SECURITY NOW!

Transcript of Episode #284

Listener Feedback #109

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-284.mp3
Quarter size (16 kbps) mp3 audio file URL: http://media.GRC.com/sn/sn-284-lq.mp3

Leo Laporte: This is Security Now! with Steve Gibson, Episode 284, recorded January 19, 2011: Your questions, Steve's answers #109.

It's time for Security Now!, the show that covers your security and privacy online. And here he is, ladies and gentlemen, the man in charge, the key keeper, Gozer, the Keymaster - no, Steve Gibson of GRC.com, the Gibson Research Corporation; the creator of SpinRite; the man who discovered the first spyware, named it spyware, and created the first spyware fighter, a task which he has handed off, of course, to many other companies since. Hey, Steve. How are you today?

Steve Gibson: Hey, Leo. Great to be with you again, as always.

Leo: A Q&A today.

Steve: Yes, our 109th Q&A for Episode 284. And not much has happened, happily.

Leo: Yeah, that's always good news. No news is good news in the security world.

Steve: Exactly. I'm never complaining when that's the case. We have, I mean, the other culprits, we had Microsoft, of course, with their second Tuesday, Patch Tuesday as it's become known, last week. But Adobe and, I mean, I imagine Google's Chrome browser is probably continuing to sneak forward and just keep itself updated in a stealthy fashion, as they have designed it. But we've got a little bit of news and some errata, and then 10 great questions and comments and thoughts from our listeners.

Leo: We should remind people that every other show we do a Q&A, and you can always leave your questions for Steve at GRC.com/feedback. And that's where you got the questions for today. Steve culls the questions. We'll get to those in a second. But what's happening in the - there is some security news.

Steve: There is a little bit of news. I noted, actually I chuckled to myself because our listeners may remember that last week on our podcast I made the claim or the statement that, given what was still coming out about the Stuxnet worm, which was believed to have been designed specifically to go after Iran's nuclear enrichment program, when I learned that it was masking its presence by sending back information to the monitoring systems indicating that the systems were still running, that the centrifuges were still running at the specified and correct speeds when in fact it was screwing around with them to make them run faster so they would damage themselves, I said, okay. If this is all the case, there's no way that this wasn't tested extensively beforehand by someone who had access to exactly this equipment. That is, you could not theoretically design this software just from specs and not put it through extensive testing.

So what I got a kick out of was the front page story of The New York Times Sunday said that Israel and the U.S. - and it named the locations where exactly this equipment was set up and used for developing and testing the operation of the Stuxnet worm.

Leo: They had the centrifuges. It was exactly what you posited two weeks ago.

Steve: Right.

Leo: In fact, somebody sent me an email saying, "Well, the Times finally caught on." So, well done, Steve. You predicted it.

Steve: You just couldn't design something that was this specific to this hardware without having, like, a lab set up that had exactly, I mean, a prototype of what existed in Iran had to be somewhere else. And so, and I noted that John Markoff was among the reporters that was in this. And of course he's been with the industry for a long time.

Leo: He's the most technically sophisticated reporter in the business, mainstream reporter in the business, I would guess.

Steve: Then I got a kick out of another little bit of news, and that is that on June 8th, which happens to be a Wednesday, so we'll be recording a podcast on June 8th, is the first global-scale IPv6 trial date.

Leo: Yay, we can be there. It'll be like the Y2K; right?

Steve: Well, and what I noted was it's like, wait a minute, we're running out of addresses in November.

Leo: Yeah, they're not wasting - they're not, hmm, getting on it that quickly. You know we're going to have iPocalypse November; right? I mean, it's just going to melt down; right?

Steve: Well, actually one of the questions that we'll be getting to later on in this podcast is what exactly is it that's going to happen when we run out.

Leo: Right. Good.

Steve: And so...

Leo: Imageddon.

Steve: ...we'll talk about this. So what's happening on June 8th is that Facebook, Google, Yahoo!, Akamai, and Limelight, at least those five services, are going to be making their facilities available over IPv6. Now, actually Google has had search on IPv6 for quite a while. You could, if you had an IPv6 address, you could go to ipv6.google.com and come into Google through IPv6. So they're already been there. But the idea is that, for 24 hours on June 8th, all of those services will be available specifically through IPv6.

And again, while that's good, I mean, this should have been, like, a year ago. And so it's like, okay, I guess, I mean, it's like, well, let's see, June, July, August, September, October, November. I mean, we're guessing November timeframe for running out of this. And I saw someone trying to explain to someone who didn't get it what IPv4 depletion was. And I had tried to describe it myself to some friends not long ago. And someone said, well, just imagine, like, the phone system running out of phone numbers.

Leo: Which it kind of did. Remember, we got all those new area codes? That's what happened.

Steve: Exactly. The original structure, it used to be, for example, that an area code, the center digit was either a zero or a one, which is why we had 714, 405, 213...

Leo: Right. And the bigger metros had a one because it was faster to dial a one than it was a zero on the dial.

Steve: And the primitive electromechanical dialing systems, they used that key of whether the center digit of three was a zero or a one to determine whether it was an area code because none of the prefixes had that. And so that allowed them to disambiguate those two things. But the other thing, the other place where we ran out, remember the 800 numbers. It used to be that toll-free numbers were only 800 and then something. Well, they did run out of 800 numbers and then had to add [888] and a few others.

Leo: Oh, right, yeah; 888, yeah.

Steve: So we've sort of seen that before. But anyway, so we'll talk about the consequence of that later in the show. But I did sort of think, yes, well, it's good that we're having a global-scale trial of IPv6 this summer. But it does seem a little bit late for relative to, you know...

Leo: I can't wait. So a little later on in the show we're going to find out what that means, what the consequences would be in November or whenever this runs out.

Steve: And then another very good piece of news, apparently there has been pressure from the Federal Trade Commission, FTC, the U.S. FTC and others about the fact that Flash is being so pervasively used to store tracking data in the so-called LSO, the Local Storage Objects which Flash supports. And we've talked about the fact that there are companies that specialize in reconstituting deleted browser cookies from the so-called Flash cookies, which are these LSOs, and that there's no obvious or simple user interface for Flash. Flash is a browser plug-in, so it's sort of transparent. You go to a site that's using Flash, and things happen. Ads jump around. In fact, their whole website could be Flash-based, as is the case in some cases where - but Flash itself doesn't have a user interface. It's just meant to be used as a plug-in.

So what Adobe has done, and I'm led to believe that Chrome within several weeks will be the first to have a UI for it, Adobe has published an API, an Application Programming Interface, for access to their local storage objects for the sake of allowing them to be deleted by browsers which could be providing a user interface to that, Chrome apparently being the first. But Mozilla, Google, and Adobe - I'm sorry. Mozilla, Google, and Apple teamed up with Adobe to do this. So I would imagine that Firefox will offer the same sort of thing very quickly since they were part there, and presumably Apple with their Safari browser. So that's just good news. That gives users the ability to do this.

The reason browsers are involved at all is that browsers are essentially presenting this facility through Flash to the user. So it does make sense, since browsers have a mature user interface of controls and settings and toolbars and so forth, why not allow the object which is instantiating the Flash object, which is the browser, to also give it the control that it otherwise lacks? So that's good news. We'll keep an eye on that. And as announcements of browsers that are supporting this LSO deletion occur, I'll let our listeners know.

Leo: Very good.

Steve: I've been a user of the eBay and PayPal dongle, which we talked about quite a while ago, ever since we talked about it. It's that little football, as we've often referred to it, which you press the button, and you get a six-digit code, which you then enter into, for example, PayPal when you're logging into PayPal to authenticate yourself. It's a one-time password approach that gives multifactor authentication. But the problem is, being a physical token, if you ever want to, for example, in my case, purchase something on eBay or through PayPal when it's not with you, that's a problem.

Just the other day I decided - I was, like, browsing around and noted that VeriSign, which is the source of these things, supported a BlackBerry client. So I added it to my BlackBerry and was very pleased and impressed with how simple this is, and essentially how functional it is. And this is now available for many of the different smart phones. I didn't think ahead to check to see if it's Android.

Leo: Yeah. I think it's - yeah.

Steve: I think it's iPhone.

Leo: I think both.

Steve: I know it's - yes. And so essentially, when I run this on my BlackBerry, run the little app, it comes up and shows me the serial number for this instance, which is unique, of this instance of this little authentication app that I'm running, and a six-digit code, and also shows me a little expiration timer, which counts down from 30 seconds. And every 30 seconds it changes this number. And so they show me that so that I'm able to see whether I have time to type it in before that expires. And I simply registered the serial number that was presented on the screen with PayPal, telling them, look, I have a second authentication dongle now. And now I'm able to do the same level of multifactor, one-time-password-style authentication wherever I am because I've always got my BlackBerry with me when I'm out roaming around. So...

Leo: Yeah, I think that's better than a dongle, I really do, because...

Steve: I agree.

Leo: ...people have their phones all the time. I really like doing it that way.

Steve: Yes. I think it's terrific. And in fact now I'm back having to tell PayPal which of the two objects I'm using, which is annoying because it makes me go through that step. Whereas when I only had one of them, it knew which one I had, and that allowed me to just add that six-character or the six-digit passcode after my password in one phase. So I'm now seriously considering deregistering my football and always using my BlackBerry. So I wanted to let our users know that this has all actually really happened. Because I know when we talked about the football originally there was a huge amount of interest in it as an opportunity for increasing security. And now that VeriSign has pushed all these little clients out to smart phones, as you say, Leo, it's just a terrific solution.

Leo: Yeah. Yeah, it is on my Android, I believe. And I believe I saw an iPhone app, so I think it's everywhere. I mean, it would only make sense. If they're going to put it on BlackBerry, they're going to put it on iPhone and Android.

Steve: Right. Probably first, in fact.

Leo: Yeah, exactly.

Steve: So finally, in errata, I got a note from someone who didn't disclose his full name. He called himself Ken F. He's a cybersecurity manager with an undisclosed government agency. And he said: "Steve, huge fan of TNO" - which of course is my acronym for Trust No One - "and been listening for the past two years" to Security Now!. "A few episodes ago you were discussing the exfiltration of the data from the government's classified networks relating to WikiLeaks. I wanted to provide you the correct pronunciation of that classified network." I was calling it sy-per-net, SIPRNET. And he sort of breaks it up phonetically, and he says it's pronounced sip-per-net, so SIPRNET. He says, "Thank you for all your stellar work in this field, along with breaking down the many complex issues to usable and understandable chunks." Ken F., cybersecurity manager from an undisclosed government agency.

Leo: I love that.

Steve: So, SIPRNET.

Leo: SIPRNET.

Steve: Now we know. And I did have a short little note from another listener of ours, Tom Leonard, who said he wanted to drop a note to let me know about SpinRite. He said: "I just recently purchased it as I provide computer tech support for the South Dakota School of Mines and Technology in Rapid City, South Dakota."

Leo: I always loved that name. It sounds so 18th, so 19th Century. Mines and Technology.

Steve: Yeah, South Dakota School of Mines and Technology. He says, "I found out about" - well, it's sort of like 3M, you know, the well-known 3M Corporation?

Leo: Right, Minnesota, what is it, Minnesota Mining and Milling?

Steve: Manufacturing.

Leo: Manufacturing, yeah.

Steve: Yeah, that's what 3M actually stands for. It's like, okay. Good thing they shortened it. Anyway...

Leo: And you know KFC stands for Kentucky Fried Chicken.

Steve: Right.

Leo: They didn't like the "fried" part anymore, yeah.

Steve: "I found out about your product from watching you and Leo on the TWiT network," which is what's happening right now. "Specifically, your Security Now! segment. I truly missed Leo and his time he spent on The Screensavers and Call For Help. But now I think he's found his niche and is truly the best for us geeks.

"Anyways, on to my story. I purchased your product about six weeks ago and thought, when am I going to try this out? Well, this morning I went to boot my personal Netbook, and the Blue Screen of Death shows up. Well, since there's no floppy or CD drive, I had to create a bootable USB flash drive. The first flash drive I tried wouldn't boot, so I went to your FAQ page for SpinRite and found that some flash drives don't work properly. So I tried another flash drive. This one worked fine and booted right into SpinRite.

"I ran the recovery and repair level, No. 2, I believe, and it started checking things. It got to about 7 percent and just seemed to stall, but it appeared to be continuing to work. It moved the files and sectors, declared one sector to be unrecoverable, but apparently recovered most of that sector's data. After about four hours total, it finished. I rebooted, and up came the computer. It now starts up and shuts down much faster. So this area of the disk must have been going bad for a while. Steve, thanks again for SpinRite. And I plan to keep on watching you on TWiT with Leo. Thanks, Tom Leonard."

Leo: Aw. I think that's generally the case. A lot of people complain about their systems slowing down. And many times it's merely one or two bad sectors on the drive which Windows spends a lot of time trying to read in order to boot or whatever.

Steve: Exactly.

Leo: So moving stuff off of that one sector can make a huge difference.

Steve: And then SpinRite will remove that sector from use, causing the drive to swap in a spare. SpinRite will then replace the data. And one of the things that SpinRite can uniquely do is, even if it's finally absolutely unable to recover every last bit of data, it will return as much of the 512 bytes as it was able to read, even if not all. And sometimes that makes a difference. It'll let you know that that's what it did. But, for example, if that's a chunk of directory space, it may actually be that the part that it couldn't read wasn't necessary, but was just slowing things down.

Leo: Could be slack space, yeah.

Steve: Even an unrecoverable sector, it's able to get most of it.

Leo: Right. Rob in Melbourne with our first question of the day. Are you ready, Steven?

Steve: Absolutely.

Leo: He says: NoScript is already adding do not - I shouldn't do my Australian. It's terrible. And every time I do it, I get an email from somebody saying, "Hey, mate, we love the show, but don't try to talk Australian." Steve, just thought I'd drop you a line to tell you that NoScript seems to be adding headers to my HTTP queries regarding web tracking, specifically a header "X-Do-Not-Track" and "X-Behavioral-Ad-Opt-Out." Cheers from sunny Australia, Rob.

Steve: Well, when I saw that...

Leo: First of all, somebody has to abide by those headers; right?

Steve: Well, yes. You may notice that, I mean, I've asked for this, I mean, suggested that this would be a great solution for dealing with the problem with tracking. But this was the first I had heard that NoScript might be doing that.

Leo: But is this a standard? I mean, are these headers standard?

Steve: No.

Leo: No.

Steve: Well, so, but that's fine because we have a chicken-and-egg problem here. I mean, someone has to start doing this so that these things exist. And so I fired up my packet capture, refreshed a Google search page that I had up, and sure enough, those two headers are being, ever since the 28th of December, 2010, so not yet for a month, so I think it was v2.0.9 is when Giorgio added this to NoScript. And so I shot him a note saying, "Hey, this is fantastic," and we corresponded briefly about it.

What he wrote on his posting was, he says: "From now on, a web browser with NoScript installed warns every HTTP server it contacts that its user does not want to be tracked, i.e., that his data must not be collected for profiling and persistent identification purposes. I believe this is a safe assumption about the feelings of most, if not all, NoScript users. As stupid as it may sound (why parties who are interested in tracking you would comply?)...

Leo: True. I mean, it's not in their interest.

Steve: Correct, "...a means to clearly express your will of not being tracked is going to be useful, especially when backed by law or industry self-regulation, as explained here. Therefore, it seems in the interest of NoScript users and privacy-concerned netizens in general to participate in this effort. In its current release, NoScript allows the 'Do Not Track' feature to be disabled or tweaked by opening about:config and editing the noscript.doNotTrack.* preferences...."

And so he's got three preferences where you can enable it, the whole system, or not, meaning that you could disable it if you just didn't want that to be added to your queries. Then you can list a set of URL patterns, which are space-separated, of destinations, query destinations which are not to be sent the do-not-track-me message. And then you can also - so that's called "exceptions," where you can have exceptions to the add-the-do-not-track header. And then, thirdly, you can force specific ones that do qualify to still be forced. And he says, a graphical user interface, "A GUI for these options, and possibly finer grained controls (e.g., to allow some or all of the third-party trackers on certain websites only) will be added in future releases."

So anyway, I just - this is great. This is something - we know, for example, do-not-track legislation is being considered. There's talks of this happening. I can't think of a better place, as I have said often before, than for our browsers to simply add a header that says "I do not want this query to be tracked." And it's here. So, yes, it's true, no one obeys it yet. But we have to have it for anybody to obey it. So it would be great, for example, if Safari and Chrome and Opera and the other browsers, and IE, were to pick up on this and implement the same thing. Then it would be a simple matter of legislators saying, look, if somebody has this in their headers, you're not to track them.

And then of course sites would have then the ability to say, well, we really do need tracking in order to support ourselves. So then they would be able to present to the user a notification saying, whoa, you've tried to enter the site with tracking blocked. You're going to have to make an exception for us if you want our content. In which case the user could decide, eh, don't need it that much, and vote by saying no, thanks. Or they'd say yes. And then, again, Giorgio and other UI designers could make it simple to add an allow-tracking exception on a site-by-site basis. So if this all happens, we're beginning to get to where we want to be.

Leo: Another great reason to use NoScript.

Steve: Yes.

Leo: You know, just as a side note, I've seen these X headers in email. You can have an arbitrary X header in email. And whether the server sees it or not or acts on it or not is completely up to the server. I didn't realize you could also do the same thing with HTTP requests. It's the same mechanism, I guess.

Steve: Yes. The idea is that the X- as the prefix says this is not part of the standard. So, for example, we'd have query headers, for example, like the expires or the referrer, the HTTP referrer header. And so any query does have headers which are part of the so-called metadata. There's not something that the user sees, but it's something that the browser is sending. And in the spec it says non-spec-specified headers, that is, sort of optional headers, can be included just by sending X- and then the header name. And in this case what I saw in the packet capture was that they were the header:1 for X-Do-Not-Track, and the same thing for X-Behavioral-Ad-Opt-Out was a :1, so essentially saying true that I do not want to be tracked. So, yes. Exactly in analogous fashion, as you said, Leo, to email, this can be done. I know, for example, because I've done a lot of work over in NNTP, the Network News Transport Protocol that newsgroups, for example GRC's newsgroups, use has the same sort of facility. And I invented some headers for our own purposes that run in the same vein.

Leo: Okay. Question #2, Jamie in England - I won't do an English accent, either, or a phony one, anyway - wonders about IPv4 doomsday. Steve, when you were recently talking about IPv4 address depletion, you said that the day we run out of all IPv4 addresses would be doomsday. Well, how can this be the case? Surely all the equipment we already have on the 'Net will be fine and continue to talk to each other. It's just that no one new will be able to join us. Am I correct? So it's only a mild concern; right? Could the new clients joining the 'Net not simply go through an IPv4 proxy to talk to the rest of us? Thanks, love the show, keep up the good work. So what does it mean, IPv4 Doomsday?

Steve: Okay. So first of all, I wasn't...

Leo: I guess we should say upfront, just to set the stage, anybody that listens to the show I'm sure knows that IPv4 dotted quad allows for, what is it, two billion addresses?

Steve: Four.

Leo: Four billion addresses.

Steve: 4.3 billion different unique IPs.

Leo: Every computer that is on the Internet has to have a unique public address, just like a unique phone number. And we're running out. We've gone through, we'll have gone through four billion addresses in November, roughly.

Steve: Right. Essentially, back when the Internet was being designed, there was - it's very much like the same story with RAM. Remember that, for example, the Apple 4 allowed you, the original - I mean, Apple 4.

Leo: II.

Steve: The original Apple computer allowed you to have 64K of RAM. And we all knew no one would ever need more than that.

Leo: Plenty.

Steve: Plenty, exactly. I mean, what could you possibly do with more than 64K? So this is not the first time we've ever, like, run out of resources one way or the other. We tend to do this because the technology lives much longer than we expect. It ends up not being obsoleted as quickly as we expect. It just wants to grow forever. So back when the original designers established 32-bit addresses, even then they allocated it inefficiently. So there are big chunks of that 4.3 billion addresses which cannot be used. We've talked about this several times in the past, so I won't go all the way through it. But we are, around the end of this year, running out of space.

Now, if I ever said "doomsday," then I'm not happy with myself for having declared it doomsday because it's not doomsday, just exactly as Jamie in England asks. It is exactly like you suggested, Leo, if we ran out of phone numbers. Well, if we ran out of phone numbers, then people wanting new phone numbers would have a problem because all the existing phone numbers would be in use. But the ones that were already there would still work.

So the bad news is that the transition from IPv4 to IPv6 is going to be an incredible mess. There isn't an elegant way to do it. I think it's one of the reasons everyone is dragging their feet as much as they are, the reason we're not even doing this full global test of only five large sites until summertime. It is a catastrophe just that everyone wishes and is hoping somehow we're not going to have to address. I mean, even for me, my entire infrastructure, GRC.com's infrastructure is all IPv4. ShieldsUP!, all the code that I've written, all of my packet management stuff, everything is 32-bit IPv4 addresses. And so the day that I have to bite the bullet and implement this as all IPv6, I'm not looking forward to. I mean, I can because I wrote all this code. It's all my own raw packet stuff. So there's nothing preventing me from manufacturing IPv6 packets except I'm going to have to go off and write a whole bunch of code that I'm not looking forward to.

So certainly on this podcast during 2011 - as I have said before, 2011 is going to be the year that IPv6 really does happen - we will be talking about transitional things a lot. We'll be talking about the conversion from IPv4 to v6, proxying and NATing and gateways. There are things like IPv4 tunneling, where you tunnel IPv6 content through IPv4 through network segments that aren't IPv6 aware, but they still can support tunneling. I mean, it's just going to be a real nightmare. So we'll have lots to talk about.

Obviously we'll get there someday because this IPv4 depletion really is going to be putting, finally putting a lot of pressure on our ISPs and network engineers to start taking this very seriously. Everyone, including me, has been able to ignore it until now and wants to continue ignoring it as long as we possibly can because it's just going to be a lot of work without any feature change. Basically, it's not like we get anything, any great new benefits from it. It's just a lot more address bits is essentially all that happens.

So the world doesn't end. And I'm sure what we're going to see is IPv4 having a presence on the Internet probably forever. I don't think it's ever going to go away. It'll just be always there. I hope I get to keep all my IPv4 IP addresses, and IPv6 users will still be able to get to me that way. So, and I already have had people asking, hey, when is ShieldsUP! going to support IPv6? And it's like, uh, I don't know when. But...

Leo: So is there a workaround if somebody - okay. So in November we run out. And...

Steve: Okay, so for example...

Leo: Well, everybody has pools of numbers that aren't allocated. So your Internet service provider probably has plenty of free numbers that they can allocate.

Steve: Yeah. Look, for example, at cell phones. You could argue that cell phones are probably a class of device that really doesn't care what its IP address is. The user doesn't even know. There's, like, there's no transparency to the end user about the IP address of a cell phone. So that's a place where you could easily - a high-growth place because that's where lots of these, like Verizon is adding support for iPhones. And so that's a place where you could easily sort of transparently bring up IPv6 in a way that end users would not be made uncomfortable at all. And so, yeah, a place where you could have a huge increase in space.

Leo: Okay. I guess I won't worry about it.

Steve: It'll just happen.

Leo: I'll let you wise guys figure it out.

Steve: And we'll be talking about it all year long. It's going to continue...

Leo: Oh, yeah. There'll be a lot to say.

Steve: ...coming up. And in fact, what I'm planning to do, when we do, as we will this year, as I've promised, our from the ground up how the Internet works, we'll have a huge chunk of new content which we've never discussed, which is IPv6, all the gory details and all the transitioning nightmares that we're going to be going through.

Leo: Woohoo. Question #3, Tom Zerucha in the Detroit area brings up a good point about SSDs, encryption, and the TRIM command: Steve, if the whole disk is encrypted in such a way that every sector is marked used, it will increase wear and maybe slow things on SSDs since it will have to shuffle full blocks. If only the used sectors are encrypted, instead of the whole disk, then the TRIM command can work to erase blocks for the unused sectors. Windows 7 is, of course, the only OS that currently supports TRIM. This will make it faster, more reliable.

By the way, I guess we should explain that there's a weird effect on SSDs that kind of is like fragmentation, and it can - slows the SSD down. I asked at CES, I talked to two of the guys responsible for Intel SSDs. And Intel is really the crme de la crme of solid-state hard drives. And we're talking about those flash-based hard drives. And I asked about TRIM. And Windows 7 is the only OS that supports TRIM. And most of the controllers, except for the SandForce controller, maybe a couple of others, don't support TRIM anyway. So most hard drives don't support TRIM. And they said, well, what happens with most hard drives is there's this peak theoretical speed. There's a drop as you use it. And then it levels out, and it pretty much stays there. All TRIM does is get it back.

They actually, the two guys debated each other. One guy said, oh, it's important. The other guy said it's not important in real world. So there is debate even whether you need TRIM. But now let's continue on with your answer. I'm sorry, I didn't mean to interrupt. But I just thought that was kind of interesting. They don't even agree.

Steve: Yeah, that's great background, which I didn't have. I have the low-level technology side of it. So here's the deal. Tom's question is a good one, and it raises a very good point because he's responding to my answering a question two weeks ago about the whole idea of full-disk encryption on an SSD. And I think the question was would that slow things down in the same way that fragmenting might on a hard drive. And I said no because, once you were encrypted, the whole drive was encrypted, then you were just reading the same sectors from the SSD that you would otherwise. And this issue of TRIM means that I was incorrect two weeks ago because of the way SSDs work.

So here's the deal. When you write a sector to an SSD, it must erase the contents due to the nature of the physics of the way an SSD drive works. It has to erase the sector before it can write it. The problem is that, again due to the physics of SSDs, an SSD cannot erase a single sector. It is forced to, architecturally, erase a much larger block of sectors. Now, if the SSD knew, if it had a way of knowing that the other sectors in the block were not in use, did not have data in them, then it would not have to first read them and cache them inside itself, then modify the sector to be written, and then write them all back. Because, again, we're trying to change or write to one sector. But to do that we have to erase that sector, which means we have to erase the whole block that that sector is part of. And if we're going to erase the whole block, we have to first read the whole block, then erase the whole block, then write it all back. So you can see there's much more work being done if the other sectors in that block have valid data.

What the TRIM command - oh, and so I should also mention that, in order to deal with this, in order to make SSDs functionally identical to hard drives, they manage all this internally. It's amazing. And you referred to the SSD controllers. That's the job of the controller inside the SSD package which does all of what I just said transparently. It maintains its own proprietary bitmap of every single sector in the space of the SSD and whether that sector has ever been written to. So as you write to sectors in the SSD, those little bits get set, telling it that that sector contains valid data. So over time, as you're writing to more and more of the SSD, more of these little bits are being set.

But at the operating system level, you may be deleting files. When you delete files, as we know, we're only marking those sectors or clusters, because modern file systems allocate in cluster sizes, which is a cluster of sectors, we're marking those clusters as no longer in use. We know that the file system is not going out and actually erasing them because that's how undelete utilities work is they come back and say, well, let's get the data. If it hasn't been overwritten, we can recover the data that was deleted, and the user regretted making that deletion.

What TRIM support in the - well, okay. So file systems are marking areas deleted, but that information is not being given to the SSD. So as the evolution of the ATA, the AT Attachment specification, has progressed, the designers of SSDs wanted to provide a means by which the operating system using the drive could reset those little bits, saying these sectors are in use. So that's what TRIM command does. The TRIM command is an extension to the ATA, the AT Attachment specification, providing a means for communicating to the drive that the following sectors no longer contain valid data as far as the operating system is concerned.

So when we say that only Windows 7 supports the TRIM command, we're meaning that only Windows 7 is a popular operating system in use which, when and as you delete files from the file system, Windows 7 sends a batch of TRIM commands down to the SSD, telling it that those sectors are no longer in use. And so the beauty of that is that it prevents these bits, these little in-use bits from just accumulating without end, which they otherwise would in the SSD, telling the SSD that, as you've deleted files, those sectors are no longer in use.

So getting back to Tom's exact question, he said, if you ran TrueCrypt, for example, to encrypt the entire drive it would set every single one of those in-use bits in the SSD because you have written to every single sector of the drive in order to encrypt the whole thing. And he is exactly right. So what you'd really like to do is run whole-disk encryption and then have a means for sending a full drive worth of TRIM commands telling the SSD, reset all of your sector-in-use bits which you're using to manage those blocks back to zero because, even though we just wrote to the entire drive to encrypt it, we didn't store any valid data there yet.

Leo: Okay.

Steve: So there is a Linux utility called hdparm which has that facility. There is some driver support. There's a utility that Intel has, like an add-on utility where you can manually scan the file system, and it will look at the clusters that are in use and then issue TRIM commands for those that are not. But I've tried to purchase two SSD drives from Intel which offer this support, and I've failed both times, which is really annoying because I care about this kind of thing. So, and as you said, Leo, it's not even really clear that this is more than sort of a theoretical problem. The controllers are doing a very good job of managing their SSDs. It's not like performance continues to descend forever until it becomes really, really slow. You do see a drop as these bits are being set. But then at some point it's not such a big deal afterwards.

Leo: I asked Allyn Malventano, who's the guy who discovered this and kind of publicized it on PCPer.com and who is of course a regular on our both PC Per and TWiCH podcasts. And I said, "Well, Allyn, in order to create this benchmark, and to show this, you had to really kind of create this synthetic, make a lot of small files, erase a lot of small files kind of a situation." So it's not even - the question is not whether this happens, because you can demonstrate it. But the question is whether in real-world use it would be a significant degradation in performance. And even the Intel guys disagreed. That's what I liked about it. One guy said, oh, no, it's a problem. And it's completely moot unless you're using Windows 7 because no operating system except Windows 7 does it anyway.

Steve: Correct.

Leo: So unless your operating system does it, it's not - you're not going to - so it's a really actually kind of an angels dancing on the head of a pin argument. And yet I think it's fascinating, and it certainly is something that is relevant, if you want to use an SSD.

Steve: And now...

Leo: I've never - I'm using it in OS X, so I've never noticed.

Steve: Right. I was saying, and now all of our listeners understand what the whole TRIM thing is with SSD.

Leo: Yeah. Now they're in, man, they're in, they're with it. Jamie Hunt, England - did you write this, or did he write Jamie Hunt in England, UK?

Steve: That's the way he wrote it.

Leo: Okay. I mean, I know they're not the same thing, but I just think it's funny. It's like saying in the United States, North America. Wonders about driver update scanning: Steve, there seem to be millions of sites scattered about the Internet saying they will scan my PC for outdated drivers. Oy gevalt.

Steve: Uh-huh.

Leo: Thanks, but no thanks. But 95 percent of these programs seem to be from an unreputable source, or at least a source with no reputation. My question to you is, do you recommend using a program that will scan for outdated drivers and tell me to update, sort of like Secunia PSI for drivers? And, if so, which one? Thanks.

Steve: Well, you said it, Leo. I wouldn't say that 95 percent of them are unreputable. I would say 99.9.

Leo: It's basically giving them permission to see what you've got.

Steve: Yeah. And, okay. So here's the deal. It is so tempting to want to have the very latest drivers for your machine. But it makes me feel like - I feel even more strongly about this than I do about the temptation not to always update to the latest software because drivers are even providing less incremental functionality than new versions of software. And it's not clear that you get a lot when you update to a new version of software. So I guess what I'm saying is that my feeling is, if some device on your system is not working, then certainly try updating the device driver and see if that fixes the problem. But if it's all working, then I don't know what you gain from updating drivers to the latest versions, hoping that they're going to work better, because drivers don't have that much ability to sort of, like, only work part way. They're generally working or not. And so my feeling is I've never done a scan through some third-party site to, like, have it look at all my drivers.

Leo: Yeah, it's a bad idea.

Steve: I really agree. I think it's a very bad idea. I know that - I'm a Lenovo user. And the Lenovo system has some, for several years now, a facility for checking its own drivers versus its own database and notifying you if things have changed. And invariably what I notice is it's trying to give me new versions of drivers where the only difference is it now supports laptops I don't own. And so it's like, well, okay, why do I care about changing this driver to support a laptop that Lenovo now offers that I don't own?

So I look carefully at what the benefits are of updating a given driver. If it says, oh, we've made great strides in power management, it's like, oh, I need that. That sounds like a good thing. But if it's we now support 16 new laptops, it's like, okay, I don't think I'm going to be making the move for that reason. And again, I would stick with the source of my machine, rather than some random site that says let us scan your machine and let you know if there's something new. Seems like a bad idea.

Leo: Right. Windows itself actually does a pretty good job of keeping drivers up to date. They're not in the critical updates, they're in the optional updates. But that's what I do. I just run Windows update, and I look and see if there's any driver updates because I generally will use the ones provided by Microsoft just because they're tested for Windows. I mean, they may not be the latest always...

Steve: But known to be compatible.

Leo: But they're known good, yeah. And I find that that's fine. It's funny how old habits die hard. And people who've been PC users for a long time have all sorts...

Steve: Skeptical.

Leo: Yeah, well, they have all sorts of things that they do that were maybe needed, like defragging and updating drivers, checking your video driver. And I think as we've gotten to be a more mature system, if you're using Windows 7, I think you probably can eliminate a lot of the stuff that we used to do to make, you know, you don't have to edit your config.bat anymore.

Steve: And defrag your registry.

Leo: Yeah.

Steve: Pretty much, if you're using Windows 7, you've given up.

Leo: Yeah. Well, that's another matter. You know, yesterday I did "Live with Regis and Kelly," and we taped a second segment that'll air in a couple of weeks. They're going to do a Twitter week all week long. And so I went up to Regis's office to tape this segment on teaching him how to use Twitter, which was fun, and he was using Internet...

Steve: Oh, how fun. When is it going to air, Leo?

Leo: January 31st.

Steve: Okay, great.

Leo: I don't want to tip it. But it was really fun. And I think what's most interesting, you know, yesterday he announced that he was leaving the show at the end of the summer, which was a big shock to everyone. No one knew this. I didn't know it. None of the producers knew it. Everybody went - because they're all going, uh-oh. What do we do?

Steve: Well, and Regis is such a fun non-techie that I can imagine...

Leo: He's wonderful.

Steve: ...you guys must have just had a ball.

Leo: I adore him. First of all, he's 80 years old. He is sharp as a tack. I mean, there are very few 80 year olds who are as quick and as with it and on top of it as he is. I read this after he announced the retirement, which was breaking news everywhere, said he is currently the longest-running on-air talent, you know, ever, in the world. I mean, there's nobody been doing it longer than Regis Philbin. He started with Joey Bishop in the '50s; you know? So in any event, so it's a real honor. And I have always liked Regis. I've just - we hit it off very well. And so we're doing this thing.

And after it's done, he's using IE, it crashes. It crashes on him. And he says, "What's this, Leo?" I said, "I don't know, we were just tweeting and it crashed." And Gelman goes, "God, I hate Internet Explorer." It's, like, it's so sad that normal people have to go, huh? I was just tweeting. What happened? And it crashed the browser. I'm sure he was using, I didn't look, but I'm sure he was using, like, XP with IE6. But I didn't really even want to know. It was like, I just stepped back and said, "I guess we're done." I don't think that will make it into the tape, by the way.

We have a long one. This is from Charles Victorian in Houston, Texas, a follow-up on what he learned about frequency hopping spread spectrum. Which - this sounds like it's apocryphal, but I believe it's true - was invented by Hedy Lamarr, the movie star. Did you know that?

Steve: No.

Leo: Doesn't sound right.

Steve: Sure doesn't.

Leo: It doesn't sound right.

Steve: Unless he's a secret genius.

Leo: She was a starlet, a gorgeous woman. Orson Welles dated her for a long time.

Steve: I did hear this once somewhere. I remember thinking, what?

Leo: She invented spread spectrum. I don't know how. I'd like to know more about this. Frequency, I mean, Wikipedia talks a little bit about this. It was patented earlier than that. But it says in Wikipedia, "The most celebrated invention of frequency hopping was that of actress Hedy Lamarr and composer George Antheil, who in 1942 received [a] U.S. patent" during World War II "for their 'Secret Communications System.' Lamarr had learned at defense meetings she had attended with her former husband Friedrich Mandl that radio-guided missiles' signals could easily be jammed." They "used a piano roll to change among 88 different frequencies ... to make radio-guided torpedoes harder for enemies to detect or jam."

Steve: Wow.

Leo: And this patent came to light when ITT and other private firms began to develop CDMA. And they did the patent research, they said, whoa. Hedy Lamarr owns this.

Steve: And 88 keys is the number of keys on a piano.

Leo: Yeah. Isn't that interesting?

Steve: Yeah, yeah.

Leo: Others, there's prior art. But independently, Hedy Lamarr thought of it. I think it's fascinating. Steve, thank you so much - just historic note. Steve, thank you so much for taking on my question about the Lorex Live Snap and its use of Frequency Hopping Spread Spectrum (FHSS) technology. I was encouraged by your comments on the security which might be implemented by the camera system. He had a baby monitor, I think; right?

Steve: Right.

Leo: And he didn't want people to watch his camera, his baby. And he said, well, what is this FHSS they say they use? Enough that I not only opened the one I had already purchased, but went out and bought a second system immediately after hearing the podcast so I could do a little intersystem testing of my own. Now, this is our listeners. This is why I love our listeners. He didn't just say, oh, okay. He said, let me see.

Reading over the User's Guide, which I previously didn't have access to, explains how to pair up cameras, which gave me hope. The guide states, "The camera(s) included with the monitor have already been 'paired up' with the monitor." So that's why he didn't have to do any configuration with the new one.

Steve: Right.

Leo: It goes on to explain how to pair up additional cameras since the base system comes with two cameras, yet the monitor supports up to four. I won't bore you with the details of the relatively simple process. But I will note in particular it requires you to begin with the camera turned off. Additionally, under the "Tips" section, the guide states, "The camera and monitor should be around a foot apart during the pairing process."

So, with some confidence gained by your coverage of FHSS on the podcast, and a better understanding of how the product works from reading the included User's Guide, I tore into the second box. While using a camera and monitor from the first set, I put the second monitor in "pair" mode. It responded with "Pairing" and some symbols indicating to wait. The process exited with "No Device Found" even though this monitor was as close as I could get to the

currently broadcasting camera (they were plugged into separate outlets, and I didn't have an extension cord handy). Nevertheless, any neighbor, et cetera, is not going to be able to get closer than I was without being inside my house. So it is clear that, once the camera is paired to a monitor, another stock monitor can't just show up and receive the signal. A good start.

It seems like, as in Bluetooth, the pairing process could leave you vulnerable for a few seconds, when you're discoverable, effectively; but then the signal, once it's locked in and paired, should be inaccessible. It also seems that you would know if someone had hijacked your camera since your monitor would then say "No Device Found," and that would be a clue.

I know that this doesn't address reverse-engineering the system or building some sort of separate hacked monitor, but at least it isn't going to be easy thing for somebody to drive by, set up a monitor, and watch my baby. Thanks for doing what you do. I always look forward to listening to every episode. I really miss Tech TV, but at least we still have you and Leo. Regards, Charles Victorian. I like that kind of listener.

Steve: So, well, yes. And the fact that they referred to pairing, of course, that's the first really great piece of news. Our listeners will probably remember that basically we covered frequency hopping spread spectrum several times relative to Bluetooth. And then when I first entertained Charles's question he was - he hadn't even opened the box because he wanted to know whether it could be safe enough, or whether it was just a scam. And of course we couldn't answer the question, but I looked at their website and explained what frequency hopping spread spectrum was. Then that encouraged him to open the box, where he found the manual.

And then they do take him through a process by which the transmitter and the receiver need to get to know each other. And then he further, as his note indicated, further tested this by taking, not only just some random monitor, but the company's same brand of monitor and tried to see whether it would be able to receive the same signal when it hadn't been paired with the camera, and it wasn't. So again, I think this is just a great example. I wanted to follow up, but also to give our listeners a sense for how you can determine this kind of thing on your own for anything else of this nature in the future.

Leo: Yeah. It's a little inductive reasoning, really.

Steve: Yeah.

Leo: Question #6, Brian Voeller in USA.Oregon.Medford - I like how people are identifying themselves. That's great. Planet Earth in the Milky Way. No, Solar System, Milky Way, Universe. Universe X394.

Steve: Now you realize that's how I'm going to get these things sent to me in the future.

Leo: From now on. Narrow it down. Hello, guys. Regarding Episode 282 and the question about the security of a wireless baby monitor camera that touted frequency hopping as a security mechanism, I would not regard that as effective, particular in a video transmission context. Being a cheap camera, it's probably sending analog NTSC, which is what standard definition television in the U.S. is, or EIA, that's a low quality version of NTSC, and hopping frequency on the completion of each frame, since the vertical blanking interval would make a convenient opportunity to let the tuner lock onto the next frequency. All that makes sense. Individual frames could be captured by scanning the frequency range slowly. Assuming 256 channels, once you found one of them, you'd get one frame every 8.5 seconds. Oh, that's an interesting point.

Steve: So, yeah. I saw Brian's question or comment and thought, well, this makes a good companion to the one we just read, and that is to suggest that frequency hopping is not security, as he does. And of course he's right. Frequency hopping is not encryption. It is mostly used to avoid jamming and interference. This is exactly what Hedy Lamarr apparently developed this concept for back in war time, was not so much for security, but for avoiding interference, the idea being that, if one particular frequency is being jammed, you're not going to spend much time there. You'll still be able to get the bulk of your signal through.

So I feel, I mean, Brian's point is that, if somebody was absolutely determined to monitor this information, what they'd be getting is essentially a very nonstandard signal. They wouldn't be getting a video signal, they'd be getting one frame of video. Assuming that it's not digitized, and it's not encrypted - which we don't know, it could be digitized and encrypted in addition to the frequency hopping spread spectrum - but they'd be getting one frame on a given frequency, as he said, every 8.5 seconds. But again, you'd have to then capture that and figure out how to display it because no regular monitor is going to display that.

So anyway, I still feel that this makes great security, much more so than just having a video transmitter sending video out on a single frequency where, as we know, it's very possible to easily eavesdrop. So, yes, not what we would call NSA-grade security. But you really have to work in order to get a useful picture back out of this thing.

Leo: Every eight seconds.

Steve: Yeah. And then you're just getting a frame of data that you then have to go to a lot of work to reconstruct.

Leo: Right.

Steve: My feeling is, if you can't use the manufacturer's own monitor to receive it, then you're pretty much out of luck.

Leo: Yeah, that's sufficient.

Steve: And that's what our prior questioner verified.

Leo: Right. Bill Bolton in Australia raises a great point about IPv6 modem routers: Steve, if the world is going to be forced to move to IPv6 by year's end, why are almost no IPv6-capable consumer modem routers available as yet? He's talking about, when you get a DSL line, often case they'll give you the DSL modem with a built-in router since they know you're going to hook up a router anyway. There must be well over a hundred different models of various makes on the market. I guess it's true also of cable modems, but...

Steve: Yeah, just, exactly. And just other NAT routers.

Leo: Yeah. Any time you're getting online, it's not unusual to combine the two. Only three of the hundred different makes on the market have the necessary features. It seems more than a little strange to me, with one group crying we're running out of IP addresses, that the manufacturers are saying, "Huh? What?"

Steve: Yeah, I completely agree. Here's another example of it's not going to happen until it has to happen. If you look at NAT routers, they're still IPv4. I've not yet seen one that is supporting IPv6 features. It's just like no one's actually doing it yet. Yet we're running out of space. So it's going to be a really interesting year, Leo.

Leo: Yeah. We talked to - at CES we had Bob Frankston on. I know you know that name. Bob...

Steve: Oh, cool, yeah.

Leo: ...yeah, along with Dan Bricklin wrote VisiCalc. Actually Bob did the coding many years ago. Probably, if you were to say there was one application that put personal computing on the map, it was that. Because initially the Apple II was a toy. And as soon as you put VisiCalc on it, the first spreadsheet program - nobody'd written a spreadsheet. Nobody even thought of a digital spreadsheet. Soon as you did that, it was like, business was now, okay, now I'm interested. Now I might want to buy one of those things. Anyway, Bob, after his stint at VisiCalc, went to work for Microsoft as probably a high-level fellow.

Steve: Thinker.

Leo: Yeah, thinker. And this, he said, in these days the ISPs wanted to charge you for each user in the house. And I do remember those days where basically each one would have a static IP address, and you would have to pay the full freight, or maybe a slightly discounted rate. He said, we were starting to create routers at the time. Or it was his suggestion to put NAT in. And he said, I knew, but I didn't tell anyone, that this would effectively make it impossible for ISPs to charge per user. They'd have to charge per household because of course the NAT would hide all the additional users. He said, I never mentioned that feature. I just said it'd be a good thing to put NAT in these routers.

Steve: And that's what connection sharing is, of course.

Leo: Right. And so he says, you can thank me for the fact that you are paying what you're paying for your Internet access. He had some other very interesting things to say. We decided to invite him in for a full hour interview for our triangulation show because he's a fascinating guy. And what I like about him, yeah, he created history 40 years ago with VisiCalc, but he's not sat on his laurels. And he's doing something very interesting right now, really about taking back the Internet, that I thought was fascinating. He said the Net Neutrality conversation is misguided. That's not what we need. He said there's a way to handle this.

Steve: The other thing, too, is that by putting NAT in all these routers, not only were we preventing ISPs from charging per user, but we did hugely slow down the depletion of the IPv4 address space. Because, you know, we've got, I mean, I'm sure probably all of our listeners have many different machines behind their single IP that's out there, their public IP. And we always are talking about 192.168.x.x addresses, which we're all sharing, but which are kept separate.

Leo: Yeah. Very good point. Thank you, Bob. Bob Frankston. Aloke Prasad in Ohio has a question for you. He notes that Microsoft disagrees with you about swap files on SSDs. Okay, this I've got to see. You said it was unwise to use an SSD for the Windows swap file. You're not alone, by the way. We mentioned Allyn Malventano. You talked about your friend, our friend...

Steve: Mark Thompson.

Leo: ...Mark Thompson. The following article from Microsoft says otherwise. It's blogs.msdn.com. It's a May 2009 article, "Support Q&A for Solid-State Drives."

"Should the pagefile be placed on SSDs? Yes. Most pagefile operations are small random reads or larger sequential writes, both of which are types of operations that SSDs handle well. In looking at telemetry data from thousands of traces and focusing on pagefile reads and writes, we find that, one, pagefile.sys reads outnumber pagefile.sys writes by about 40 to 1." Well, that's good to know. That's interesting. So in other words, there's a lot more reading going on than writing, 40 times more.

"Two, pagefile.sys read sizes are typically quite small, with 67 percent less than or equal to 4KB, and 88 percent less than 16KB. Three, pagefile.sys writes are relatively large, with 62 percent greater than or equal to 128KB and 45 percent being exactly a megabyte in size." This is Windows, of course, only we're talking about. This is how Windows behaves. "In fact, given typical pagefile reference patterns and the favorable performance characteristics SSDs have on those patterns" - in other words, SSDs are faster with reads, they're really great with lots of small reads because the seek time is zero - "there are few files better than the pagefile to place on an SSD." Well, that kind of makes sense. The issue really more is this thrashing of the SSD. But if the files are megabyte most of the time, does that ameliorate that?

Steve: Well, this is a perfect example of a person answering a question from their perspective, but not a different perspective. That is, if all you were asking was about performance, then I completely agree. But my focus has never been on performance in this discussion. It's been on burning the things out, which this doesn't address at all.

Leo: So all of those extra writes, regardless of the size of the writes, are not good.

Steve: Correct.

Leo: Reads we don't care about on an SSD. Lots of reading we don't care about. It's the writing we care about.

Steve: Correct, because writing is a physically fatiguing process for an SSD.

Leo: You can say that again. As the author of 13 books - no, I'm just kidding. Never do it again.

Steve: And Mark Thompson and I have discussed this at length. He's performed the experiment of using an SSD for a swap file and watching it burn out the SSD. I mean, in a relatively short time it just killed it. And so, anyway, so my advice stands, which is, if you're using an SSD, hopefully before you have gone to the expense of using an SSD, which is still much more expensive than a hard drive, you will have invested money in as much RAM as your system can handle because RAM is much less expensive, and you'll get much more, you'll get huge benefit from going to the most RAM you can possible get. And if you've done that, then turn off pagefiles. And if the only drive you have is an SSD, I stand by my advice.

I agree that, from a performance standpoint, the SSD is a perfect device for containing the pagefile. Unfortunately, Microsoft thrashes their pagefile. I mean, they're writing to it a lot. Yes, 40 times less than they're reading, but it's something that's going on all the time, pretty much. I mean, we've all seen, we've watched the hard drive light flickering there, like when nothing is going on. It's like, what is it doing? Well, who knows. But we know that it's writing to the pagefile, which it does a lot. So anyway, I think it's a perfect example of two different people with very different aspects of the problem that they're addressing. I'm looking at long-term life. Microsoft's looking at performance.

Leo: That's really great. What a great illustration of that. Depending on your point of view. I love that, yeah. So we stand by our suggestion not to use the SSD for the pagefile unless you don't mind buying SSDs regularly.

Steve: Yeah. Not so much.

Leo: It will be faster, though. It will speed it up. Could you put a pagefile in a RAM disk? I guess you could. That would be a good thing to do.

Steve: Well, it would make much more sense to leave that RAM disk free for RAM.

Leo: Well, yeah, right, of course. The more RAM you have, the less you need a pagefile. Pagefile is all about what happens when you run out of RAM.

Steve: But if you had a RAM-based hard drive, that is, if you had a RAM-based physical drive that wasn't part of main memory and couldn't be part of main memory, then, so that it's like a separate paging device, then absolutely. That would make a lot of sense.

Leo: Our last question, Steven, from Jim Sanders in Irvine, California. He wonders about iPod and iPad solid-state hard drives. Because most iPods, only one, the Classic, has a moving drive. And all the iPads are solid-state. In fact, the MacBook Airs are also solid-state. Steve, you talked about the finite number of write cycles on solid-state hard drives, which I presume includes the array of portable devices like the solid-state iPod and the iPads. Given that, should we be thinking about minimizing the number of times we sync the devices? Hmm. Does syncing with the desktop shorten the lifespan of the SS hard drive? Long-time SpinRite owner. It's saved two machines for me.

Steve: One of the things that I thought I should do, I liked this question because I think maybe I've concerned people unnecessarily. The least robust technology for SSDs is called the MLC, the Multi-Level Cell, as opposed to the SLC, the Single-Level Cell, which is a much, much more expensive drive, but also more robust. But even the multi-level cell, the lesser of the two technologies in terms of robustness, has a guaranteed minimum number of write cycles of about 10,000. Now, I just divided 10,000 by 365, which is roughly the number of days in a year. And I get 27.397, which is to say that, if you rewrote the entire drive daily, that drive would last for a minimum of 27.397 years. So, yes. SSDs have a limited life. But so does the universe.

Leo: [Laughing] In a nutshell. You know, what can you do.

Steve: So don't worry about it.

Leo: Okay. That's good, that's good, I like it. Steve, you're always - I love this show. And I actually love the Q&As because of the wide range of topics. I know it's fun to drill deep into a subject, as we did last week with Bluetooth hacking. But it's also fun just to cover a wide range of topics. I always learn so much on this show, and I thank you for it. Do you know yet what we're going to do next week?

Steve: We're going to talk about fuzzing, fuzzy-wuzzy, about browser fuzzing. It's the work that was done recently by the Google security researcher who, using fuzzing, found interesting and in many cases significant problems, more than a hundred, with all of the browsers we're currently using. So we're going to talk about fuzzing...

Leo: Fascinating.

Steve: ...a different approach to finding security vulnerabilities.

Leo: I love it. Next week on Security Now!. Meanwhile, if you want 16KB versions of this show for the bandwidth impaired, if you want full transcripts and show notes, you can go to GRC.com, that's Steve's site. While you're there don't forget to take a look at SpinRite, the world's finest hard drive maintenance and recovery utility for - not for SSDs, for spinning drives. You're going to have to come up with something for SSDs, Steve. I hope you're thinking about that.

Steve: Yeah, I'm thinking about it, actually.

Leo: Yeah. That should be interesting, an interesting challenge. Also, lots of free stuff - ShieldsUP!, DCOMbobulator, Don't Shoot The Messenger, just a ton of - actually Shoot The Messenger. Kill it dead. And tons of great stuff that Steve just does out of the goodness of his heart and because he loves to write software in assembly code. You can find that all at GRC.com. You can also watch this show, we do it live every Wednesday at 11:00 a.m. Pacific, 2:00 p.m. Eastern time, at live.twit.tv. Or subscribe. You really ought to. In fact, you ought to go back and listen to all 284 episodes because it's a graduate degree in computers and how they work. That's at TWiT.tv/sn for Security Now!. Steve, have a great week.

Steve: Will do, Leo. Thanks very much.

Leo: See you next week, right here.