chrome and Firefox ever did that. They no longer do that as a consequence of what these guys found. Also, SPDY used to do that, and SPDY no longer does that. Basically, compression has been, at this level, was immediately backed off of. Now, it's not news that compression can leak information, that is, sort of generically. It's been well understood for many, many years that compression and privacy were at odds with each other for exactly the reasons I explained last week, which is because the amount of compression you get is a direct function of what you're compressing, if the bad guys have any control over what is being compressed, that allows them to probe for what they don't know, what they don't have control over, which they're trying to determine.

So in this attack they arrange to put their own data in front of unknown data and then compress the entire packet. If what they're putting in front is very similar to what they don't know is behind, it compresses highly because the compressor sees that it's already seen something similar. So instead of storing it again, it just points to what it's already seen. Thus you get the compression. So by fiddling around with injecting their own data in front of the unknown, for example, cookie in an HTTP transaction, they're able with surprising efficiency, it takes about six transactions, to decrypt one byte of cookie value. So doesn't take a long time. They're able to crack cookies which are being used for sessions and then hijack a session.

So for the moment the industry needs to rethink this. It's going to be necessary to add some protections in order to put compression back into our links. Compression is something we want. It's tremendously efficient, especially when you're compressing, like,

big web pages where they're just a huge amount of redundancy. You've got English in large blocks of text. You've got all that HTML representation, which is highly redundant. You get a huge gain in compression. But as we've seen, it's also possible, I mean, as we now know, this is something that went from a theoretical problem in information theory, like, oh, compression and privacy are at odds, to, hey, that's true. Here's how you use that fact. And these guys have.

So what's happened is the browsers have all backed off of this in order to protect us so that we listeners, users, don't have to do anything. The browser manufacturers have already taken care of it. This all got shut down before this went public, so we're safe. And the problem is well understood, very clear now. And I would imagine at some point we will come up with some solutions. So, you know, really, really interesting attack.

Leo: Yeah, no kidding.

Steve: Just another one of these where...

Leo: Clever.

Steve: ...it's very clever, and it's difficult to foresee these problems. But once they're made clear, we know how bad they are. A lot of news, I guess grumbling maybe is the right term, for an error message or a warning message or an advisory message that Microsoft's Hotmail began delivering since we last podcasted. People logging into Hotmail received a surprise notice that said: "Microsoft account passwords can contain up to 16 characters." And here's where it gets good. "If you've been using a password that has more than 16 characters, enter only the first 16." Well, okay. Our users know what this means.

Leo: It means they're not hashing; right?

Steve: Well, it means - we don't know for sure. So here's what it could mean.

Leo: I mean, 16 is a lot. So it's not like it's too short.

Steve: That's my feeling, too. 16 well-chosen wacky characters is more than enough. So that really wasn't the issue. It's what does this mean about what Microsoft is doing. So what we know is that you're able to enter only the first 16 and log in, even if the password you had been entering was longer. So passwords may have been stored in plaintext, and now only the first 16 are being checked. And I'd be surprised if that were the case, but it's certainly possible. Remember that Microsoft bought Hotmail, acquired it at some point, after it existed.

Leo: But they rewrote it because it was running on Apache. It was a LAMP stack app.

Steve: True, true.

Leo: And they rewrote it to work with .Net. So they probably...

Steve: Although in rewriting it they, as far as I know, didn't force everyone to revamp their passwords.

Leo: Well, they must have maintained the system, that's right.

Steve: They may have, yeah, left that legacy stuff in place. So maybe they were stored in plaintext, and now only the first 16 are being checked. Or they may have been stored in plaintext, and Microsoft recently decided to switch to hashing. And look at the news, I mean, all of this leakage of plaintext passwords. Imagine that Hotmail had been in plaintext, and they said, oh, we've got to hash these passwords. So what they could have done is decide to switch to hashing, but for some reason - seems arbitrary to me, but Microsoft sometimes is inscrutable in their thinking - maybe they decided to only use the first 16 characters in their hash. So that's a possibility.

Or they were always hashing only the first 16 characters, and now Microsoft is just informing people. That is, they may have always been throwing more than 16 away and just decided, well, we ought to just - I mean, the wording of this, "Microsoft account passwords can contain up to 16 characters," this feels to me like it's broader than just Hotmail, like Microsoft's trying to - because they've got SkyDrive, and they've got all these - they're in general going cloud mode. So there are other ways and things that Microsoft's users will be logging into. So maybe they're trying to unify this, and Hotmail was sort of weird compared to the way they were doing other things, and so they're trying to pull it all together. Anyway...

Leo: It does have two factor; doesn't it? Or maybe not. I have to look. They're moving everybody to Outlook.com instead of Hotmail anyway.

Steve: Right.

Leo: BSD, by the way, not a LAMP stack, my apologies.

Steve: So what we do know is that, given their ability to log people in with only the first 16 characters, it could not be that the entire length of the user's longer passwords were being hashed because they would have a hash for the whole thing, and there's no way then, by virtue of the strength and power of hashing, there's no way for Microsoft to know what the hash would be for only the first 16 characters. So we do know that they were not hashing the user's entire passwords historically. They were either in plaintext, or they were only hashing the first 16 characters. Or they were always only doing that. They may have switched to that, or they may have always been only hashing the first 16 characters, and now just sort of letting people know, don't bother typing any more, you're not getting any better security. I don't know. But people got upset.

Also, there was a little mistake that was discovered in Samsung phones.

Leo: Yeah. The GS3 that I use, as a matter of fact, yeah.

Steve: Yeah. So it was a flaw in Samsung's, what they call their "TouchWiz" software. So, first of all, this is only Samsung phones and not the pure Android phone, so not the Nexus. Only some Galaxy S2 and S3-class phones were susceptible. And in some cases this depends upon which firmware version was running. Samsung has since updated their S3, the Galaxy S3 firmware to fix the problem. But some S2 models may still be at risk. And also apparently the Galaxy Ace and the Galaxy Beam are also affected.

Okay, so what's going on? The vulnerability is the result of the way the native Samsung dialer app handles what's called USSD codes and telephone links. USSD codes are special combinations of characters that can be entered in the keypad to perform certain functions.

Leo: You know, those have always made me nervous.

Steve: Yeah, and it turns out that they just had a bunch that were not documented. It's like, okay. So, and here again, this is just pure obscurity, which is never a good idea, especially when something is this simple. So these were, like, enabling call forwarding, accessing hidden menus on the device...

Leo: Some of these are known. I mean, when you do call forwarding, Google Voice, for instance, will tell you what to, you know, the weird thing to enter. And there's also there's usually a service mode and things like that.

Steve: Yeah. So on Samsung phones, turns out there's a USSD code for factory resetting the phone. Oh, and also presumably another one for nuking the phone's SIM card.

Leo: Oh, wow.

Steve: Since that's been reported to be possible, as well.

Leo: Oh, my. And there's no check. It just does it.

Steve: Nope. It requires no user interaction. It doesn't pop up and say, are you really sure you want to factory reset your phone? Who knows why they did it this way. They just thought, oh, well, no one'll find out about it. And maybe...

Leo: And I'm sure it's long and obscure, but that doesn't mean anything.

Steve: No, no, no. Star pound zero six pound.

Leo: That's it?

Steve: That's it. Don't type that, anybody. Don't enter that into your Samsung phone.

Leo: Oh, my goodness.

Steve: That's all it is. So, okay. The good news is there is a test. Oh, I should mention that what's bizarre is that, on top of all of that, so that's if you are using the native dialer, well, it turns out that for various fancy easy-of-use purposes, it's possible for other Samsung apps to forward those to the dialer. As a consequence, QR codes, NFC events, and URLs dropped into the standard browser will invoke the dialer and can maliciously give it that code and wipe your phone completely, immediately.

Leo: So they could text, send you a text message that would wipe your phone.

Steve: Don't know if text - I'd be surprised. But certainly QR codes and URLs. So the concern is that you can have a malicious website that would tend to be visited by Samsung owners, and they would go there, and their phone would wipe itself.

Leo: Wow.

Steve: So there is a test, for anybody who's worried or wondering: androidcentral.com/ussd-test. And it's a benign test. Again, androidcentral.com/ussd-test.

Leo: Obviously do this on your phone.

Steve: On your phone. Yes, you go there on your phone, with your phone's native browser.

Leo: Oh, okay. Not Chrome. So I'm going to use the stock browser, okay.

Steve: Your phone's stock browser.

Leo: Okay

Steve: Go there. And then click the button. What will be shown if your phone is vulnerable is your own phone's IMEI number, the International Mobile Equipment Identity number. If you see that, it's very likely that your phone is vulnerable.

Leo: Wow.

Steve: But if your dialer just pops up showing you either nothing or that star pound zero six pound, then that's been disabled on your firmware, and you're probably safe.

Leo: And Samsung did push an over-the-air update. Many of you got that today.

Steve: Right.

Leo: What is that? Androidcentral.com/...

Steve: Ussd-test.

Leo: Why couldn't they just make it shorter?

Steve: I know.

Leo: All right. I'm going to try it on mine. Now, I'm running - the good news is I'm running CyanogenMod. I'm running a mod on here. So I'm sure there are other vulnerabilities, but not that one.

Steve: Now, okay. So at this point, if our users or our listeners have, for example, Galaxy S2 phones, for which updated firmware is not yet available, if you switch to a third-party dialer, such as apparently there's one called Dialer One, Dialer One isn't susceptible to this. So moving away from the Samsung dialer, it's Samsung's dialer that knows about these special codes. If you switch to a third-party dialer as your default dialer for the phone, then you'll also be safe. And you can of course verify that using this Androidcentral.com test.

Leo: Wow.

Steve: Yeah.

Leo: That's amazing. So when you get to that page, it has a test, and you have to click that, click here to begin.

Steve: Right.

Leo: And then it will run the thing. And what happened with me is the dialer popped

up with that string, that star pound.

Steve: Yup.

Leo: So, and then I dialed it, but it said nothing happened, so...

Steve: Good.

Leo: It's okay. What you don't want is for it to display your IMEI.

Steve: Correct. If you see that, that's not good news.

Leo: And that's a long, much longer number.

Steve: Oh, yeah, it's like a huge serial number kind of thing.

Leo: Right, right. Wow.

Steve: So as you mentioned earlier, Leo, the IEEE.org website turned out to have a rather substantial username and password leak. A TA, teaching assistant at the University of Copenhagen, Radu Dragusin, reported that he found 100,000 usernames and passwords stored in plaintext that had been sitting for a month on a publicly accessible FTP server belonging to the Institute of Electrical and Electronics Engineers, the IEEE.

Leo: Amazing.

Steve: But after finding that, he poked around the Internet some more and found 15 web pages' worth of 14-month-old IEEE log folders on a Russian website. Which tells us that the IEEE files may have been publicly accessible for more than a year. So another instance of, whoops, you know, information leakage from a site that we would hope would be more secure than that.

Leo: And by the way, I believe Radu is a listener because he was in our chatroom earlier.

Steve: Oh, cool.

Leo: Yeah. So there you go. Nice job.

Steve: Yes, very. Finally, Bruce Schneier, our good friend/cryptographer guru in the industry who's designed a bunch of great ciphers, had an interesting blog post. He mentioned that the six-year-running competition to select the successor to the SHA-2, or SHA-2, as it's sometimes pronounced, family of secure hash algorithms is coming to an end, and that the NIST, the standards-setting body, is nearing, very near to choosing the final hash for what will be called SHA-3. So this is similar to what we saw happening years ago where Rijndael, the Rijndael cipher, won the next-generation cipher competition to be assigned the designation of AES for that standard, the Advanced Encryption Standard.

So in this case they started off six years ago with 64 contenders for this title, now winnowed down to just five, one of which is Bruce's team's own - they call it "Skein," S-k-e-i-n, which is based on Bruce's ThreeFish large-block cipher. We know that Bruce did Blowfish, which is still strong and useful. Then he did another generation called TwoFish. And now he has ThreeFish. And this sounds like a Dr. Seuss book.

Leo: By the way...

Steve: OneFish, TwoFish, ThreeFish.

Leo: ...Radu came back in the chatroom, and he said, "And not only that, but I used Squarespace for the site that I demonstrated this on." So not only is he a listener, he supports our sponsors. Thank you, Radu. Isn't that awesome. That's great. I love that. I'm sorry. Back to the OneFish TwoFish ThreeFish BlueFish.

Steve: ThreeFish and BlueFish, yeah. Blowfish. So what happened was, okay, I should explain that Bruce thinks none of this is necessary, even if his own hash wins. He explains in his posting that six years ago this competition was initiated for the successor to the SHA-2 family because it was just assumed that, with computers getting faster and crypto getting better, and everybody being better at understanding vulnerabilities and cryptanalysis, the assumption was made that the SHA-2 family would not withstand the test of time.

And he says, but six years later, SHA-512, which of course is a hash that gives us a 512-bit digest, more bits being better, for example, the SHA-256 is there, and it's fine, but SHA-512, he says, is holding up extremely well. That is, we don't need anything more. And so he points out that none of the five currently surviving contenders for the SHA-3 crown is enough significantly superior to really justify switching. He says some are faster, but not orders of magnitude faster. Some are smaller, but not orders of magnitude smaller.

And he said that, when SHA-3 is announced, he's going to recommend that, unless the improvements are critical to users' applications, that is, for example, if speed really matters, or if size really matters, whatever this next generation is will probably be faster and smaller. So there is that aspect of evolution in our ability to design a secure hash function. That we have seen over the course of since SHA-2 family was designed. He said, but unless those improvements are critical, he's going to recommend that people stick with the tried-and-true SHA-512, at least for the time being, at least until there's some reason to move. So I thought that was interesting, that what we've got now is holding up.

Now, essentially we'll have a - assuming that NIST does pick some, or one, we'll have a successor sort of waiting in the wings that is ready to be deployed. And you might just go ahead and use it for new things. But absolutely no reason, as far as we know at this point, to stop using the existing SHA-2 larger digest hashes because they're, as far as anyone knows, completely strong and solid.

Leo: How rarely do you hear that?

Steve: I know.

Leo: Usually it's like, oh, they're broke. No, these work.

Steve: Yeah, I know.

Leo: They're not broken. They work.

Steve: I got a note, a tweet from a listener of ours, Thomas Fors in Chicago, who brought to my attention Adobe's release of a very nice-looking new font that might be of interest to any of our coding listeners. It's called Source Code Pro. And if you just Google the phrase "Source Code Pro," the first link is to their announcement. And there's a number of links there. And it's very attractive. They did a Source Sans, which is a very nice sans serif typeface. But then the Source Code Pro is a monospace typeface, meaning that every character is the same width, so that they all line up nicely. And they paid special attention to disambiguating - that's one of my favorite words, I don't get to use it often - disambiguating lookalike characters. So, for example, the numeric zero has a dot in its center to help make it very clear that it's not a capital "O" alphabetic. So I just wanted to let people know...

Leo: Yeah, I like that. I tell you, because I have passwords that have zeroes, capital O's and zeroes in them. And I can never tell the difference. I like a font that will let me know that.

Steve: Yup. And like the number "1" and the lowercase "l," those get confused. And, yeah. So this is very nice. They did a great job of making them extremely clear.

Leo: Good.

Steve: And then finally, we've talked about this before, or actually relative to Nevada, but just yesterday California joined Nevada in officially allowing "autonomous self-driving automobiles."

Leo: Yeah?

Steve: On the roads. It's like, okay. I guess if you look at the car passing you, and there's no one there, it's not - you're not seeing a ghost. That's just Google driving by.

Leo: Did you see the announcement where Jerry Brown's talking about it, and of course Sergey Brin's there in his Google glasses. I guess he wears them all the time now. And it's just strange as heck. I mean, it feels like it should have been Arnold Schwarzenegger. Then it would have been like, okay, "The Terminator," I get it.

Steve: Right.

Leo: But it was a weird picture. I'll see if I can find that.

Steve: So the news is that Arizona, Hawaii, Florida, and Oklahoma for some reason are also currently considering similar legislation. I am curious about why Florida and Oklahoma and Hawaii and Arizona. But, you know, I guess..

Leo: Well, they're going for every state eventually. That's the idea.

Steve: Yeah, yeah. And I've been meaning to mention that I'm still on my stair climber reading "Kill Decision" by Daniel Suarez. And, oh, it is just terrific.

Leo: He's great.

Steve: So when you get through with "Zero Day" and "Trojan Horse," Mark Russinovich's books, if you haven't picked up "Kill Decision," I'm just really enjoying it. It's great writing.

Leo: Very nice, Steve Gibson. I just wanted to pass along before you - I know we probably want to do a SpinRite and everything. But Radu, as I mentioned, the guy who discovered the IEEE flaw, is in our chatroom. And he just passed along, he said, here's the tweet, this is the site that he announced this on. And he said to me and to Squarespace, "I want to say I'm a happy customer." Slashdot could not bring this IEEE log down that he created. And look at all the bandwidth, all of a sudden it's 74,000 page views at the peak, 60,000 - yeah, yeah.

Steve: So that's cool.

Leo: That's really great. Radu, it's nice to have you as a Security Now! listener. They're doing big things. Can I also mention Teespring real quickly? I just want to - there's only a few days, there's like two weeks left on this unique T-shirt - and we will send you one, Steve, by the way, because I think it's in all the hosts'...

Steve: What is it?

Leo: This is just something a little different that we're doing. We have a T-shirt store and everything. But it turns out that there is - if you do one-of-a-kind designs that are for a limited time, people really love that. So we are going to start doing this. This was a user-submitted design. And it's got all the - you can't tell, but all the names of the shows make up the TWiT logo. That's on the back and the front, it just subtly says TWiT.tv. And we're selling this, we're going to raise quite a bit of money, we're trying to raise enough money to buy a new streaming box, which is about a $20,000 streaming box. And so far we've sold 1,054 shirts. So we're a good way along.

There are a few, let me see, it says here, 16 days left. So a little more than two weeks left. If you want the shirt, Teespring.com/twit. And we've reached our goal, so we know we'll be printing them. You get to choose between two very high-quality 100% cotton shirts, American Apparel and Fruit of the Loom. So you get to choose. And we have all sizes. So this is what we're - yeah, I think you can buy one in Switzerland. But this - I think international, as well. This is something we're trying a little different. This is a different manufacturer. They do very high-quality silkscreen shirts. Little more pricey, but I thought I'd pass that along. Teespring.com/twit.

And I think we would love to do a Security Now! shirt at some point. So if you have a design that you'd like to submit for a Security Now! shirt, email Glenn with two N's at TWiT.tv with your design. And we picked this one from a bunch of user submissions. So it's kind of fun.

Steve: Yeah, that'd be cool, Leo.

Leo: Yeah, Security Now!, we should have a Security Now! shirt. Maybe a TNO shirt. Wouldn't that be fun?

Steve: Ooh, ooh, that's perfect.

Leo: Wouldn't that be good?

Steve: That's perfect. Yes.

Leo: Actually we don't even need a design. In simple black letters, Trust No One.

Steve: That's it.

Leo: I love it. Anyway, thank you for letting me interrupt. Go right ahead.

Steve: No, actually the first question is a...

Leo: Is a SpinRite?

Steve: ...is from a listener about SpinRite. So I figured I would kick off our Q&A with that.

Leo: Simplify things. Hey, I downloaded all those source fonts already from Adobe. I have a font folder that I keep on Dropbox. So whenever I set up a new computer I have the fonts that I like. And among them I have a number of programmer fonts - Inconsolata, Droid Sans, Monaco. But this one looks really nice, looks better than...

Steve: Consolas is a very nice one.

Leo: Consolas is - Inconsolata is a free version of Consolas, yeah.

Steve: Ah.

Leo: All right. Starting with our questions. And our first question from Dustin B. in Seattle, Washington.

Steve: Wait. Which Q&A are you reading?

Leo: Oh, wait a minute. Number 360. Holy cow. This is from July. That would explain why the notes didn't...

Steve: Yeah, because I cannot pronounce the name. I was going to saddle you with the pronunciation of our listener in Norway.

Leo: Oh, you know what? I've got them sorted in ascending, not descending order. Let's go. Let's do today's questions. How 'bout that, Leo? This is from - oh, you're right.

Steve: Yeah.

Leo: Okey-dokey. Okey-dokily-dokey. Maybe somebody in the chatroom...

Steve: I'm not reading that.

Leo: ...can help me with this one.

Steve: Sounds like a great guy.

Leo: Irvelive Sturevirpe in Stian Skarsb Solheim, Norway, posts a terrific question - I hope, Odd Inge, I hope I did that justice. Odd Inge's our regular Norwegian listener - about SpinRite's Level 1 operation: I am currently running SpinRite on a SATA drive, connected directly to the motherboard, that has been drastically slowing down lately. I started a Level 1 scan, and after a couple of analyzed sectors the read speed went down for about 10 minutes. When it picked up again, SpinRite had marked the sector as recovered - green "R." Now, this is Level 1. So I was wondering, does this mean the sector's successfully been recovered, or that a problem was found and can and will be fixed by running Level 2+ at a later time? If it has indeed been recovered, how did it do it, since Level 1's not supposed to write any data to the disk? Long-time SN listener, utmost trust in SpinRite. Just a little bit confused, says Irvelive Sturevirpe in Stian Skarsbo Solheim, Norway.

Steve: So I ran across this when I was going through the mailbag for the Q&A. And I thought, well, I'll just answer that question as my mention of SpinRite for the week. What happens is a Level 1 scan we've talked about before because it would be good, for example, with SSDs, for recovering SSDs where you do not want to be writing necessarily to SSDs. It also is the quickest way of running SpinRite on a drive where there's a problem. Essentially, SpinRite runs forward at full speed, just simply reading every single sector from the drive until and if it has a problem.

Now, it's possible that, when SpinRite asks the drive to read a sector, that the sector, first of all, could read perfectly, which would result in no change. It's also possible that the drive could, when asked to read a sector, see that enough correction, that is, enough error correction is necessary that it's outside of the drive's safety margin or comfort factor. So the drive would, on the fly, recover the data, that is, apply error correction, then move that sector somewhere safe and lock that spot on the drive so that it cannot be used, and return the data. At which case, again, SpinRite would show no problems. It helped the drive to do that, that is, running SpinRite on the drive was beneficial. But you still wouldn't see anything.

Where you see something like this green "Recovered" is where SpinRite asks the drive, or encounters in its reading, a sector which will not read, that is, the drive attempts to read the data and says I can't. It is unable to, even with error correction, to successfully read the data that was originally recorded there. Well, so it returns a "This sector won't read," a sector error. Well, other software gives up. SpinRite's job is not to give up. So what it does is it, with caching, with the drive's own internal caching disabled so that that doesn't get in the way, it reads a randomly chosen sector on the drive, which pushes the head over to some other sector. Then it returns to the sector that it is trying to read.

So what that does is it moves the head a random distance and direction from the target sector and then brings the head back. So the chances are that the head - because we don't have zero friction, we have low friction, and we have friction working together with the servoing going on. But the chances are the head will end up in a slightly different position. Just a little bit different. Or the sector we're coming from will be rotationally in a different place. So we end up getting onto the target track at a different location, so things have settled differently.

The idea is, though, that we keep asking for this sector over and over and over, giving the drive essentially every possible opportunity to read it. And then the idea is we just need it one more time. The drive has said, can't fix it. We say, are you sure? Are you

really sure? Are you really, really sure? And not just redundantly asking, but also asking in slightly different ways each time because we're arriving at that sector in a different way each time. Finally, and this is what - I can't pronounce his name, Irvelive.

Leo: Irvelive. I'm just making it up.

Steve: What he saw and what so many of our users find is that, yes, SpinRite, by being really patient and persistent, gets the drive to read the sector just one last time. The drive says, oh, my god, I've got it. I was able to correct it. It sees that it was probably at the limit of its ability to perform error correction to recover the sector. But that's fine. That's all it needed was one last success. Then, on that read, it does what it did before. It marks that spot as bad, don't use that anymore, grabs a spare sector out of its spares pool, puts the data there, and reports to SpinRite, I got the data. Here it is.

So what SpinRite sees is that it got sector error, sector error, sector error, and sector error many, many times, and then finally a successful read. So the data got recovered. It puts a green "R" there and moves on. So we are doing a read-only scan of the drive at Level 1, but still able to perform data recovery and repair when we hit spots that need it.

Leo: Interesting.

Steve: Yeah, it's cool.

Leo: Thank you, Irvelive. Cal in the U.K. has our next question. How do you keep up - I get this question a lot - with all the security news? Steve, I was wondering how you keep up to date with security news, if you could recommend some good sites? I try to keep up with tech news, and only find out about the security stuff if tech blogs cover it or from your show. I feel there must be some good security news sites out there. I'd like to know what you read. And I wish you'd make an OPML or something of this. Be very handy.

Steve: My approach has evolved over time really as a consequence of my increasing use of Twitter. I used to be on my own, essentially, operating without the benefit of the incredible dragnet of listeners that we have who make sure that I know what's going on. And so my strategy really has evolved. My main go-to, it's not a site, resource is the SANS Security Institute. It's possible to subscribe to their mailing list. And they send out a couple times a week various types of news. And they do a compendium, very nicely organized, of what's happening. So they're very active in tracking what's going on. And so I was relying on them almost exclusively. I mean, I would sort of - we have Brian Krebs, and we have other security columns and things. But there wasn't anything really very organized for me except the SANS Security Institute was just - it was my crucial resource.

And now what's happened is, thanks to SGgrc on Twitter, we've got all of these listeners who themselves have all these resources that they're checking. I mean, I see Slashdot. I'll get duplicates often, and I sort of smile because I know then, like, where the source of that was, and our listeners are keeping tracking of that. But I get the benefit of the concentration of all of our listeners seeing something that they know would be interesting to me and to other listeners of the podcast, who just make a mention, @SGgrc, and I

read my feed and then go and pursue that. So...

Leo: And then we just all listen to you, and it makes it very easy.

Steve: Everybody listens here, yes.

Leo: I was just looking because, you know, I have a fairly long list of things I peruse on my Google Reader list. And I was just looking at what I read. And I do, SANS is great, and also Krebs on Security, we love him.

Steve: Yup.

Leo: He left the Washington Post, but he has his own blog now, and it's very good. Schneier on Security, you were just talking about Bruce.

Steve: Yes.

Leo: And then there's a SecurityFocus site that I also - from years gone by, used to follow SecurityFocus. So those four are pretty good sources of information for me. But Security Now! is the best. If you just listen to this show...

Steve: Well, we do a lot of filtering. There's stuff that it's like, eh, doesn't quite make the cut. And I try to make those decisions and make them correctly.

Leo: Very nice. Always good to know. Question 4, Mr. G., question 4 is from Andrew Stevenson in Dorset. And he very kindly provides the pronunciation, Dorset, U.K. Software firewalls? What's the point? A friend said to me the other day, why do you run a software firewall when you already have a "rooter"? That's how they say it in Dorset. That got me thinking, and I wanted to hear what you thought about having a software firewall installed as well as having a router/rooter, which contains NAT and firewall technologies. I personally have a desktop machine that never leaves the protection of my home network. I have a feeling that having a software firewall is a good thing in terms of security, as relying on a single form of defense is never a good idea. I also feel the fine-tuning of a software firewall and IDS, Intrusion Detection System, also makes me more secure. Also that the software firewall is protecting me from other nodes on the local network - aha - rather than just incoming Internet traffic.

Steve: Exactly. My feeling is that our software firewalls are unintrusive enough that they're just not a problem. We have interfaces, like the operating system has interfaces that allows it to ask for inbound - to deliberately ask to allow inbound traffic when and where it wants; and that the rest of the time, having the protection of dropping packets which are not expected and are unwanted is a good thing. I mean, really a firewall sounds like a big deal. All it is is something that looks at some characteristics of data arriving on the wire and decides whether to pass it on upward into the computer or just

say, eh, don't think we need that. And so it's sort of impressive-sounding. But in implementation it's not that big a deal.

So I absolutely feel that, since it's not something that requires constant maintenance and tending, it's just not in your way any longer. I mean, I absolutely like the idea of operating behind a NAT router. That kind of border protection makes lots of sense. And Leo, you said "Aha" when you read his comment about protecting us from other machines inside our protected perimeter, which is a very good point. Many of the state-of-the-art malware tries to look for local machines that it's able to infect on the LAN. So having our machines keeping their defenses up, essentially individual little islands which selectively allow data in, I think that just - that it makes sense to have a layered security model where local individual firewalls form another layer of protection.

Leo: Is it enough to have just the Windows firewall? Or do you want to go out and - well, you were the guy who discovered and really promoted ZoneAlarm way back when.

Steve: Well, yeah. I was promoting firewalls before they were in the operating system. And I recognized as Microsoft began picking that technology up themselves that third-party firewalls were endangered, just as third-party antivirus is now endangered because ultimately Microsoft is going to move those technologies into the OS. So that has happened. And of course Mac has a firewall as part of its technology, as does now Windows. So, yeah, I just don't think that firewalls represent enough of a problem that there's any reason not to have them. They're there. Just leave them on.

Leo: And use the operating system built-in. That's sufficient.

Steve: Yup, I really think so.

Leo: All right. Here's a complicated one from Tom Ribbens in Belgium. And just as a setup, last time we talked about the fact that the British, like, retirement system or something has an entire /8 block of addresses, which they apparently don't use publicly. Now...

Steve: More than one 256th of the entire Internet's address space.

Leo: And of course, as you all know, with the current system, the IPv4 system, we just are - we've run out, in fact. We are out of addresses. So we'll have to move to IPv6. But in the meantime, a significant number, millions of addresses, are being kind of, well, it seems, misused by the British trust: Steve, I thought your discussion last week about the 51.0.0.0/8 - this address space we talked about - was completely off. You said, "Well, all they have to do is change the 51 to 10," which is of course an internal designation.

This is not as simple as you might think. This would take weeks of planning and preparation and will cause issues along the way. And what for? When the IANA still had blocks of IPs to give to the RIRs, I believe they were crunching through them at

the rate of two /8s a month. That would mean, even if we got that /8 back, it would only move the problem away for a couple of weeks. Seeing that there is no way any sizeable organization could renumber their whole network in two weeks, this is not a viable trade off. Even if we would find 10 such companies who could give back /8 blocks back - and I think there are probably that many - it would still only help for another half year.

You know just as well as I do that the real solution is IPv6, and that adoption will only happen when everybody is forced to adapt. It might cause a little mayhem when it really will be absolutely necessary, but delaying it another year is not going to help a thing because we'll hardly be better prepared, as there's almost no incentive currently to do so. Tom Ribbens. I think he's right.

Steve: Well, yes. I guess the point was that there has historically been a huge amount of waste.

Leo: Right.

Steve: Because we thought, oh, 4.3 billion IPs, we'll never use all of those. And so, early on, huge blocks were being handed out very easily. So the way I view this is sort of a - is a struggle with tension between competing interests and the need to implement IPv6. The problem is people who already have allocations of IPv4 IPs, they want to keep them. They've had them for a long time. They figure that they're entitled to do so. And they probably have a good point. They could make that case convincingly.

At the same time, we need to move to IPv6, but it is a pain. I mean, it requires the replacement and upgrading in some cases of entire networks and switches and routers within a company. And you could also argue that any company that already has IPv4 is disinclined to move away from four over to six. IPv4 addressing will never go away. I mean, probably never. It was first - it will continue to be supported. New allocations at some point will have to be IPv6. But at the same time, looking at huge blocks of unused IPv4 does create some tension because it would be easier to reuse that than it would be to make the move to IPv6.

So first of all, I mean, the one area where I disagree with Tom is the rate of consumption. It is no longer the case that /8s are being consumed at the rate of several a month. Remember a /8, as I mentioned before, is a huge chunk of the Internet. It's about a 200th of the Internet, of the entire address space. That's massive. So today, now that we know IPv4 IPs are so scarce, they are being managed far more carefully than they were in the past.

So I just sort of see this as a set of competing pressures. There is pressure to better use existing IPv4 space. There is pressure to move to IPv6. And we are running out of IPv4 space over time. Yet people who have large allocations of IPv4 that they are not using, there's some argument to be made for freeing some of that up to release some of the pressure. But, yes, ultimately, new people are going to have to be using IPv6. We'll get to a point where there will be no more IPv4 space.

Leo: Yikes.

Steve: Yeah.

Leo: Yeah, I mean, and it seems wasteful. I guess you could go back and forth on this. And there's a large camp of people that say, well, look, we've just got to have the pain, or it's not going to happen. And I'm not sure I disagree with that. Because, I don't know, I think what's going to happen is you're going to have ISP NAT. We're never going to go to v6 at home. It's going to be the ISPs who do it all. And we're actually...

Steve: Yeah...

Leo: Go ahead.

Steve: Well, imagine that a company said to another, a squatter, we'll pay you X amount of money for a chunk of your IPv4 space that you're not using because it's easier for us and more cost-effective for us to do that than it is to move our infrastructure to IPv6. So that may happen. I did see some dollar signs associated with the value of IPv4 space. And it was stunning.

Leo: Well, yeah. Especially as it gets more valuable as there's less of it.

Steve: Yeah, I just think that what we'll see is future, like, existing companies that have been around that have IPv4 probably get to keep it. Now, if they're offered chunks of money, where it makes sense for them to move, well, maybe they'll choose to give up some. But new allocations will probably, by virtue of the fact that there won't be a choice, will be in IPv6.

Leo: Question 6 is Chad Jacobson, Burlington, Vermont. He wonders about LastPass. He's quoting the transcription from Elaine of Episode 369. Quote: "I brought up IMDB, the Internet Movie Database, which I poke around from time to time. And this was an app on my iPad. Same experience under Firefox, for example, in Windows. And it prompted me to log in with IMDB, Amazon, Facebook, or Google. And I thought, Amazon? What? And sure enough, if I click on Amazon, I jump over to Amazon, LastPass sees that I'm being prompted to log into Amazon, it does that for me, and I'm back to IMDB, having logged - and it knows my name." You were talking about OATH.

Steve: Right.

Leo: The ability to use a trusted third-party for logins.

Steve: Actually, OAUTH.

Leo: OAUTH. Right. This isn't OATH, this is OAUTH. As an iPad owner and LastPass Premium customer myself, my first question is were you browsing in the LastPass Tab when this automatic login took place, since to the best of my knowledge LastPass does not, unfortunately, integrate directly into Safari or any other browser on either the iPhone or the iPad. I frequently forget to start my browsing sessions within LastPass on my iPad, and should I need to log in somewhere I am forced to "move" my session over to the LastPass app or copy over my very forgettable LastPass-generated passwords into my browser.

You may have covered my second question in a past episode, but why can't LastPass be integrated directly into iPhone and iPad browsers? Would Apple need to give away the keys to its iOS kingdom to make this happen? Does the nature of iOS make it functionally impossible? Or is it the functionality of LastPass that makes that level of integration unfeasible?

My thanks to you and Leo for what is without a doubt the finest technology podcast, sorry netcast, available. I will continue to consume every episode as long as you are willing to produce them. Chad Jacobson, also a very satisfied SpinRite owner.

Steve: So...

Leo: Question 1 first, the iOS question, yeah.

Steve: Yeah, a couple things. I tried to recreate that on my iPad and didn't see what I remember seeing. So I may have misspoken, and this was the behavior I had in Windows with Firefox.

Leo: Well, and I'll tell you what might have happened. If you pick Facebook or Twitter, the iPad preserves, iOS preserves your Facebook and Twitter - Twitter for a while, and Facebook since iOS 6, credentials and does the login for you. So Amazon would not have done that.

Steve: Right.

Leo: But had you said, oh, let me use the Facebook login, the iOS will, my experience has been, will do that for you automatically. Once you verify Facebook and Twitter connections on the iPad or iPhone, it will log you in to other sites through those places.

Steve: Okay.

Leo: So that's maybe what happened.

Steve: Yeah. For what it's worth, I feel like Chad and I are on exactly the same page, and probably many of our listeners are. And that is this annoyance that iOS and iOS's

Safari mini browser is unable to integrate with LastPass. This, of course, is by Apple's design.

Leo: You bet.

Steve: They do not have a Safari plugin ecosystem and don't want one.

Leo: And you can see why they don't want one.

Steve: Yes.

Leo: It's a security issue, absolutely.

Steve: Yes, absolutely. And so I similarly use LastPass Tab when I'm needing to log in somewhere, and I have no idea what my password is on many obscure websites. LastPass knows what it is, and so I have to switch over to that browser and happily use it to log me in.

Leo: It's its own browser, it's just it's using the WebKit, it's using the...

Steve: It's using the same technology.

Leo: ...same stuff.

Steve: Yup. It's just...

Leo: I've got, you know, I have what - if you're a Pro user, LastPass Tab is a future. And I keep forgetting to use that. But I probably should make that my default browser almost.

Steve: It's really nice. I mean, they did a nice job with it. I'm really happy with it.

Leo: It's basically the same as Safari. The only issue is, of course, iOS does not allow you to change default browsers, either.

Steve: Exactly.

Leo: You have to remember to launch it each time.

Steve: Right.

Leo: And his second question - oh, yeah, you answered it, which is why doesn't Apple allow that. Well, that's why.

Steve: Yup.

Leo: Tony Wall in Port Dover, Ontario wants to know about Tomato: Steve, I bought a new router with USB ports in the hope of streaming media from my hard drive to an Xbox. Well, even with tech support we couldn't get the Xbox to read the hard drive. So after searching around I came across the open source Tomato firmware. After flashing the router, everything worked great. It's a nice upgrade to the OEM firmware. My question is, as it's open source, how secure do you think it is? Thanks for a great show. You and Leo do a wonderful job. P.S.: Love the Honor Harrington series and Mark Russinovich books, so keep those suggestions coming. I would add to Tomato DD-WRT, which is another open source router firmware that many routers support.

Steve: Yup, there's DD-WRT, which actually has wider compatibility with router hardware than any other of the third-party firmwares. And it's very feature rich. Tomato firmware is not quite as feature rich as DD-WRT, but has a very friendly user interface. There's OpenWrt.

Leo: That's the one. DD-WRT's older, yeah. Open is the new one, yeah.

Steve: Right. And it's meant to be an open platform for add-ons. It does not have a native GUI itself, but X-Wrt adds one to it. Then there's FreeWrt, which is a fork of OpenWrt that's more sort of aimed at developer experimentation, but not so friendly. It's command line configuration only. If you wanted to set up a public controlled hotspot, there's one called Chillifire, which turns a router into a for-pay or free public access hotspot with the kinds of controls that you might want. And then there's finally one called Gargoyle, which does not have lots of features, but it does offer lots of bandwidth management, like band management quotas and network access rules.

So there's a bunch of different firmwares available. It sounded a little bit like Tony hadn't encountered this before and wanted some assurance that, for example, Tomato, which is oddly named, is a good one, and it absolutely is. You and I, Leo...

Leo: Love it.

Steve: ...have heard about it for years and know of it and recommend it. So I would say you can absolutely use that, and really any of those, with confidence.

Leo: Colleen put Tomato on a lot of our old Linksys routers, and we do a Know How episode on how to flash your router firmware. I think we used, I can't remember, I

thought we used OpenWrt. Anyway, Know How 3 has that, TWiT.tv/kh, and Episode 3 will tell you how to do it.

Michael in Europe raises a very good point about a fundamental OAUTH weakness and how someone might steal an unwitting person's logon credentials: Steve, while I'm sure OAUTH is a great solution for avid Security Now! listeners, I'm worried about the millions of less tech-savvy computer users like my mom, who come across OAUTH and get used to it. Wouldn't it be easy to put up a number of malicious websites and online shops that leave the impressions of forwarding users for authentication purposes to a faked Facebook, Amazon or Google site, then just grab their logon credentials?

If the faked authentication site looks real, I'm sure many less security-aware users wouldn't even recognize that the fake OAUTH page is sitting on some domain other than Facebook, Amazon, or Google and readily fill in their user name and password. In the case of Amazon, the password thief could be out doing his shopping in a matter of minutes. Of course, one could try to educate people to pay attention to the domain that they're forwarded to when using OAUTH, but that doesn't seem to be a working solution for people that are already challenged with the many do's and don'ts of using a computer.

LastPass could be a solution, as it probably wouldn't readily fill in your login credentials on a faked OAUTH Facebook password or Amazon or Google site, but that'd require use of LastPass in the first place, which one can't really expect from millions of Internet users that might be easily tricked with this scam. What are your thoughts? Keep up the great work. Michael.

Steve: [Sighing] That is really a good point. Consider what the - the beauty of OAUTH, exactly as we were just talking about, for example, when I had the experience of logging in to IMDB, is it offers you a menu. And we've always, you know, we're seeing increasingly often "Log in with your Facebook ID." So more and more sites are doing that. People are saying, oh, yeah, that's what I do. I log into sites with my Facebook ID. And so they click on that. Well, and you bounce over to the Facebook login and into your credentials there. And then you submit that, and you bounce back.

Michael's point is that the taking you to that Facebook site is under control of the site you're logging - the primary site you're logging into. So what if it takes you instead to Faceback.com, which is a domain they own, and present you with the Facebook.com lookalike login page and acquire your Facebook credentials, just like that? So here, I mean, this is a classic instance of where ease of use is a really great feature. But it's abused because, I mean, the very ease of use gets us accustomed to it, and we stop really paying attention to see where we are.

And I think this is, I mean, mark my words, this is going to happen. We will see somebody, once OAUTH becomes popular, people are going to create fraudulent bounce sites that you get taken to in order to steal those popular credentials. It's foreseeable.

Leo: They're already doing it.

Steve: And I don't see any way around it.

Leo: It's easy enough to create a fake Facebook site and get people to go there by clicking a link in an email or a text message or whatever. I mean, that's...

Steve: Well, but remember, here they control where you go. So you say, yes, I want to log in with Facebook, and you're blinked over to a site, assuming that it's Facebook, that looks like Facebook. How many people are going to carefully look at the URL to make sure that that's in fact where they are?

Leo: Right.

Steve: It's going to get abused.

Leo: Jeff Horning in Indianapolis suggests maybe it's time to revisit the question of periodic password changes: Guys, thanks for all you do to make our cyber world more secure. I think, with some of the topics you've covered recently, perhaps it's time to revisit the periodic password change option. Here's my Top Five list of occasional change rationales:

No. 5, your phone and tablet display it as you type. Most people don't change this setting. No. 4, when email accounts are hijacked, the hijacker does not have to make his presence known. He's watching your online accounts, contacts, et cetera. He assumes you are using this password elsewhere. Only changing it can kick him out. No. 3, in corporate settings it's very easy to have your password seen when you enter it several times a day. No. 2, hopefully you're listening to Steve Gibson and will make a better password now than you did years ago. No. 1 reason for changing your passwords from time to time: Mat Honan. I know everyone railed against this last time it came up, but I think most of the argument against it boiled down to convenience. Keep up the good work. Thanks for having Mark Russinovich on. Love his books. Jeff Hornung.

Steve: So this was interesting. I'm still, I guess, somewhat dubious about the need to change a really good password just because a month or two of use, or maybe a year, has gone by. But Jeff's point is that there is some level of leakage or potential leakage, and that changing those passwords periodically makes some sense. And of course we now have management technology like LastPass that allows a password change to immediately propagate and take hold on all of our devices. So the impact on us is much lower than it would have been before. So I think that's sort of worth considering, the idea that, well, we're not having to manage our passwords to the degree that we did. And there is some potential leakage of even a really good password. So, yeah, I mean, I guess I'm still not hugely moved. But I can certainly see Jeff's point.

Leo: We'll leave it as up to you, listening at home, as to what you wish to do. But I think what we were talking or railing against was these kind of mandatory, you must change your password every three months. Dropbox, just for reasons I don't know why, made me change my passwords, which was not a minor inconvenience because I use Dropbox on a lot of machines, et cetera. And in fact what was ironic is, because I knew I was going to have to enter it on mobile and so forth all over again,

I made it a much easier to remember password because I had a LastPass-generated password. And I thought, well, if I'm going to have to enter this 20 more times everywhere - fortunately I did not have to enter it because the token was not invalidated by the new password. Which seems to me a flaw.

Steve: Yeah, that's odd.

Leo: So I did create a new password, but I didn't have to reenter it on most of my mobile devices. They already had a token.

Steve: Right. You were still logged in with the old password.

Leo: Seems like Dropbox kind of dropped the ball on that one. They should have invalidated the tokens, shouldn't they? Otherwise, what's the point?

Steve: Yeah.

Leo: So I'm kind of doubly angry at them. Vern Mastel at the Bismarck Public Library, Bismarck, North Dakota shares some very real-world experiences with antivirus solutions: I admin the Windows network for a medium-size public library. The system has more than 200 computers of all varieties, from Windows 2000 to Windows Server 2008 R2. All machines - Windows 2000, wow.

Steve: Yeah.

Leo: All machines - it's like Steve Gibson. All machines are patched on schedule - except for Windows 2000 - have unused or unneeded ports and services closed or disabled, have commercial antiviruses. But still, still, still in the past three years I've had plenty of encounters with malware infections on staff machines, always as a result of drive-by-downloads from hostile web sites.

For many years I used Symantec antivirus products. They failed repeatedly. I have tried others with similar results. Licenses are expensive, several thousand dollars a year, lots of money out the window. I've taken the opportunity, when faced with compromised machines, to test all the various malware detection and removal tools I could get my hands on. My success rate with this approach is zero, 100 percent failure. Usually the tool finds nothing wrong, when it could actually be run. And when it did, it was unable to do more than simply scuff up the malware. That's a good way of putting it. It was like shooting a pistol at a tank. I always end up formatting and reloading the computer. I no longer waste my time testing. I simply go directly to the wipe-and-reload.

Of all the different software products we have on our computers, antivirus/antimalware products are the only ones I can't actually test. Sure, I can feed EICAR files to the AV, but that's hardly a definitive test. Out in the real world,

my real world, it's a total bust. I read all the published test reports about how well the commercial products work, but I no longer believe them because I have never been able to reproduce their results. I cannot get a list of web sites, say 10 or 20 known to be malicious, and then use sacrificial machines to test the functionality of the antivirus/antimalware. Instead I'm just burned over and over.

I have contacted all the major "anti" vendors about this. None of them will cooperate. "Trust us, our products work. That'll be 2,250, please." And by that I mean two thousand, two hundred fifty bucks. "Oh, and yes, we do take credit cards." Here's a challenge for you. If you were in a situation, how would you test an antivirus product? Thanks for listening.

He's got a good point. You can't test it in the wild very effectively. I mean, you can only test it with these fake - EICAR is a kind of a synthetic test that's a virus bundle. Which all the antimalware companies know what's in there, so...

Steve: So they make sure they pass that test.

Leo: We're going to pass that one.

Steve: Yeah, I mean, I really sympathize with Vern. I remember there was a period when a good friend of mine who's really at the expert level with computers, you met Bob when we were up in Vancouver that time.

Leo: Oh, yeah, yeah, yeah.

Steve: He just got a bee in his bonnet once because some friend of his got their machine infected, and he was determined he was going to remove this malware. And he just kept calling every couple of hours and asking what I thought and asking if I'd heard of this file and so forth. And, I mean, he really knew Windows inside and out. And he was never able to root this thing out of the machine. It had just dug itself in and hidden parts of itself and renamed critical files and, I mean, it was just - it was impossible to remove it. So I got a kick out of Vern's comment about just scuffing up the malware and being unable to get rid of it.

And I don't want to say that AV is a scam. It's not that, certainly. I know that it provides benefits for people, or I know that it can. But when Microsoft began offering their own solutions, it was easy for me to say, I'm just going to use that. I'm going to use Microsoft Security Essentials. It's continuing to score well and get good marks, and it's there. And Microsoft doesn't want Windows infected, and I never bet against Microsoft on things that they really care about. They generally end up winning in the long run. So, and you and I, Leo, have historically been not big users of third-party AV. We just really watch our behavior. And of course Vern doesn't have that opportunity because he's...

Leo: He knows people are going to behave badly in a library.

Steve: Yup. He's got a 200-machine network of miscreants that are going to constantly

cause a headache. So I can sympathize. And I really get his frustration. I don't know, I don't think there is a solution. I think it's accept the fact that these Windows machines are just prone to this kind of attack. And maybe he's...

Leo: Well, in an unusually harsh environment. And so that's part of it is that. And also, I mean, get rid of the Windows 2000 machines. They're not being updated. So, I mean, they're past end-of-life. So Microsoft is not fixing security exploits.

Steve: Way past, yeah.

Leo: So I think for all Windows 7 machines he probably could with some certitude lock them down a little bit better than - this is a harsh environment.

Steve: Yeah. The harshest.

Leo: The harshest, right. Were they public computers as well as staff computers? I wasn't clear.

Steve: Yeah, well, he said medium-size public library. He mentioned staff machines. But I would...

Leo: Yeah, but maybe he's, I mean...

Steve: Unless it's a huge, I mean, 200 machines, that's got to be some carrels that are publicly available.

Leo: Yeah. So that's absolutely the worst-case scenario.

Steve: Yeah.

Leo: Because those are inexperienced users who really don't care.

Steve: Yeah. And in fact...

Leo: And so they're doing any kind of weird stuff.

Steve: ...they may very well be going to the library to do their...

Leo: Bad stuff, yeah.

Steve: ...shady downloads and...

Leo: Their porn collections, yeah.

Steve: Exactly.

Leo: Now, I do think Microsoft SteadyState, which unfortunately Microsoft stopped making, but will be part of Windows 8, and the Faronics Deep Freeze, those are used frequently in public computers and work quite well, I think, where you just basically, every day, you start fresh.

Steve: You start over, yes.

Leo: You reboot it, and you're just like you were. And that would be, to me, the best solution on those computers. Obviously not so good for staff computers. I don't understand why people just don't like it when all your data gets deleted each time, but...

Steve: Yeah, they're picky, you know, Leo.

Leo: But for the public computers - and maybe he is doing that. Maybe these are only the staff computers that are really a problem. I would like tomato basil soup, thank you very much. I was just being asked a question, and that was the answer. You know what it means? It's time for lunch.

Time to say goodbye to Steve Gibson of GRC. He is the creator and the guy behind the best hard drive utility ever made. People sometimes say, oh, come on, Chkdsk or Norton Disk Doctor. You don't understand. You don't understand. Steve invented SpinRite before these programs came out. He was the original. And since he wouldn't license it to these guys, they invented their own reverse-engineered, not-so-good version. There's one and only one. SpinRite, baby. GRC.com is the place to get it. You can also find free stuff, lots of it. In fact, Steve's really a Good Samaritan. He gives away a lot more. This is the only thing he charges for. Although are you still working on this encryption solution thing you were going to do?

Steve: Eh, it's there. I've got to - I need to get a bunch of the things that I've almost finished, finished. And then it's time to get back and give SpinRite some time and catch it up with some things that have occurred...

Leo: Oh, interesting.

Steve: ...since 6.0 was finished. So I'm going to do a 6.1, which will be free for all users of SpinRite because I feel it's my responsibility to keep it current. And then I'll look around and decide what makes the most sense, once I've got SpinRite current.

Leo: As long as you keep doing Security Now!, I don't care what you do with the rest of your life.

Steve: I'll be right here every week, my friend.

Leo: I just want to see you here on Wednesdays.

Steve: Absolutely.

Leo: 11:00 a.m. Pacific, 2:00 p.m. Eastern time, 1800 UTC. Watch live at TWiT.tv. Radu did. But you can also listen after the fact. On-demand versions are available in a variety of formats. In fact, this show is available in more formats than any other. There's a transcription that Steve pays for and gets done, and that's on his website, as well as a 16Kb audio version, which is the smallest multimedia version of the show. We have larger, higher quality audio as well as video available on TWiT.tv/sn.

Steve: Yup.

Leo: Hey, Steve. Great show. Do we know what we're doing next week, or is it a question?

Steve: You know, I've had near field technology...

Leo: Oh.

Steve: ...on my mind a lot lately.

Leo: Yeah.

Steve: So I think we need to talk about the technology of near field because it's being adopted in phones and in laptops, and it makes me nervous. So let's maybe take a look at whether that's justified.

Leo: Very timely because NFC, well, Apple decided not to put it in the iPhone. But it is in an increasing number of Android phones. It's in my Galaxy S3 and...

Steve: I've got it in my BlackBerry.

Leo: Yeah. So NFC, which is a variant, a near field variant of RFID.

Steve: Yup.

Leo: And we've done an RFID show, haven't we? [SN-278]

Steve: Yeahhhh [SN-278].

Leo: So maybe we need to, yeah, how does this stuff work. It's a fascinating...

[Talking simultaneously]

Leo: 'Cause these things are passive.

Steve: Yes.

Leo: But from inductance they get, I mean, it's really a clever hack.

Steve: Yeah, it's cool. They're like little transponders.

Leo: I have - this is an NFC tag which Samsung sells, but other companies sell these. And if you look on the back of it, it's circuitry.

Steve: Yeah.

Leo: It's really kind of cool. All right. There's memory on that. That's amazing.

Steve: There's nonvolatile memory. There's counters. It's powered by the - it's passive and powered by the reader.

Leo: Isn't that wild?

Steve: Yeah.

Leo: Anyway, that would be a good topic. Well, if you want to do that next week, I'm all ears. But no matter what...

Steve: I think we're going to plow into that and figure out what's going on.

Leo: Good. Make sure you tune in next week and every week to Security Now!. We'll see you next time, Steve.

Steve: Thanks, Leo.