SECURITY NOW!

Transcript of Episode #480

Listener Feedback #200

Description: Leo and I discuss the week's major security events and discuss questions and comments from listeners of previous episodes. We 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-480.mp3
Quarter size (16 kbps) mp3 audio file URL: http://media.GRC.com/sn/sn-480-lq.mp3

SHOW TEASE: It's time for Security Now!. Steve Gibson is here. We'll review the latest security news and then get down to work with eight questions from our audience. Steve's answers, your questions, coming up next on Security Now!.

Leo Laporte: This is Security Now! with Steve Gibson, Episode 480, recorded November 4th, 2014: Your questions, Steve's answers, #200.

It's time for Security Now!, the show where we talk about security. Now.

Steve Gibson: Yeah, yeah.

Leo: Yeah. That's Steve Gibson there, that guy there. That's the guy in charge there of the show. He is at GRC.com, Gibson Research Corporation. He's created - which sounds like a big giant conglomerate.

Steve: Conglomerate, yeah.

Leo: It's really just Steve.

Steve: Yeah.

Leo: A few other people. Creates the world's best hard drive maintenance and recovery utility, however. You might know it as SpinRite.

Steve: SpinRite.

Leo: SpinRite. Lots of free stuff there. And every week for the last almost 10 years now we have been getting together on various days, Tuesdays these days, to talk about security.

Steve: Yeah. And we did a Q&A last week, but we've been skipping Q&As because there's just been so many catastrophes in the security world. And so I thought, let's pull some more questions. Also there was a bunch of news I wanted to talk about. And having a heavy news podcast sort of squeezes out time from getting into any serious propellerhead stuff. And I just thought, yeah, let's do another Q&A. So...

Leo: I love the Q&A. And I know our audience does. So that's fine.

Steve: Yeah. And I like it because it gives our listeners a chance to have their voices heard. Like the guy whose question we took last week about random MAC addresses, we misunderstood what he meant. And so sort of a chance to come back and fix that. So I wanted to talk about a couple things. First of all, I have to say, I just love your banter with Sarah. Sarah, I don't know what it is. It's you're so comfortable with each other.

Leo: We've known each other for even longer than I've known you.

Steve: It really is special. It is just, you know, she is at the top of her game. She is quick and sharp and easy on the eyes. And actually it was the one before last where you were doing the Christmas music, but just crazy stuff. And even yesterday I just - I really look forward to iPad Today for the pre-show stuff. And in fact it's interesting because later, I guess a couple days ago, you capturing the whole stream because it was the pre-show stuff with you and Sarah that was just - it was wonderful.

Leo: Now, this wasn't on a podcast. That was you were watching live. Or you were watching a rerun.

Steve: I was watching live. No, but someone reran that.

Leo: Oh, yeah. We put that in reruns. We try to put that in reruns.

Steve: Oh, and it is just wonderful.

Leo: Aw, Steve, thank you. I appreciate it.

Steve: It just really shows how comfortable you guys are with each other. It's like you are different in a way because you're just so...

Leo: Well, she's so cute, I'm kind of giddy.

Steve: She is. And you're super comfortable together. Anyway, it's just - it's a great side. So...

Leo: I've always had such a crush on Sarah. I'm just mad for her.

Steve: And I was - well, who wouldn't?

Leo: And so I guess that shows; right?

Steve: I mean, she's got it going on. But...

Leo: Yeah, we're such good friends, and we've known each other for so long. And you know what, we never ruined our relationship by being in a relationship. So I think that was a good thing, too. Yeah, at this point now it's a very easy friendship. So thank you. I appreciate that. Because gosh knows I love doing that show with her. I really do. It's so much fun.

Steve: Well, it's just - it's a romp.

Leo: A romp.

Steve: And sadly, there's no link...

Leo: A madcap romp.

Steve: Sadly, there's no link that I know of where anyone could listen to that pre-show banter from week before last. It just popped onto your feed again. And I was so glad to see it because, frankly, once you started, it was like sort of more, okay, down to business now. And it's that pre-show stuff that, you know, and you and I often have that, which is why, as I said, this time I was biting my tongue about a couple things I wanted to talk to you about to make sure it got into the podcast rather than just stuff we were rambling on about beforehand.

Leo: I'm a little torn because one of the things - this all has - this is a complicated story, and I don't want to waste your time. But it has to do with the evolution of TWiT because, as you know, it was originally just audio podcasts. And all you could ever listen to was the produced final product because we didn't...

Steve: Right.

Leo: But then once we started putting up cameras, and it originally was just simple spy cams, you could watch more than just the show. You can watch us produce the show. And I think that's really what TWiT Live, the live version, really always is and has been is you're not - we're not - and I think sometimes people don't understand this. It's not like - we're not trying to do a TV station that, you know, show show show show. You're watching us produce programs for later download. But then I also feel like the live thing is good. And we want people to watch live if we can get them to, even though we have yet to figure out a good way to count live viewers for advertising purposes. So from a financial point of view it's a kooky thing. But I just think it's important because it's the easiest way to watch.

Steve: Yes, the stream has so much extra stuff.

Leo: Right. And that's intentional.

Steve: So we're recording now. Everybody is hearing this. So for what it's worth, you know, and I just - when I'm coding, and I'm not doing, like, I'm not writing because I can't have, as I've talked about, for me the distinction between a linguistic process and a nonlinguistic process, I'll just turn on to see what's going on. And I get all this extra stuff from live.twit.tv which is not in the podcasts. And a lot of it's just really worthwhile.

Leo: Thank you. I mean, that's kind of the design. I don't know why, but I just feel like live is exciting and interesting and more fun. And so we encourage people to watch live. And of course live is the only way you can interact with the shows; right? If you're not watching live, you can't exactly interact with us.

Steve: Right, right. And I just think it's organic. I mean, that's - it's you. It's reality. It's what's going on there. And while you do have regimented, scheduled things, because you have to have that or everyone would just sit around playing...

Leo: Kind of sort of regimented.

Steve: ...with spaghetti all day.

Leo: But there is this concept called "Leo time."

Steve: Yeah, I just - it just - I just think it's great. And so I would encourage people, if they have the opportunity, to just watch the feed sometimes. Sort of consider that there's more than these snipped out, edited, tuned, produced, I mean, and I'm as much into the, okay, here's all my show notes, here's all the Q&A, I mean, there's a whole bunch of production that goes into Security Now! every week, which is one of the reasons I think that, for those people who want that, we have that. But watching what goes on, the antics before and after, that's different than the podcasts, but I really enjoy it. So I just wanted to make sure people knew there was something more available, and just to sort of turn the stream on sometime and see what's going on because...

Leo: Thank you, Steve. I'm very flattered, thank you.

Steve: ...there's often really good stuff.

Leo: We do our best.

Steve: So I keep listening to you talking about the Apple Watch. And I just wanted to chime in, off topic as it is. But, you know, we're techies and so forth. I completely think that all of this stuff is way early. And the analogy I draw, because there is one that you and I have lived through, is the laptop. The laptop took a long time to actually get practical, to the point where it's in many cases now people's only computer, or it's their preferred platform. But, I mean, we went through three-hour battery life.

Leo: That's right, I forgot, yeah.

Steve: And, you know, laptops with wheels, and laptops with really dim screens that were 80 characters by 16 lines. I mean, and a floppy drive where it was like, well, how do I get my data into there? Oh, you put it on a floppy and then you stick it in this little slot. It's like, no, no, no, no. And so, I mean, and we endured this, we early adopters, for years. I was a Toshiba loyalist, and so I sort of followed Toshiba through. And, I mean, with lifetime measured in minutes, battery life in minutes rather than in days, and really bad screens, and no peripherals or I/O, small storage, I mean, they were really poor cousins of what we had on the desktop. And it took, it really took the evolution of technology, which we went through, to enable the machines we have today.

And so I completely agree. I look at these watches, and I just - they have no interest for me at all. I wear a wristwatch. I'm a wristwatch wearer. And occasionally I glance at it to see what time it is. And I don't have to shake it, or talk to it, or push any buttons, or beg it or pray or charge or do anything. It goes for years on some battery that I go back to the jewelers, and they change it because it's got a mechanical movement in it, and so I don't want to get any, you know, they're able to do it in a dust-free environment. But, I mean, it is abso- well, in fact I did have to deal with it on Sunday because I had to change it by an hour. And it's polite enough that I can just twist the crown, and the hour hand jumps by hour increments. So even going, you know, daylight savings times plus and minus isn't increasing my chance of having a heart attack on Monday or Tuesday following...

Leo: You heard us talk about that.

Steve: Uh-huh.

Leo: Ten percent higher chance of a heart attack.

Steve: Amazing.

Leo: I know.

Steve: Amazing. So anyway, I wanted to interject my two cents' worth in about all of this watch business. Which, again, I don't think it's bad, but I'm not going to get any of those arrows in my back. I'm going to wait because maybe someday it'll evolve to a point where we have a watch that just, I mean, it uses so little power that it's got a solar strap which just absorbs light from the ambient and keeps itself going, and then it updates an eInk display so you don't have to mess with it. I mean, who knows. But I just, you know, it's just fun to see where we start versus where we are. I mean, I have a refrigerator full of Tungsten Palms that I thought, oh, this is the one. I don't want to ever...

Leo: I do mention that a bit, and I hope you don't mind that I use you as an example.

Steve: No. Yeah, you know, sometimes you get it right. And you guys were talking about on MacBreak Weekly the absolute ultimate calculator app in the world. That PCalc...

Leo: Oh, you like that, huh?

Steve: Oh, my god. It is - and, oh, on the iPhone 6 it is a joy. It is my go-to calculator. And so I was watching the drama of, not that I use it in widget mode, it just sort of never occurs to me, but I was curious to see that it was there. And I have seen it in widget mode. But it was fun to get Rene's extra rich information into the background of that because I care about the app and the author and its success. And in fact I've edited mine. You just - you hold a button down, and it's just very much like shifting the icons around on your iOS device. You can then drag and you can move the buttons around. And I'm using exponentiation because I'm doing scientific stuff enough that I've reordered, I've sort of changed the design a little bit. And anyway, it is a wonderful application.

Leo: There's an example of something you already have, you don't really need, and yet if it adds that much value - do you use it, I bet you use it in RPN mode, don't you.

Steve: Absolutely.

Leo: Oh, god. You're such a geek.

Steve: What would you do with an equal sign? Yeah. And, I mean, here are my two RPNs...

Leo: Oh, look, you've got your HPs, yeah.

Steve: Those are my two RPN HPs. A 16, which does hex, decimal, and binary and so forth. And then the scientific one. And these are not in the refrigerator, but I do have quite an inventory of them because...

Leo: Those are the 12Cs, you said?

Steve: We've got 16C, of course, 16 is the programmer's calculator. And it's got extra things like masking and rotation and all that, setting bit and clearing bits, and they're all programmable. And then the 15C is the scientific. And it's just, you know, what you want, you want a device where, as we've talked about often in different contexts, where much like the telephone, the user interface fades, and you're not conscious of it. It's not in the way. It just helps you get your job done. When you're on the phone, you're not thinking about holding a handset and talking into a microphone. You reach through it into the mind of the person you're speaking with. And so the UI becomes transparent. And for me, these calculators, once you get comfortable with them, that's the way they are. And PCalc on iOS, oh, I mean, it's a reason to have a phone is just to have...

Leo: Wow.

Steve: ...for people who calculate.

Leo: So it's like an HP 16C on your phone, basically.

Steve: Yes.

Leo: Ah.

Steve: With a bunch of really nice extra features. You can drag the display to the left and go to previous displays. You can pull it down in order to shrink the keyboard and create more space for the screen. And, oh, it's skinnable. And he's got, like, I don't know how many dozens of skins. And I've just chosen one that I think is best. But people have, you know, opinions vary. I think there's now probably a Windows 10 or 8 or whatever it is, the new flat theme skin. But anyway, I just really, really like that.

Leo: I'm going to install it right now on your recommendation.

Steve: Oh, Leo, it's just, I mean, if you're not someone who calculates, then obviously...

Leo: I'm not. But I want a good calculator.

Steve: It is THE calculator. Really, it's just nailed it.

Leo: Last time I used a calculator. Well, occasionally I'll use it in the radio show, mostly for bandwidth calculations. I wish there were like a bandwidth calculator. But the last time I can remember that I wanted a calculator on my phone, I was visiting a friend in court. He's a litigator. And it was a recess, and he had to quickly figure out what the damages he should ask for should be. And he's, "Quick, give me your phone." And he calculated it on - I thought, okay, I would think you'd think about that a head of time, but - he won the case, by the way.

Steve: Well, and the cool thing is it's not like you have to carry a second calculator. Or, that is, you have a phone and a calculator. Back in the day we used to carry, we geeks, a calculator because we didn't have phones that could do double-duty. But here you always have a really top-end calculator with you, whether you're carrying your pad or your iPhone. And I'm sure there are good calculators over on the Android platform.

Leo: Oh, yeah. You know what, actually I use Google a lot because you can do calculations with Google Search. And you can actually do fairly complex calculations with Google Search.

Steve: I use it for monetary conversion and for units conversion sometimes. You just sort of ask it the question, and it says, oh, here you go.

Leo: But you can do weird things like - in fact, I should try this with bandwidth, like how many hours would it take to download a 500MB file with a 3MB connection or something like that. And it's smart that way. It's kind of like Wolfram Alpha. That's where you go if you've really got some serious stuff. That's like having Mathematica at your fingertips.

Steve: Okay. So I have an important correction to make at the top of the show.

Leo: Uh-oh, yeah.

Steve: To a mistake I made last week. We're going to talk about CurrentC; an interesting new entry into Tempest Broadcasting. Tempest, of course, is the famous technology for extracting information from just the operation of a computer. A Mac OS X privilege elevation that was recently uncovered, but I don't think is a big deal.

TextSecure gets an audit. The EFF finishes a comprehensive evaluation of messaging app security. Another ruling on fingerprints and passwords and our rights. And a nice little piece about state-of-the-art TVs, and a Q&A. So a bunch of great stuff to talk about.

Okay, so Department of Corrections. In answering the question that one of our listeners asked when he was talking about same-origin policy, I got sort of sidetracked by the correctness of his supposition, which is no excuse because I didn't give him the correct answer. And I missed the point of Adam Langley's note on his page when he said that any web page is able to access any other domain. Well, of course it is. The same-origin policy, what it does is it prevents anything in one origin from accessing information in another origin, which is completely irrelevant to Poodle.

The point is it's not the script in the browser that needs cross-origin access because the attacker is outside. And so it's absolutely the case that script in a browser can cause queries to any other domain it wants. I mean, that's what a web page is. I often talk about how web pages are now composed of crap coming from 40 different servers, 40 different domains. So an external attacker is going to see the traffic from all of that, even if the page itself has no access to the content of those other domains. It doesn't need to. Poodle doesn't need it to because the attacker is on the outside.

And so it's like, this hit me during breakfast the following morning. It was Wednesday morning, and I was just, I don't know, I just was sitting there sipping coffee, I thought, oh, my lord. And I immediately tweeted, from where I was, I said, "Realized that I missed the point Adam Langley was making about Poodle. JavaScript cannot get internal access to other domains." Next tweet: "But JavaScript can cause the browser to make queries to other domains, and that's all we need since the attacker is outside to observe." And then the third tweet: "I'll correct my confusion at the top of next week's podcast. In the meantime, that point I made was wrong." So I wanted to...

Leo: Wow. Well, that's big of you.

Steve: Well, it's correct. So I don't like to make mistakes. Everyone knows I try hard not to. But when they happen, they happen. So the only thing we can do is fix them. So that's fixed.

Okay. CurrentC. And I know you've been talking about it. This happened Wednesday morning, immediately after last Tuesday's podcast, as so many things seem to. They happen the day after the podcast, so we have to go a whole week before we can talk about it. And it turns out that it was certainly a black eye. And in looking at it more closely, it's also sort of not as big a deal as it was easily sort of made to be. It turns out that it was apparently a security breach at the email provider that the Merchant Customer Exchange, MCX, uses.

Leo: Well, but...

Steve: I know. Certainly, I mean, I'm not making an excuse for them.

Leo: They gave it to a third party. So...

Steve: Yeah. So not secure.

Leo: Not secure.

Steve: Yeah.

Leo: And the third party wasn't secure. So it doesn't matter who it came from. If they lost it, they lost it.

Steve: So for people who don't know already, the day after last week's podcast where we were talking about this, CurrentC had what you'd have to consider a major breach because the email addresses of all of their demo users and early testers leaked from their email provider. Dekkers Davidson is the CEO of MCX. And so he tried to downplay this. But there were a couple interesting nuggets that were worth - and that actually I saw as positive, just in terms of our interaction, of the world's interaction with him, because he did confirm that there is an exclusivity relationship for retailers who use MCX. That is, they are not also, as part of their agreement, not also able to use Apple Pay today. But two things he also said: There are no fines associated with exiting the consortium.

Leo: Yeah, see, that's interesting. So you could drop out.

Steve: Yes.

Leo: But can you do MCX plus Apple Pay?

Steve: And he said that's not being ruled out in the future.

Leo: Yeah, bet it's not.

Steve: That's the second point, yes, yeah.

Leo: This is a backpedal because really the whole point of MCX was them both keeping your personal information to themselves, and Apple doesn't give them personal information, and disintermediating the credit card companies so they didn't have to pay the percent to them. And both things would happen with Apple Pay. So I think this is backpedaling.

Steve: Yeah, which is fine.

Leo: Yeah, backpedal away. Go, go, go.

Steve: Let's backpedal to turn that stuff on because we want to use Apple Pay.

Leo: Yeah.

Steve: However you do it, however long it takes. But the good news is that retailers can decide, if Apple Pay continues to happen, because as we know, what was it, more than a million credit card activations in the first three days. So Apple was out touting their initial launch success. And we'll just have to see how it goes. Clearly, if it succeeds, and if any retailer - Rite Aid, of course, who famously turned theirs off - if they realize people are going to Walgreens specifically because, you know, in protest, they're going to say, "Uh, sorry, MCX, we need to enable our customers to use what they want, and you didn't get your thing off the ground in time." So it's interesting because MCX actually began two years ago, back in 2012. And it's just sort of been limping along and didn't really happen. And, I mean, even now it's not happening until 2015, sometime next year. So it's like, they may have missed the boat. We'll see.

Okay. Air Hopper. Many people picked up on this news, which is interesting. This was the result of a group of researchers who presented a paper at what's called MALCON, which has only been around relatively briefly, at the IEEE 9th International Conference on Malicious and Unwanted Software. I thought that was nice. That's sort of a nice title: The International Conference on Malicious and Unwanted Software. So it's not necessarily evil, but it's unwanted. And so we're going to call it - that's under the umbrella.

Anyway, so what these guys demonstrated was not Tempest as much as sort of a toned-down version. Now, Leo, you've been around as long as I have. I remember putting an AM radio, sitting it on top of an IBM 1401 sort of small mainframe, it certainly wasn't a minicomputer, basically it could add, but it also wasn't very powerful, and playing Christmas music by feeding in punch cards...

Leo: You did this?

Steve: ...that caused, oh, yeah, that caused loops to run at...

Leo: [Making odd noises]

Steve: Oh, it was awful.

Leo: I can't believe you did this.

Steve: Oh, yeah.

Leo: You're hysterical.

Steve: And then years later I worked for a company called Minicomputer Technology, where not surprisingly we had minicomputers. And we did the same thing, just because, you know, the days were long back then. And again, you would tune a radio in between stations, and it would pick up all kinds of noise from the operation of this machine that it was sitting above because these machines all used core memory. And core used pulses of current through wires to flip the magnetization of ring magnets into clockwise and counterclockwise directions. So that required a substantial pulse of current.

Well, that pulse of current that generated a local magnetic field also broadcasted electromagnetically, powerfully. So much so, in fact, that in my PDP-8 mini computers that I have, they consumed one of their precious card slots with nothing but a copper grid between the core memory region that you sort of stick toward the back of the back plane, and all the processing and I/O, because the core memory was generating so much electromagnetic noise from its functioning that it could flip bits in the neighboring electronics if you didn't have essentially what was sort of a Faraday screen, an electromagnetic screen. And card slots were precious, but they gave one up just to have this barrier between the core memory and the rest of the computing electronics.

So the point is that, although - well, and everyone has looked at modern-day machines and noticed that, to varying degrees, these things are clearly noisy inside, and the manufacturers attempt to quiet them down, to keep those emanations from getting out by doing all kinds of things, you know, you'll have clearly conductive copper fingers that reach out and touch the lid as you close it in order to create an electrically conductive closed cage that prevents leakage. And people have seen those lumps on their cables, which are specifically there to absorb sharp edges in the transition of the current because edges - nature doesn't like edges, as we've talked about. And those broadcast unless you snub them by essentially running them through a big piece of iron that absorbs that transient.

So it turns out that today's computers are comparatively quiet, except that screens are not. Screens, by their nature, have to be sort of out there exposed. And they create an opportunity for broadcast. So what this team of researchers showed was, if you could - and this is why this is not like a super danger. Remember that the way Tempest worked, the idea was you could surreptitiously receive from a CRT enough emanation from it that you could decode the image it was displaying, that is, the technology of the CRT was such that, if you could aim like a parabolic dish from a distance through a window at this CRT, or even like the back of it, you could suck in all that data and process it and figure out what it was displaying. So modern LCDs don't operate in the same fashion, don't operate at the same voltages at all.

So what these guys showed - but my point was that you needed no preparation. Well, these guys showed, if you could prepare, what could you do? Meaning that, if you managed to infect - so you have to first infect the computer. And it then has malware - thus the MALCON Conference - it has malware running in it which is able to use the video subsystem to surreptitiously get an FM transmission out of the machine into a nearby recording FM radio. So you need a smartphone that is able to also receive FM, that is, as in broadcast FM, because that's about the frequency range they're able to put out. And that smartphone has to know where to be tuned. It's got to be tuned to the malware, essentially. It's got to be within one to seven meters, so it's not long range. And it's able to - it's not super high bandwidth. It's only able to get about 60 bytes per second transmitted through this channel.

So it was an interesting capabilities display. As we know, these things only ever get better. They typically don't get worse. So we could imagine successive refinement. In any kind of a proof-of-concept, these guys are just demonstrating that their first idea works. If you really needed to get down to it, it's very much like a modem. Remember that modems used to be happy to get 300 baud through a phone line. Now I can't believe the amount of data we get over copper.

Leo: Analog.

Steve: Basically dirty copper twisted, yeah, analog twisted pair. It's unbelievable. So one could imagine successive refinements of this concept, the idea of something, rather than just being a passive leaker of information, if you put something malicious in the computer that was able to select what it wanted to send and arrange to use the existing channels to do so, it probably could. On the other hand, this is one of those things where there's probably easier ways. Oh, and the point was this was called AirHopper because this would be for air-gapped computers. That was the point they were making was, yes, if it was on the Internet, or any 'Net, then thank you very much for the connection to the world. I'm not going to be limited to 60 bytes per second or seven-meter range. I can talk to Moscow directly.

So, but in this case, if the thing is off by itself in a basement and not connected, and you think you're being really clever, well, if something bad got in there, then it turns out these guys demonstrated you could leak information at a low rate and in close proximity, just through the video channel, just video noise that you're able to control. So a really interesting sort of a proof-of-concept.

There was concern about a Mac OS X privilege elevation exploit. I saw a couple people refer to it as zero-day. It's like, well, okay, except that it's a local privilege elevation. Apple knows about it. They'll be patching it. And what it allows is it allows a bad guy who knows what to do - and this was not revealed. It was demonstrated at a recent conference. But the person who discovered it did not reveal what it was. He's waiting until after it's fixed. Then he'll tell us what it was. So he's being responsible. So in that sense it's not a zero-day because it's not in the wild. Nobody, as far as we know, has figured out how to do this.

What it allows you to do is to enable an admin account, which is the normal default user account on a Mac, to obtain root privilege which you normally don't have access to. You normally need extra magic in order to get that. And so the workaround, I mean, if you were really worried about this, and I don't know that anyone needs to be, but again, it's a function of the environment of your Mac, if you're the only one using your Mac by yourself in your house, then I think you're probably fine. This is not a remote network takeover sort of thing. But if you had a super high-value Mac in an environment where many people had access to it, and you're waiting until January, which is the earliest that we expect Apple to be able to address this, then maybe you want to go to some extra measures.

It turns out you can create another standard admin account, and then you can reach over back into the original one and remove admin privileges from it and then make that the normal default user account so you've neutered that. And that's a workable, temporary measure, if you consider this a problem, until Apple fixes it. But so consider all of that. It may not be even worth bothering about because it's not in the wild. As far as we know, nobody else knows how to do it. You do need physical access to the machine. And we already know that pretty much any time we have physical access, the jig is up anyway.

Leo: But you don't have to know the admin password to do this.

Steve: Correct. Basically it's an admin password bypass.

Leo: Right. We wouldn't want that.

Steve: TextSecure, that is Moxie Marlinspike's creation. And it has an interesting past. As I recall, didn't it get purchased by Twitter, and then they open-sourced it and released it, and now it's been audited. And so the news here is...

Leo: Oh, good. Oh, good.

Steve: Yes, that it and Cryptocat are the two open-source solutions which have been audited and complete their across-the-board green stars. The EFF just published a secure messaging scorecard. And in fact the first page of the podcast notes this week I have a sort of a small screen cap of that. But TextSecure passed the audit with one - these guys really dug deep. And it is a very complex protocol. I mean, it is, like, whoa, eye-crossingly complex. But they found something that they named the "unknown key share attack" and found mitigations for it, shared it with the developers of TextSecure, who are already on fixing it.

So just a snippet from the paper's abstract. They wrote: "In this paper, we present the first complete description of TextSecure's complex cryptographic protocol and are the first to provide a thorough security analysis of TextSecure. Among other findings, we present an unknown key-share attack on the protocol, along with a mitigation strategy, which has been acknowledged by TextSecure's developers. Furthermore, we formally prove that, if our mitigation is applied, TextSecure's push messaging can indeed achieve the goals of authenticity and confidentiality."

So those are the two things you want. TextSecure succeeds with a formal, mathematical cryptographic-strength proof. So I'm sure TextSecure will shortly get revved to close this one opportunity for the so-called "unknown key-share attack," which I didn't even look at because it's going to be gone before we can get to it. And so we've got a really useful proof of security for TextSecure.

Leo: I'm installing it right now. How about RedPhone? That's also from the same company, from Moxie Marlinspike.

Steve: Yes.

Leo: That's their secure phone solution.

Steve: Yup, also very good. So I mentioned EFF's scorecard. You may want to click the link and just bring this up on the screen while I'm talking.

Leo: I love this thing, by the way. They just put this out; right?

Steve: Yes. Yes, just now. So they analyzed basically all of the instant messaging solutions under seven criteria: whether the message is encrypted in transit; whether it's encrypted from its provider; whether the contact identity, that is, the other end is verifiable; does it offer forward secrecy, which we know means if the keys are compromised in the future, are messages in the past vulnerable; is the implementation code open for independent review; does it have a proper security architecture; and has it been independently audited. Under all of those criteria, for all of the popular instant messaging solutions, only two are green for yes, get checkmarks as opposed to slashes for all seven. And that's, as I mentioned before, Cryptocat and TextSecure.

Leo: They do say RedPhone is also all seven, as is Silent Phone and Silent Text.

Steve: Yes.

Leo: So that's good. So we can use RedPhone, too. You know, I thought RedPhone's kind of silly because you have to have RedPhone on both ends. But what will happen is you make a regular call using RedPhone, and the other end will still work, but they'll get a popup that says, if you'd like to have a secure phone call with this person, install RedPhone.

Steve: Nice.

Leo: I love that.

Steve: Nice.

Leo: So that means it kind of works across the board, gives you this as a straightforward option. I'm going to install both of these. That's great.

Steve: Yup, yup. Those are the ones. There's also a ProPublica.org has a sort of a different - it's a little more interactive. You're able to bring up the columns and then sort the columns by qualification, like to bring all the ones that have been audited up to the top, or all of the ones that provide good authentication up to the top. So you're able to - it's the expanded chart that you found there at the EFF site.

Leo: Right, right, that's nice.

Steve: So, very nice.

Leo: Thank you, EFF.

Steve: Also in good news, we have one more judge. We're still sort of feeling our way through this question of fingerprint versus password. And a Virginia Circuit Court judge ruled last Thursday the way we wanted them to, which is that a person does not need to provide a passcode to unlock their phone for the police. We've pretty much already lost the battle of the fingerprint. That seems to be pretty much gone. But at least the issue of what you know cannot be extracted. That's still considered testimony, which is protected under the Fifth Amendment to the - in the U.S., I should say, to the U.S. Constitution.

So a couple quote snippets here: "Giving police a fingerprint" - and this is from the judge's comments. "Giving police a fingerprint is akin to providing a DNA or handwriting sample or an actual key, all which the law permits. A passcode, though, requires the defendant to divulge knowledge, which the law protects." And then, lastly: "A communication is testimonial only when it reveals the contents of your mind. We cannot invoke the privilege against self-incrimination to prevent the government from collecting biometrics like fingerprints, DNA samples, or voice exemplars because the courts have decided that this evidence doesn't reveal anything you know. It's not therefore testimonial."

So the takeaway is, convenient as Touch ID is, if you're ever in a situation where you may be losing access to your phone, or you want control over access, then you want to turn it - you want to do a hard turnoff of the phone. Or remember that Touch ID resets after 48 hours of non-use.

Leo: Right, right, right.

Steve: And it's typically trivial to get your attorney to create a two-hour delay. That's only two hours.

Leo: Two days.

Steve: Two days. A two-day delay. So just file some paperwork or whatever. And that's like, oh, wait a minute, you know, we require an extension or whatever. And 48 hours goes by, and then even your fingerprint won't unlock. And of course, if you wanted to be - if you were in an environment where you thought it was worthwhile, then don't register your thumb, register a more obscure finger because, after four mistakes, it locks and then requires you to enter your passcode, which, boy, you know, the court just cannot compel you to divulge. So, I mean, that's not a concern I have, but I know that there are listeners who really like to keep a moat around their belt and suspenders.

So this is where we stand at the moment. And again, I think the more affirmation we have of the fact that you cannot be compelled to reveal a password, the better, because it is testimony against. Oh, and it was interesting, too, because in this case - it was weird. I remember as I was reading the case, it was somebody - there was something with a girlfriend. I think he assaulted his girlfriend. And the police wondered if maybe there might be a video of that on his phone. And it's like, what? Who's going to videotape themselves doing that in the first place? And I know that there is some issue in the law of how much, like, a fishing expedition. If they're like, well, we'd like to go look at the phone, it's like, well, of course you would. But, sorry, unless you have some reason...

Leo: You need probable cause. You have to have probable cause.

Steve: Yes, exactly, probable cause.

Leo: And, you know, there's a case, a big case right now in California, as I'm sure you know, CHP officers have been taking people's, attractive young women's phones and looking for nude pictures and then sending them around.

Steve: What?

Leo: Oh, you didn't see that.

Steve: No. I must be watching the wrong channels.

Leo: Well, it's...

Steve: Wow.

Leo: But, I mean, the point being...

Steve: Why would they do that? That's just so...

Leo: It's so wrong.

Steve: Yeah.

Leo: But that's the point is that, while there are constitutional protections and legal protections, when you're in the hands of law enforcement, sometimes it's scary and threatening. And if a police officer says, "May I look at your phone," people often, despite their legal protection, might say yes.

Steve: They're intimidated, yeah.

Leo: They're intimidated.

Steve: They don't realize they're not going to get themselves in more trouble by saying no. And so, yeah. Wow.

Leo: So, yeah. I mean, I think that's shameful and really, by the way, he was charged and fired, but...

Steve: Good. Okay. So I just liked this piece. This is a little fluffy, but this was Michael Prince, who contributed this short observation to Salon. He's counsel in the Liberty and National Security Program at the Brennan Center for Justice at the NYU School of Law. So he understands what's going on. And so what he posted, just the beginning of his piece in Salon, he said: "I just bought" - oh, and the reason I brought this up is many people tweeted this to me because they were like, oh, my lord. It's like, well, good observation. He says: "I just bought a new TV. The old one had a good run, but after the volume got stuck on 63, I decided it was time to replace it. I'm now the owner of a new 'smart' TV..."

Leo: Ooh.

Steve: Yeah, "...which promises to deliver streaming multimedia content, games, apps, social media, and Internet browsing. Oh, and some TV also. The only problem is that I'm now afraid to turn it on." Actually, you may not even need to turn it on. "You would be, too, if you read through the 46-page privacy policy. The amount of data this thing collects is staggering. It logs where, when, how, and for how long you use the TV. It sets tracking cookies and beacons designed to detect" - quote from the 46 pages - "'when you have viewed particular content or read a particular email message.' It records, quote, 'the apps you use, the websites you visit, and how you interact with content,' unquote. It ignores do-not-track requests as a considered matter of policy.

"It also has a built-in camera with facial recognition. The purpose is to provide gesture control for the TV and enable you to log into a personalized account using your face. On the upside, the images are saved on the TV instead of uploaded to a corporate server. On the downside, the Internet connection makes the whole TV vulnerable to hackers who have demonstrated the ability to take complete control of the machine.

"More troubling is the microphone. The TV boasts a voice recognition feature that allows viewers to control the screen with voice commands. But the service comes with a rather ominous warning: 'Please be aware that if your spoken words include personal or other sensitive information, that information will be among the data captured and transmitted to a third party.' Got that? Don't say personal or sensitive stuff in front of your TV. You may not be watching, but the telescreen is listening." So it's like, oh, boy. Maybe it is time for us to read the privacy policies. I'm just so happy that I have a stupid, 13-year-old car. This thing, it's 2001, and I like it because it does...

Leo: TV.

Steve: No, car.

Leo: Car.

Steve: I'm driving a 2001 car and very happy because it's got low mileage, 100,000 miles, and I take good care of it, it's still running just beautifully, and I don't want any brains in my car. I just want it to take me somewhere.

Leo: My TV does all of that. I couldn't - who cares.

Steve: I know.

Leo: Nobody's listening.

Steve: I know. Until they are.

Leo: Well, maybe they are. Who knows.

Steve: We have a listener of ours asked me to mention, he called himself, well, his Twitter handle is @therealjampers, J-A-M-P-E-R-S. I put the link in the show notes for anyone who's interested: github.com/therealjampers/spritzjs. He's done a very nice sort of reference JavaScript implementation of the new Spritz cipher that we talked about. This is the recently released updated version of RC4, which really begs for simple implementations because - yup, there it is on the screen. And very nice.

If you scroll down and click on the Spritz.js link, right there at the top, near the top in that little link zone, you can see his implementation, which is very straightforward. He's done some neat things, like he kept the variable names in his source identical to those in the paper. So it sort of represents a live running implementation of the cipher that follows the description. So if anyone was interested in messing around with that and perhaps contributing, he'd love to have, for example, sample test vectors for the cipher and so forth. By all means, go over to that on GitHub.

Leo: He credits you with the inspiration, along with Bruce Schneier.

Steve: Oh, cool.

Leo: You're in the source code, man.

Steve: And, well, he is a listener of the podcast. And I tweeted two links to a neat-looking sci-fi trailer. One of them was taken down due to copyright constraints by its publisher, so only one has survived as of this morning when I put the links - I was pulling the links over and checking them. But anyway, it's due out not too far in the future, April 2015. Out of habit, I tweeted 2014 and had to correct that. It's called "Ex Machina," E-X space M-A-C-H-I-N-A. So you could google that. It's on IMDB, and it's on YouTube. And, ooh, it looks like some tasty sci-fi.

So apparently it's a - you get from the trailer that some young jock has acquired or taken over a major technical corporation, and he's looking through the books, and he sees that there's some project, sort of off the books, run by some genius who's been disappeared. And so he takes a helicopter flight over the river and through the woods out into this way remote, one-person-occupied R&D facility and discovers what this person has been working on. So it looks really, really tasty. So I'll just give everyone a heads-up, anyone who likes sci-fi, "Ex Machina." Go find it on YouTube. You'll be glad you clicked the link. And then you'll be waiting five months impatiently for it.

And it's been quite a while since I shared one of our traditional SpinRite testimonials with our listeners. I've been talking about the technology and various, sort of conversationally, various aspects of it. But one was sent this morning at 2:03 a.m., which I guess is probably in the middle of Chris Day's day, but it's from Chris Day in Princes Risborough in Buckinghamshire in the U.K., with the title "SpinRite recovers a Samsung SSD 840 EVO from the Performance Restoration Software." And I didn't quite understand that. But it turns out that performance restoration software is Samsung's software to restore performance, which broke the drive.

So anyway, he said: "Hi, Steve. I've been a SpinRite owner for several years now and have used your excellent product from time to time on my systems and servers at home. I recently heard about the problems with the Samsung EVO SSDs slowing down on your - oh, on your brilliant Security Now! podcast with Leo."

Leo: It was a trim issue. They weren't properly implementing trim.

Steve: Right. And he said: "I'm a CISSP and learned all I needed to know about crypto, hashing, et cetera, et cetera, to pass the CISSP exam just from listening to your podcast over the last 10 years. As my laptop has a Samsung SSD 840 EVO, and Samsung have recently released the Performance Restoration Software to resolve the slowdown issues, I decided to apply the update to my system. What harm could it do?"

Leo: [Laughs]

Steve: I downloaded and ran the software, following all the instructions, apart from the backup, as all my data is synchronized with my server, and I have a base-build image of a patched Win 7 OS and core programs as I rebuild my laptop every six months or so." Good for you, Chris. You're doing it.

"So everything went smoothly. The drive firmware was updated, the laptop rebooted, and the Performance Restoration Software went to work rewriting every sector on the drive and completed successfully. I left the computer, and the next day when I came to start work the laptop wouldn't boot into Windows."

Leo: [Groan]

Steve: Oopsie.

Leo: [Groan]

Steve: "I ran through several cycles of rebooting to make sure, and not even recovery mode would work. Nothing. So I grabbed my copy of SpinRite and ran a Level 2 scan on my Windows drive. Thirty minutes later, SpinRite completed, and its completion screen proclaimed all was well and nothing amiss. I rebooted the laptop and, as expected, it fired right up into Windows perfectly.

"Now, I can't give you a great sob story of how my life's work was on the machine without any backups, and how SpinRite saved my children from destitution. But I can attest to the efficacy of the product and tell you and your listeners that it's the best $89 I ever spent."

Leo: Yay.

Steve: "Regards, Chris Day - MBA, BSc with Honors, CISSP, MCSE, CITP, and MBCS."

Leo: OverAchiever.com.

Steve: And XYZ. Longtime Security Now! listener and first-time contributor.

Leo: That's great.

Steve: And Chris, thank you very much for sharing your success.

Leo: I love it. All right, Steverino. Got questions for you. And I know you have answers.

Steve: And I knew we'd be long, so I chose eight rather than our regular 10.

Leo: Easy, easy.

Steve: Yeah, it'll be fine.

Leo: Steve, are you ready? Do you feel - you've got your thinking cap on?

Steve: I've had some more coffee, so yes.

Leo: During that break.

Steve: I've been sipping.

Leo: Paul from Burlington kicks us off with our Q&A, our "listener-driven potpourri," as you are wont to call it. He's been thinking about these persistent cookies from Verizon, and I guess now we know AT&T's also doing it. He writes: Steve and Leo, I thought during the Q&A, if you want to block the cell service provider cookies, well, use proXPN because proXPN is the only people that ever see the cell provider tags. The remainder of your browsing traffic will be within the encrypted payload. Of course you could always just use WiFi or, better yet, both - proXPN over WiFi. Is that true?

Steve: Well, yeah, except that this spoke of a little bit of a confusion that I wanted to take the opportunity to correct because I saw that a number of people had it. What Verizon and AT&T are doing is they're adding - and this is actually why techno purists like me and so many others are annoyed is they're actually modifying the traffic. They're seeing a query in the clear, not encrypted, going out of one of their subscribers. And they're adding their own header to our headers. And it's just, I mean, the idea that they're doing it just, you know, that they're making a change to our headers. It would almost be like they themselves monitoring the cookies that our browser is sending. I mean, who knows that they're not? They could be. But in this case they're adding their own cookie because, if the traffic is unencrypted, they can do that. And that's the key.

Leo: Yeah.

Steve: It doesn't matter how you encrypt it. You could encrypt it with proXPN or just HTTPS. They're unable to do this completely during any encrypted transaction, whether it's tunneling encryption that completely bypasses them or, you know, HTTP is a tunneling protocol. Technically it brings up a standard TCP connection and then negotiates with a handshake the keying and authentication that'll be used to verify the endpoint, that you're talking to the server you expect. And then nothing you're doing in there is available to the outside. So these guys are completely blind to any form of encryption. So where Paul sort of misunderstood what was going on was he said that only proXPN could see the service provider cookies.

Well, in fact there are no service provider cookies. Either they're able to alter the actual headers in the query to embed their cookie, or not. And if not, then they're adulterating your traffic in no way at all. They're making no change at all. So any kind of encryption that you imply. Yes, a VPN means that secure or nonsecure traffic is protected. Or if you can arrange for all the sites you go to that you care about, and more of the major sites every day are encrypted, then that's protection, too. So it ends up only being those sites that don't have persistent security where there's really any problem represented. And we're seeing more and more sites bringing up HTTPS all the time, which is good.

Leo: Yeah. Greg, writing from an undisclosed location, wonders about big data center security: Steve and Leo, longtime listener and all of that, since the very beginning. The podcast usually covers down-in-the-weeds technical stuff. But I've been curious about a larger strategic security topic: How on earth do medium and large websites stay online despite all of the constant zero-day exploits out there and the complex software running behind most web servers? I wonder that myself sometimes.

There are always some vulnerabilities we don't know about. So wouldn't it make sense for attackers to use those on high-profile targets like Google or CNN? Obviously these sites have a lot of redundancy built in, full-time security staff. But why don't we see more frequent defacing or takedowns of high-profile sites? What measures do businesses put in place to remain online? That's kind of the more interesting question.

Steve: Yeah. And I thought it was a great question. And at one point, as you were reading that, I was thinking, yes, on a wing and a prayer. But really it's the case that we now understand how to do this. I would argue that in the beginning we were still, "we" as an industry kind of collectively, we're still learning about things like cross-site scripting vulnerabilities and cross-site referral exploits and, lord help us, SQL exposures. And so there were a lot of mistakes being made. And hopefully we saw other people getting bit by those and then fixed the same things we were doing before we got bit by them.

So it's definitely the case, as Greg mentions, that these large sites are very complex. But it isn't impossible technology to get right. It's just hard. So the fact that they've got a full-time IT staff, and that they scrutinize everything that they do, and that they have an architecture designed for security, that is designed with security in mind. It's no longer the case that you can take a system out of the box and set it up and, without any regard for security, go on the 'Net, make this thing globally available with all kinds of bells and whistles and have no problems. You're going to have problems.

And I remember the time, back like 10 years ago, or actually a little bit more, when I was implementing my own eCommerce system from scratch. There were shopping cart applications you could get. That's the last thing I was going to do because they were all proprietary. They were closed source. And we kept seeing them having egregious problems. Well, we don't see that so much anymore. So those lessons have been learned. We understand now to a much greater degree how incredibly careful we have to be if we're going to put this content on the 'Net.

And it's still the case that the smaller, less conscientious sites are having problems because they have unprofessional people who are sort of using the default setup. They've got SQL Server running and with exposed ports because, oh, it's convenient, or that was the default settings. And where they're not bolted down, they're still getting low-profile attacks. But the big boys really, by really focusing on what the vulnerabilities are, today I think we've learned how to make them secure. We didn't understand that to the degree we do 10 years ago, but we're moving forward, thankfully.

Leo: Yeah, I could talk about this on and on and on.

Steve: Yeah.

Leo: Because we have, you know, there's a lot we have to do.

Steve: Well, and you guys have used things like Drupal, where there have been security problems, not in your code, but in the packages that you were importing.

Leo: Right. And there just was a big one in Drupal itself.

Steve: Yeah, that's why it came to mind, yeah.

Leo: So it's somewhat similar, though, in thrust and idea, to the way a normal user keeps themselves secure.

Steve: Right.

Leo: You keep updated. You keep up on what's going on. It's just that there's a lot more software, and you have a public face. So there is a place people can bang on you. At home your public face is your router, and it's stupid. But we have a server as a public-facing entity.

Steve: And I think in the typical web experience you also have a more heterogeneous environment. That is to say that many websites are assembled from pieces to fit the need.

Leo: Right.

Steve: And so many companies' needs are different. So no two are exactly the same. And so there can be unsuspected interactions between them to create opportunities for exploitation that exist here but not there. So it's not easy. And you're right, Leo, it's something that it's important to do, and it's worth doing. And look at the value that we now get from our sites. It's just, you know, it's the way the world works.

Leo: Yeah. But it's doable. It's not as hard as - and as you say, we've learned a lot. From Pedro Tarrinho in Porto, Portugal - oh, I like this. Let's just go there to answer this in person. Steve, my name is Pedro - and I know I'm saying it wrong, Pedro - Tarrinho, from Portugal. I'm a new viewer and listener and enjoying the podcast very much. Regarding the comparison on digital certs, besides the support level, there's another big difference. To my knowledge there are three kinds of certs: the DV, Domain Validation; the OV, Organization Validation; and the EV, Extended Validation. The DV will only validate the domain and can be done automatically and quickly. The OV will do the previous analysis, additionally verify the domain is registered to a company, and that the request comes from an authorized individual at that company. The EV, well, that's an exhaustive analysis which includes the previous ones, plus further verifies the identity of the company making the request. Keep up the good work. Regards, Pedro.

Steve: I really appreciated this. I don't think I've stressed enough that this is the way certificates break down. I know I've talked about Extended Validation from a technology standpoint extensively. But I was glad for Pedro's addition here because we were talking about the cheap certificates and, like, scratching our heads, like wait, how can this be, whatever it was, $4.80 a year or something. Well, it is only for Domain Validation. That is, essentially all they're asserting is that the certificate for the domain is for the domain. And so it's sort of the default level of establishing a secure connection with a remote server. But we've also said oftentimes that's all you need. I mean, increasingly, it's useful to know that you're really connected to an authoritative source.

So PayPal and eBay and GRC and other properties will be protected by an Extended Validation domain, broadcasting the fact that all of this research that Pedro enumerated did go into and goes into continuously every two years, because that's as long as those certs can live, to assert absolutely what's known about the entity that was issued this certificate, which the server you've just connected to has provided to you. But for blog posting and protecting low-level properties, where you just want privacy, there's absolutely nothing wrong with the so-called Domain Validation certificate, where the only thing it's asserting is that this is a certificate for this domain. So good luck.

And so they're not very expensive. And we would like them not to be very expensive because we'd like to promote the use of HTTPS everywhere and as much as possible. They can have longer lives, so they're less inconvenient. And all they're asserting is this is a certificate for that domain. And that's all we're saying. So I really - I'm glad Pedro mentioned this because I have not gone into enough that there really is this spectrum from minimal assertion to maximal assertion. And what you pay for is the work on the other end of standing behind those.

Leo: Yeah. Question 4 from Troy K. in Kansas. He says he's wondering about Secure Erase and Enhanced Secure Erase: I use Secure Erase to clear SSDs. The following site discusses the difference between Secure and Enhanced Secure Erase, it's partedmagic.com. I admit I'm confused by how this would work on a spindle drive, but it makes sense on a solid-state drive. A verbose report can be generated at the end of the process. I've had the media tested. Data has not been recovered from these drives. There appears to be some documentation available. But you know what, we get this question a lot. It is worth revisiting the topic because you are the king of hard drives.

Steve: Well, and I'm going to - I wanted to say - he asks is there any chance we will revisit the topic. And I still have my own drive-erasing utility on the burner for work or for creation after I'm through with the SpinRite series.

Leo: Oh, good.

Steve: Well, the SpinRite 6 series, and before I start on 7. I have a trademark on the name Beyond Recall for years. So this is something I've always thought made sense. So rather than attempting to address it now, I'm going to wait until I am deep in the weeds, developing something that will be able to claim absolute mastery over erasure. And at that point we will know all about it.

Leo: But there are specific challenges to solid-state that don't exist with spindle drives because of the wear leveling; right?

Steve: Well, kind of, except that the physical media has sector relocation. And it's different than wear leveling. Wear leveling is being done on the fly all the time in order to, as it sounds like, level the wear on an area. What's been found is that access patterns across a large chunk of mass storage is extremely non-uniform. You've got a swap file, typically. You have a temporary file directory. You have even the directory tree itself. That so-called metadata sits in specific sectors and clusters on the drive. And during the course of use, many of those are being rewritten to a much greater degree than, for example, the original OS files that are only being read. They're almost never being rewritten. Or files, you know, like applications that you install, and they're only being read and never written.

But as we know, these solid-state drives fatigue because the way they work is by deliberately stressing their insulation. By forcing electrons through an insulating layer and stranding them to create an electrostatic charge, we are able to store a bit. And the act of pushing electrons through the insulation fatigues, it breaks down that insulation over time. And so reading is - it doesn't require moving electrons. It requires only sensing them. But writing requires that we either pump or pull electrons through that insulating barrier, and so that fatigues the drive. What that means is that excessively writing in one area will physically fatigue certain locations.

So understanding this, this sort of a meta architecture was developed where unused areas, the actual contents of unused areas would get replaced with the contents of heavily used areas. That is, so that even though, from the outside, we always appear to be writing on the same physical sectors, the same sector numbers, there's a translation layer which says, oh, now at the moment we've moved that data over here because it was being written a lot. So now it's over here, so intercept any read and write requests to this range and send it over to that range.

And so this process is going on all the time. What that means is that it may be that some areas of the drive have older data that has stopped being written to in favor of wear leveling which is writing to a different area. Yet forensically, if you were to penetrate that translation, that meta translation layer, and access all of the physical drive, you could find data that you thought you had erased, but you'd erased it through the translation layer, which wouldn't have given you access to this briefly out of service, previously written area.

So what secure erase does is essentially says, okay, we're deliberately shutting down meta translation. We're going to do a true physical access to the actual media and, one last time, or maybe not, if you want to keep using the drive, but we're going to absolutely wipe it all so that no one can get to it again. And so this is similar to the secure erase function on hard drives because hard drives take sectors out of service which the drive no longer trusts. It may still be readable. It's just that the error correction required started to freak the drive out. It's like, oh, this is requiring near my maximum amount of correction, or retries in order to read it at all. This is getting flaky. I'm going to stop using it.

But it leaves the data there because it's in a hurry, sort of someone's waiting for their data, and so it just says, okay, stop using it, we're going to copy - now, we did successfully recover the data from that sector. We're going to put it over here. And future references go here, not there. But the data's still sitting there. So again, the different types of erase penetrate these translations in different ways and to different degrees. But it turns out it's not simple. It's not just a matter of saying, oh, format, which we learned long ago when Peter Norton famously taught us that format does almost nothing because, if you can have an unformat command, you know you didn't really achieve much.

Leo: Hey, you know that drive you formatted? Can you stop that?

Steve: It'd be like unerase. Oh, I want to issue the unerase command.

Leo: Right.

Steve: Except, wait a minute, if you can do that, then...

Leo: It ain't erased.

Steve: ...we've got a problem, yeah.

Leo: Yeah. Curtis in Phoenix wonders about credit card payments with SQRL. SQRL is, of course, Steve's amazing magical website authentication protocol that he's...

Steve: To be demonstrated momentarily, actually.

Leo: Oh. That's exciting.

Steve: Yeah. We're about to have it - it's about to be running.

Leo: That's exciting. Can't wait for that.

Steve: Yup.

Leo: I envision scanning a QR code displayed on a payment terminal by my phone, he says. Using Target as an example, I set up an account at Target.com with my CC info and my SQRL ID. When I'm in the Target store, the terminal displays the QR code. I scan that with my phone. And then the magic happens, by telling Target who I am, authenticating, and then using that credit card information that Target already has on file.

Now, granted, they'd still hold a database of credit card information that could be stolen, but I would think those databases are more secure than the pay terminals. These options would make it just that much harder for an attacker to gain information. Another option would be to transmit the credit card info or some ID like Apple Pay uses in the - I've got the hiccups, sorry about that - in the SQRL conversation between my phone and Target's servers. That way they wouldn't even have to store the credit card number. What do you think, Stevie?

Steve: Okay, so I'll talk while you get some water or drink upside down or...

Leo: I'm going to drink upside down.

Steve: I think, you know, I did hear that a spoonful of sugar, that's the one.

Leo: That's the actual remedy.

Steve: Yes.

Leo: Not an old wives' tale, but scientifically proven.

Steve: Knocks them right out.

Leo: To calm the spasming diaphragm.

Steve: Yeah. Interesting, yeah. Okay. So, Curtis, yes. It's true that exactly what he suggested would work. SQRL is, from my standpoint - and people are starting to ask about FIDO and does FIDO make SQRL obsolete, or where does it sit and so forth. The thing that I like about SQRL, that I still like about SQRL, is its incredible simplicity. What it is in its heart is a very simple, using state-of-the-art but easily explainable crypto, authentication system. It avoids the problem with replay attacks. The problem with a credit card is replay. A credit card attack is a replay attack. Someone gets your credit card, and they use it again. They replay the number that they got. So, and username and password, replay. They capture your username and password, and they replay it. They use it again.

So what SQRL does is it just - it's like the smallest thing you could do to solve that problem, which is you have a secret, and the server generates a random challenge, which you sign using your secret, and you send back to the server your signature of its random challenge. And the challenge is always random. So we solved the replay attack. The server is never going to, or never going to predictably offer that same challenge again. And only if you're really you do you know, do you have the secret that allows you to sign that random challenge. So when that goes back to the server, it checks the signature, and it says, oh, yeah, someone just signed this correctly. That must be him because nobody else can. And it's that simple.

So what Curtis described is how you would use a generic authentication system - and that's what SQRL is, it's a generic simple authentication system, but it's cryptographically secure - how you would use it for credit cards. And so while it's, yes, that would work, my feeling is, well, but then so does MCX and Apple Pay. And so if alternatives didn't already exist that people were already going to be using, and if we sort of like needed SQRL for that, then, yeah, it could certainly work that way. But I'll be surprised if that happens. On the other hand, websites in general is still my target because they still haven't got a universal simple comprehensive solution to replace usernames and passwords. And SQRL does that very easily. So I think there it makes sense.

It could absolutely, I mean, I know - the reason I spent some time on this is that there's all kinds of applications for authentication. Maybe we will see this simple protocol that I've developed for SQRL used elsewhere. And I know people are going to be saying, hey, about this and how about that? It's like, yeah, it can do anything where you have a random challenge, and you are able to assert you're you by signing that challenge and sending it back. It's able to do that.

So what enables this that we have now is smartphones or clients in our machines. Once upon a time, all we had, you know, we didn't have those, so we had to use username and password, something that we could memorize that didn't change. And of course we now know that's dangerous in all kinds of ways. So the good news is the world has moved forward. It's easy to run clients in our computers or our smartphones that are active and have the computational power to answer a cryptographic challenge. And so that's really all SQRL is. I mean, it's pretty simple.

Leo: Does it eat nuts? I'm sorry.

Steve: It can do that, too.

Leo: I'm having some right now.

Steve: It can solve your hiccups.

Leo: Mm-hmm. I'm eating a little sugar. It worked. Bubba Mustafa, or is it Bubba? Bubba Mustafa - I'm sorry, I'm sure it's his real name, and I shouldn't be laughing - wonders about long-term data storage.

Steve: It's a beautiful fun name.

Leo: Bubba Mustafa. Man, if my name were Bubba Mustafa, I would be so happy.

Steve: I love it. I love it.

Leo: Steve, I heard you talking about data decay. What are our options, say for a true 10-year retention? Do archival-quality CDs and DVDs live up to the sales pitch? The CD version seems just to be the original gold CDRs when CDRs first came out. I have to admit, for regular CDRs I have had the aluminum coating just delaminate and peel right off. Thanks. Still haven't needed to use my SpinRite, but I'm happy to know it's there for the inevitable.

Steve: So this is really a good question. And I'm finding myself wondering if maybe the cloud isn't, like redundant cloud provider storage isn't a solution, sort of making it somebody else's problem, essentially, and maybe adding some redundancy. The problem is, I mean, we talk about time capsules and the fact, for example, that many people would have a hard time today reading a floppy. I mean, and that's a 3.5-inch floppy. What about a big one? I think I've got one around here somewhere. Actually I have a bunch. I've got 5.25s and 8-inch floppies. Now, someone says, okay, I have an 8-inch diskette with data on it that I need. What are they going to do? In fact, I saw somebody on one of your podcasts, you'll probably remember it, Leo, in the last month or two or something. He was having a business maintaining drives that would be able to pull obscure data formats off of media.

Leo: It was on the radio show.

Steve: Oh, okay.

Leo: Yeah. So this is a - and we talk about this a lot on the radio show. I think this is something people are kind of interested in. And I always say, you know, nothing will last forever, yeah. Not even paper. Carve it into stone because that's the stuff that's lasted thousands of years. We know that CDs and DVDs not only delaminate, but they can get, in humid climates, they can rust. You're holding up a QR code. What is that?

Steve: I'm holding up what my SQRL client puts out. And because it is something that I absolutely know - I can't seem to get it centered - but there is an ASCII version of a SQRL identity.

Leo: That's what the QR code says.

Steve: Yes. And so you could either scan it with the QR code, but if you were even absent a camera, I don't know how that would happen, but...

Leo: Type it in.

Steve: Type it in. It's not that long. It'll tell you if it's right or wrong, and then you go from there. So, I'm sorry, but, yeah.

Leo: I agree with you on the cloud thing because really the issue, once it's bits, we've kind of solved some of the problems because it's infinitely copyable and perfectly copyable. So there's no copy degradation. And that's a big issue with print, with handwritten stuff and so forth.

Steve: Oh, those monks, they did amazing work, didn't they?

Leo: Yeah, but still errors would creep in, transcription errors.

Steve: Yeah, sometimes there were little embellishments and, you know, I mean, they were bored. If you were a monk, you'd have a hard time, too. It's like, this is such boring text.

Leo: So once you've got bits, kind of you've done a big thing, but now we have to keep it on media that doesn't obsolete. And you mention floppies, but ZIP disks, ORB drives, Bernoulli drives.

Steve: Got them all.

Leo: Winchester media, all of those things, they seemed like they'd be good enough.

But a few years, and they're gone. So I like your cloud idea because what you're really doing is you're transferring the issue over to the cloud provider. Sure, it's on some format that will be obsolete probably sooner than later, in a decade or two.

Steve: But they'll migrate it.

Leo: But they are now going to migrate. As long as they stay in business, they're going to migrate. So, I mean, and that's what you could do. Keep it on a hard drive, and then every few years rewrite it. Keep it on several hard drives. Of the physical media we have available to us now, what do you think is the most robust?

Steve: Optical. Everybody agrees that, even though it seems sort of counterintuitive, the recording technology - now, again, it is the case that we've seen problems with disks delaminating and the aluminum layer oxidizing and that being a problem. But that's why they use these gold disks and so-called "archival quality." I've seen studies done that said archival-grade CDs and DVDs truly have a hundred-year life. That is, you can record on them. You've got to keep them in the dark because a little bit of light coming in will accrue, essentially, sort of like film, slowly fogging over time, if you allow any light to get to it. So you've got to store them in absolute dark in a sealed, moisture-controlled, dark container.

But everything that - all the tests I've seen where they've deliberately aged these things, if you need your own media, that's what I would do. I wouldn't trust a drive, frankly, because, I mean, they've got bearings. And it's a high-tech device. And they last for, what, maybe three or four years? And then it's like, oh, well, I need another. I mean, unfortunately, their capacities are growing so much, and our need for storage is growing so much, that we're constantly moving upwards in drive size.

Leo: Yeah, and the best thing would be multiple forms of media; right? I mean, this is the...

Steve: Yeah.

Leo: So don't trust any one. I mean, not merely optical, but optical and hard drive, or maybe Blu-ray and DVD, I mean, just multiple choices; right?

Steve: Yeah. And you ought to also consider maybe, if you had - if you were writing to CDs and DVDs, store the drive you used along with them. Put them into the time capsule because the chances are, depending upon what the future holds, you'll have the drive that was able to read them once. And then your job is an electrical interfacing one to the drive, rather than, oh, my lord, how do I get the optical tracks off this thing?

Leo: Right.

Steve: So that's an easier problem. Also, you'll notice that I mentioned redundant cloud usage. Was it Bitcasa that just announced they were shutting down, and everyone's storage was being lost.

Leo: Bye-bye. Yeah.

Steve: Yeah.

Leo: So more than one cloud server - service.

Steve: Right, right.

Leo: And keep an eye on their businesses.

Steve: Or use Amazon or Google or Microsoft. Use a major league provider that has, instead of Bitcasa, which is like, okay, well, you may like them, but obviously they decided they were changing their mind about what they were going to do.

Leo: Amazon has this Glacier solution.

Steve: Yes.

Leo: Which is inexpensive. And it's very cheap because it's slow to restore. It might take a day to get your data. But who cares?

Steve: Yeah. It's actually offline storage. So they're filling up something, whether it's drives, or they might even be using those really cool, like, tape spool technology, where they fill them up, and then it's like it's actually not accessible electronically in real-time. Somebody or something, a robot arm or a person, has to go get it and plug it in, in order to access it. So, yeah. I really think there's a - the problem of destroying data is a big one. But the problem of really keeping it archivally available is interestingly big, too, actually because we don't really think about it much. It's not a problem everybody has, but some people have that problem. And there isn't really anyone really addressing it.

Leo: When we sent V'ger out into space, we put a gold disk in it because gold - not because gold is so pretty and valuable, but because it doesn't corrode.

Steve: Right.

Leo: And it has, I presume it's not analog, maybe it is, an analog recording. I guess, yeah, it probably is because probably any relatively advanced civilization could figure out what to do with that.

Steve: Yeah, they did a bunch of neat things. They did things like the size of the raster scan on the image was two prime numbers. So there was only one way to decode this into a rectangular...

Leo: So that's going to get really big brains thinking about how would we communicate with a completely alien intelligence? And oh, by the way, it's got to survive millions and millions of light years in space.

Steve: Yeah. And they know nothing about us, and they look at this gold disk and go, hmm. Well, let's see. This looks like a spiral. Well.

Leo: And then they eat it.

Steve: Oh, it looks good.

Leo: Cookies.

Steve: Because, exactly...

Leo: Love that gold.

Steve: They're creatures that live on gold, and it's like, ooh.

Leo: Good gold.

Steve: Or they immediately melt it down and resell it.

Leo: Right. Pretty.

Steve: Yeah.

Leo: Yeah, it is an analog track, apparently. I think analog makes the most sense. Although ones and zeroes, binary is pretty well understood. I think you could deduce that there's information in the switching back and forth of ones and zeroes. Maybe not. It's not decimal. That would definitely be anthropocentric.

Steve: Yes.

Leo: Fascinating, though, isn't it?

Steve: Yeah, it really is.

Leo: Kevin in North Carolina wanted to clarify. Oh, this is from last time.

Steve: You know, when those meteors land on Earth, Leo, we just think they're rocks. Little do we know.

Leo: Oh, look.

Steve: Look at that rock. And we stick it in a glass case. It turns out that's a desperate attempt at communication, and we've completely missed the point.

Leo: You know, that actually seems reasonable, frankly. What would be the chances that we would recognize the communication?

Steve: Yeah, it's like, look, what's this rock?

Leo: Yeah. Kevin in North Carolina wanted to - because we answered his - so he says: Short-time listener Kevin here again. Thank you for addressing my previous question regarding random MAC addresses. I don't mean to be a pest, but I wanted to clarify. I wasn't referencing the Apple randomization while searching for WiFi. Oh, because we had been talking about that recently.

Steve: Right.

Leo: My feedback was in regards to another question that came up previously about MAC addresses in general. The other listener asked: Why doesn't every device use random MAC addresses? Why even bother having a fixed, unique MAC address? It doesn't make any sense. So that's back to four episodes ago that we answered that one [SN-476].

Steve: Yup.

Leo: As was discussed then, it's feasible to have randomly created and assigned addresses. But there is a value in a fixed unique address because it identifies, for instance, the manufacturer. That's the first few digits. So that's what he was asking about. Anyway, thank you, Kevin, for the clarification. I don't think we have to say much about it.

Steve: No. I'll just add that, back then, when we originated Ethernet, bits were expensive. They were slow, and they were expensive. And so it made - so to have a random MAC address you'd have to have many more bits and somehow assign them randomly. And that just didn't make as much sense as chopping it up into two halves and making one be the manufacturer and one be this manufacturer's serial number. It was a clever scheme that has survived even to this day. So anyway, Kevin, yeah, thank you for the clarification.

Leo: Finally, our last question comes from Ottawa. Peter Sysak has been catching up all the way since Episode 1.

Steve: Whoo.

Leo: First of all, I want to say that I love Security Now! and TWiT. I only discovered your show a few months ago. And after listening to a few new episodes I decided, you know what, I need to start from the beginning. Holy cow. Holy cow. So for the last few months I've been trying to work my way through everything since Episode 1. I can't tell you how daunting a task that is, but I find there's always something useful. As such, I haven't been able to bring myself to skip too many episodes. I really wish I had discovered this show years ago. In a way, it's really interesting to listen to the old shows today and hear you guys go on and on about all the stuff that was happening a decade ago. A decade ago.

Steve: Our own time capsule.

Leo: Yikes. Obviously, there are a ton of things that are long forgotten today. But it's also funny how some things just never change. With that, I wanted to ask you a question about an episode I just listened to, namely Episode 81 in which you discussed the Google report on hard drive reliability. Forgive me if you've done something like this more recently, but I - 81? You mean 400 episodes ago? Eh, that's recent. I was wondering - 400 episodes ago. We do 50 a year. All right. I was wondering if there's any way you could touch on that topic again, but speaking to today's hard drives.

Is SMART still as dumb as it was then? Yes. I'm personally running a FreeNAS server with a RAID-Z configuration consisting of several 2TB drives. I'm relying on SMART tests to warn me about impending disk failures so I can go buy a new one in time. But now that I listened to Episode 81, it's kind of shattered my sense of comfort with that setup.

As a secondary question, does SpinRite do anything with SSDs? I never knew about SpinRite until I knew about the show, and basically everything I have today runs an SSD. Really? His 2TB drives are SSDs? Whew. No. Thanks for the great work you guys do. I'm slowly catching up. I'm now, ladies and gentlemen, on Episode 82. Wow.

Steve: So Peter, here's my advice about catching up. Don't be in a hurry. Don't worry, I mean, first of all, the information - we're archivists. I've got a lot of mileage left on me, and I intend to have all of these 10 years online for the foreseeable future. So unless you're 95, you're going to have time. So I would say don't be daunted by it. Don't press yourself. Also try not to skip them because there really is often surprisingly good stuff every so often in these things. So unfortunately I can't vouch for them all. But I think generally people never feel like they've just had a really dud Security Now! podcast. So take your time.

SMART is all you've got, so it's better than nothing. But as Leo added, it is still dumb. The problem is...

Leo: It hasn't gotten any smarter since then.

Steve: No.

Leo: Let's put it that way, yeah.

Steve: The reason SpinRite and SMART work so well together is that SMART can't tell you anything about the drive that it doesn't experience. And as we were just talking about how lopsided access patterns typically are in drives, where a very small area of the drive is in heavy use, it's the fact that SpinRite goes out and covers the entire drive while monitoring the feedback from SMART that makes the two of them together smart. Now, SMART will warn you, on the fly, if it's seeing a problem in advance of a collapse. But it has to experience that for that warning to be useful. So that's really where SpinRite comes in handy. SMART's better than nothing.

And having a RAID architecture which is able to periodically query SMART, because it's not an announcement protocol, it is a query-response protocol. You've got to ask the drive how its SMART subsystem is feeling right now in order to get a response. SpinRite polls SMART on the fly and displays and analyzes the information that it gets from the SMART subsystem in order to determine what's going on. Your RAID system is also polling periodically to check on the health of the drives as they're reporting it. The problem is they don't know about areas that they're not accessing dynamically.

And what's really interesting is that SpinRite works the drive hard, and we see the SMART health being depressed by SpinRite. And then over time, it recovers because, I mean, and that represents some problems on the drive that are of a severity you can judge based on how far depressed the drive reports its health to be. But the point is it has to be doing work in order to judge the health of its ability to read. And I'll just close by saying that in the testimonial in this podcast you just heard, assuming that you're still listening to the current ones and not...

Leo: He'll hear it in a couple of years, anyway.

Steve: You'll hear it when you catch up.

Leo: Yeah. Hello from 2014 to those of you in 2018.

Steve: Exactly. The testimonial was about this guy using SpinRite to recover an SSD that the SSD's own manufacturer broke when it was trying to fix it. So, yes, SpinRite absolutely does repair SSDs. And we've actually - you'll find, as you catch up, many other testimonials from SSD users, which is actually one of the reasons that I decided to get interested in a future beyond SpinRite 6, which I call 7, because even though magnetic media may stop spinning, it looks like the solid-state stuff still needs us.

Leo: Nice. My friend, we've come to the end of your question-and-answer session.

Steve: Right on time.

Leo: Right on time. On time and on budget.

Steve: And with no hiccups.

Leo: And with no hiccups.

Steve: Of any nature.

Leo: Well, I had little hiccups. Little bit of a hiccup. Thank you, my friend. Steve's so great. I love doing this show, and it's 10 years, it seems like nothing.

Steve: It does. And so it's funny, you talk about Episode 81, it's like, wow, I remember that Google drive reliability report.

Leo: Oh, yeah.

Steve: And that doesn't seem that long ago.

Leo: No.

Steve: But I guess it was.

Leo: Goes fast.

Steve: Yeah.

Leo: We love doing this show. And Steve will be back next week doing it again. We do it Tuesdays, 1:00 p.m. Pacific, 4:00 p.m. Eastern time. That would be...

Steve: Is it now...

Leo: 2200 UTC.

Steve: Okay.

Leo: I add - no, no. Oh. I had a really good...

Steve: Isn't there a minus seven? I thought it was minus seven or minus eight. But that would have been...

Leo: 2100. It's minus eight.

Steve: Oh, okay.

Leo: Was minus seven, but now we've reverted to daylight - not in daylight time, regular time.

Steve: Whatever it is.

Leo: Non-summertime. Standard. Standard time. So it's now 2100 UTC. You know, I have a method now because it's eight. It's I add eight. So 12 plus eight is 20. So I just add 20 to whatever time you start, so it's 2100 UTC.

Steve: Nice.

Leo: See? I thought about this.

Steve: That works.

Leo: If you can't watch live because you can't figure out what time it is, don't worry. On-demand versions always available. Steve's got a great little tiny 16Kb version, sounds like it came from 1928, but it's small. You could also get transcripts which are even smaller. He has handwritten human readable transcripts at his site, GRC.com. You also can find SpinRite, world's best hard drive maintenance and recovery utility, now for SSDs. You can also find all sorts of projects he's working on. And those are all pro bono, so free, free, free. So Perfect Paper Passwords, Password Haystacks, of course SQRL. You could find about Vitamin D. You could find about ketogenic diets. You could - there's all sorts of stuff there. It's kind of like the Dr. Bronner's Soap bottle of websites. Except the type is bigger. You know Dr. Bronner's Soap?

Steve: I heard of it, yeah.

Leo: Yeah. You should get some. It's nice. Peppermint. Very smelly. It smells good. Smelly good. Very good smelly.

Steve: Thus the fact that they're still in business.

Leo: Yeah. There's actually a documentary on Netflix about Dr. Bronner because he passed away a while ago.

Steve: Ah.

Leo: But the kids...

Steve: The soap lives on.

Leo: The kids are still - I think it's in your neck of the woods. You could go to the Dr. Bronner's Soap factory, which is basically a big tub with a little bit of lye...

Steve: People staring at it, waiting for it to dry out.

Leo: Yeah, yup, yup, yup. What else? Oh, we've got high-quality audio and video at our site, TWiT.tv/sn, Security Now!.

Steve: And your own archive.

Leo: Oh, yeah, we've got every show going back. We're redesigning the website, and one of the things I know everybody wants is a one-button download of everything. It's for your show only. Nobody's asked about any other show. But for your show, everybody wants all of them. So, yeah, that's a big checkmark for the new website.

Steve: Oh.

Leo: Oh, yeah.

Steve: Oh, nice.

Leo: Hey, I listen. You think I'm not listening?

Steve: Very nice.

Leo: But I'm listening. I'm knitting. I'm making socks.

Steve: I get it all the time. I get it all the time.

Leo: Yeah. No, I know.

Steve: And hiccupping.

Leo: We also have, you know, all those apps, great third-party developers who've made apps for every platform including Roku. And so you can look for the TWiT app on your favorite platform. You'll probably find it. Windows Phone, iOS, Android, Roku, that kind of stuff, Samsung TVs. And then there's iTunes and all the other places you might subscribe to a podcast. We're there, too. Because, you know, with a show that's been around for 10 years, there really is, I mean, it's like we win because everybody else gave up.

Steve: Yup.

Leo: A long time ago.

Steve: They and Dr. Bronner.

Leo: We're like V'ger. We just keep going. Keep going and going and going. All right.

Steve: Okay.

Leo: You should read, by the way, a little plug, just I'm in the middle of it right now, but it's good, "Whiskey Tango Foxtrot."

Steve: Oh, okay. I've heard you mention it.

Leo: It's a novel. And it's about privacy. And we won't know that immediately because at first it just seems like unrelated stories about people. But it's all knitted together about halfway through. And it ends up being - it's sci-fi. It ends up being a massive global conspiracy to invade our privacy, not by government, but by private industry.

Steve: No. They wouldn't do that.

Leo: They would never do that.

Steve: No. Monetization.

Leo: It's really - it's really good. It's funny, and it's fun, and I think you'll like it.

Steve: Oh, neat.

Leo: It's a cautionary tale.

Steve: Yeah.

Leo: WTF, "Whiskey Tango Foxtrot."

Steve: [Laughs]

Leo: Oh, now it comes home. Yeah, now you know what I'm...

Steve: I get it. I don't know what's slowing me up. I'm a little slow.

Leo: Now you know. Yeah. That makes it easy to remember.

Steve: Yeah.

Leo: Hey, thanks, Steve. We'll see you next - actually, stick around. Did you get the Voyage, the Kindle Voyage?

Steve: Oh, no. It's pre-ordered. Jenny and I get ours on the 17th because I just didn't know about it in time. But I've heard you guys talking about it.

Leo: My review's coming up on Before You Buy, next.

Steve: Oh, good. I will absolutely switch over to the live feed and watch because I guess from what I understand the type is so crisp, it just completely changes the experience. And not having the screen - well, I don't mean to encroach your...

Leo: It's fast. No, no, no, it's good. No, you can look. You see the page turn is so much better now than it was.

Steve: Oh, nice. I love the squeezing on the margins.

Leo: Yeah. This is natural.

Steve: I like that better than having to touch the screen.

Leo: This is so natural, isn't it?

Steve: Oh, nice.

Leo: Yeah. Well, I'm going to give it a definite - you know, it's expensive is the only negative. It's 200 bucks.

Steve: Yeah. Mine and Jenny's is on the way. I ordered them the moment I found out about it. And I guess I must just be - they must be backlogged.

Leo: Yeah. I'm sure they are.

Steve: Wow, nice.

Leo: Nicely - they did a nice job with this. And I got the Amazon weirdo case. I don't - the Origami. I don't understand it. It's too complicated for me. Does something here, I'm sure.

Steve: Oh, you're able to, like, create a stand out of it.

Leo: Yeah, it makes a stand. It's weird. Anyway, thank you, Steverino.

Steve: Okay, buddy. Talk to you next week.

Leo: We'll see you next time.

Steve: Bye.