Are You Human?

Description: Steve and Leo explore the Internet's rapidly growing need to automatically differentiate human from non-human automated clients. They discuss the advantages and limitations of many past and current approaches to this problem while paying close attention to the most commonly used visual "CAPTCHA" solutions.

High quality (64 kbps) mp3 audio file URL: http://media.GRC.com/sn/SN-101.mp3
Quarter size (16 kbps) mp3 audio file URL: http://media.GRC.com/sn/sn-101-lq.mp3

INTRO: Netcasts you love, from people you trust. This is TWiT.

Leo Laporte: Bandwidth for Security Now! is provided by AOL Radio at AOL.com/podcasting.

This is Security Now! with Steve Gibson, Episode 101 for July 19, 2007: Are You Human?

Time to talk about security, everybody's favorite subject. Well, maybe not everybody.

Steve Gibson: Well, certainly, what, about 140-some thousand?

Leo: 150,000, something like that, yeah. All the people who listen to this podcast, anyway. We're so glad you're here. Steve Gibson is here from GRC.com, the able man behind SpinRite, the disk recovery and maintenance utility that we talk about all the time and all those free programs like SecurAble. You know, somebody told me - we're going to talk in just a little bit about "Are You Human?," which will be fun. But somebody told me that one antivirus...

Steve: Oh, it's AVG, Leo. And it is driving us out of our mind.

Leo: It's a false positive; right?

Steve: Of course. And it happens with my security stuff from time to time that AV software will get tripped over my programs because very often the same sort of code sequences that I'm using to check something about security or to turn something off or on, that'll naturally be the same sort of references that malware would have for doing the same sort of thing, but for its own malicious purposes. So it's incredibly annoying. Greg, my support guy, has been getting

flooded with literally hundreds of reports, hey, you've got a virus in SecurAble; or, you know, AVG pops - it's like, no.

Leo: I've only gotten one, and of course that's what I said. These are false positives. It's not just Steve's software, it happens in other programs, too. And antivirus companies do their best to avoid them, but there's inevitably some overlap, and that's what happens.

Steve: So for what it's worth, if we have any listeners who have and use AVG and get a false positive for SecurAble, I would love it if you could put some pressure on AVG to fix their program, their virus patterns, to remove SecurAble from their list through whatever means they have for doing that because...

Leo: I think we probably have a very high percentage of AVG users listening to the show. I don't know if that's because they're cheapskates or what.

Steve: Well, smart.

Leo: Smart. They don't want to spend money on an antivirus, and AVG's good enough. They know what to do to keep themselves clean anyway, so that's all they need.

Steve: Exactly.

Leo: That makes sense. So, good, I'm glad we got that out of the way. And let's get a couple of more letters from the mailbag. Do you have any addenda, too, we should talk about before...

Steve: I do. I wanted to announce, since we're going to be now doing - we're getting a little more complex with our mathematical formulas, Leo. We're going to have the mod 4 plus 2 episodes are going to be mailbag episodes.

Leo: That's next episode, then.

Steve: Exactly. That'll be 102 will be the first mailbag episode. I wanted to give people an alternative way, an easier way for sending stuff to me. At the moment all I've ever talked about is having the web form down at the bottom of the Security Now! page. Actually, however, that web form has always simply done an email to [ADDRESS DELETED DUE TO SPAMBOT DISCOVERY], with no exclamation point. So anyone who wants to just send feedback about Security Now! issues, you can write to [ADDRESS DELETED DUE TO SPAMBOT DISCOVERY], and it'll come right to me. I did the web form initially because I really wanted to stress that people were welcome to be completely anonymous. And of course a web form rather than someone's email client is an easier way to anonymize yourself, which I wanted to promote. But for people who don't care, and we of course have lots of people who are happy to acknowledge their listenership, you can certainly just use your email client and write to [ADDRESS DELETED DUE TO SPAMBOT DISCOVERY].

Leo: [ADDRESS DELETED DUE TO SPAMBOT DISCOVERY], all right. That'll qualify you for

the mailbag, but no guarantees.

Steve: Exactly. Okay. Something else very cool happened in the last week. Well, it started with something not so cool. About two weeks ago I was at Starbucks using my T-Mobile account and my laptop to piddle around the 'Net, doing whatever I was doing. And I think I did something with PayPal that upset it. And because later that day or the next day or something, I tried to do something from home, and it said, oh, we've flagged your account for alert status. We think somebody was trying to hack into your account. So you have to jump through these nine hoops in order to - and it was about nine. I mean, it was like, oh, what is all this? I mean, I had to go through all kinds of rigmarole to pacify it. And of course I got freaked out that maybe somebody really was trying to hack my account. But I looked at the date, and I said, wait a minute, I think yesterday at Starbucks I did this. I was the one that was "hacking" my own account, just who knows why, what I did to set it off.

But the point is, I discovered that PayPal has full multifactor authentication. I now have exactly one of these cool six-digit numeric token pieces of hardware from SecurID, get this, for $5.

Leo: I should do that right now.

Steve: Everybody should do that. I wanted to let everybody know who uses PayPal that you can get a secure key from PayPal. You just log into your account. Over on the left you'll see Secure Key. It's only $5.

Leo: We use PayPal for donations. And I get every week or so somebody says, I can't do it, I'm not going to do it, I don't trust PayPal, I've had bad experiences, and so forth and so on. But I have to say, PayPal, especially since eBay bought them, has completely turned around and refocused. And I think people should give them a little second chance. Things like this, I mean, very impressive.

Steve: Well, it's very cool. Now, I have heard, as I'm sure you have too, Leo, of horror stories with PayPal. They have sort of the flavor of, well, everything's great until it goes bad. And when they go bad, it goes really bad. So who knows. But for people who use PayPal who are listeners of Security Now!, who care about the security of their use of computers, I wanted to say, hey, for $5 - well, I mean, first of all, you don't even have to authorize the key. You can buy it for $5 and just have this very cool dongle which, I mean, it's neat. It's got a button.

Leo: Now, I'm looking at my PayPal account. I'm wondering if that's because you had this problem because - this is on the front page of PayPal? I don't see it.

Steve: No, you have to log into your account. And then over on the left-hand column it says "Secure Key."

Leo: I don't have that.

Steve: I think that's what it says.

Leo: I don't have that. So I wonder if that's because at some point you...

Steve: I don't think so. I got the sense that it was...

Leo: Well, let me do a search. I'm on the new PayPal. Maybe if I go back to the original PayPal site it'll be - you know, we use PayPal quite heavily. We have a lot of money going through our PayPal account. And I'd really very much like to secure it in any way we can.

Steve: I got the strong sense that anybody who wants to do this, can. And as I was saying, you don't have to follow through and attach it to your account. You could just have it as this very cool SecurID dongle for $5. And so you press the button, it gives you a six-digit number. And exactly as we were talking about, every 30 seconds it will lock that number on the display. But if you press the button to turn it off and then press it again to turn it back on, and if you cross another one of these 30-second boundaries, you get a completely different six digits. Then once you add this to your account, you know, they mail it to you, it takes about 10 days, a week to 10 days. You then, basically you do it, you use it to authenticate yourself, to prove that you have logged on and you received the key. And then the way it works from then on is you simply append those six digits to your password. So it creates an always varying, never the same one twice, and a multifactor authentication because you have to actually have this in your possession. And once you've done that, then if you lose it or don't have it with you or whatever, then you've got to jump through even more hoops to prove to them that you are who you are because they want to make sure that, again, that your account is hardened against anyone taking advantage of that.

Leo: Well, that's cool. I'm glad to see that they...

Steve: But you still can't find it?

Leo: No. I'll find it.

Steve: Shoot. I think it's there somewhere.

Leo: I believe you. I'll find it. It'll be here somewhere.

Steve: Okay. Also we have to talk about a book you turned me onto.

Leo: Uh-oh. You know, because Audible is sponsoring - by the way, Audible is not sponsoring Security Now! for one reason only, because I said they couldn't, because we already have two commercials on Security Now!, and I never want to have more than two on any podcast. They very much wanted to be on Security Now! because they know we talk about books all the time. But that's one of the - I think one of the most fun things about this Audible thing is that we've been talking about books on all of the podcasts. And I know which one you're going to talk about, and that's - it was one of our early picks. And it's just great. So you liked it.

Steve: Oh, well, I'm not through with it yet, Leo. But I am really having fun with this book.

First of all, I had a couple of experiences. Okay, the book is titled "On Intelligence." "On Intelligence" is the title, by Jeff Hawkins, who - as a Palm follower, as I am, I knew the name immediately. And I can't wait to get his new Palm gizmo, the Foleo, because I think for me that'll be a really nice add-on to my Treo. But I've been a Palm user forever. Jeff is the inventor of Graffiti, the designer of the whole family of Palm things. He then created Handspring, where of course the Treo was born. And then that all merged...

Leo: And you love your Treo, too.

Steve: Love my Treo. Okay. So it turns out that Jeff Hawkins has a strong - has always had for his whole life a strong interest in brain function, how the brain works. Well, nobody knows it, but I have always had that, too. I mean, back in high school I read everything I could about neurons and synapses and axons and all that. And so what gives me goose bumps is that here he is, basically an engineer, a computer guy, who's been successful in Silicon Valley, who has now founded a company, I think it's Numenta, to work on creating intelligent machines. And so this book - and the reason I'm talking about it is I can, even though I'm not finished with it, I'm about I think maybe somewhere between a third and a half way through, what I've read already allows me to commend it to our listeners. It is just riveting. So if anyone is interested in sort of having a contemporary, easy to read, really engaging look at what we currently know about intelligence and how our brain functions at the cellular level, and from Jeff's standpoint, you know, why it is that AI has failed, how things function - as I was saying to you, Leo, before we began recording, I also don't want to talk about it too much because I don't want to be a spoiler for what the book has in it. But I just - I really, really like the book.

So it's funny, too, because I have it in both Sony eReader and in Palm eBook. And I have a physical copy that I got because I have a friend actually who's my second employee at GRC who's a gifted hardware engineer and software guy who ended up leaving because I just didn't have enough to keep him busy. And I'm glad because I kicked him out of the nest. And I was the first job he ever had. And he and I had actually worked together years before on the light pen stuff. He did some light pen work for me. And anyway, I'm going to have lunch with him and give him this book and probably seriously damage his life.

Leo: Why is that?

Steve: Because he's going to read this, and he's going to have a hard time remembering what it was he was doing before he started reading this book. I mean, I know him...

Leo: Is he in this field?

Steve: Yeah, yeah, yeah. Actually he went into gaming, where he's won awards and started companies and sold them and done a lot. His name is Steve Ranck. He's just one of the brightest, sharpest, neatest engineers I've ever known. And Steve will, I mean, it's going to be hard for him not to have to go build an intelligence brain after...

Leo: Good, good.

Steve: Oh, I know, I know. And in fact that's sort of what Jeff is promoting and soliciting.

Leo: Yeah, you get to the end of the book, one of the things he says at the epilogue is, if you are a high school student reading this, please make this your field. This is where we want to go. We need this.

Steve: Yeah. The book is just so full of stuff. One of the cool things is, I mean, we all have a brain, pretty much.

Leo: Some of us. Some don't.

Steve: And there's enough new stuff in here that as I'm reading it I'm thinking, oh, yeah, I never really thought about it that way, but that's exactly right. And again, it's just "On Intelligence" is the book. You can get it, as we know from you, Leo, in Audible format if you want to listen to it. You can get it in Sony Reader format, if you want to play with the E Ink reader that you and I are both using, or in Palm, or eReader. You can read it on Windows because of course there is eReader for Windows. And the physical paper format is also accessible. I should mention that I got it also back on my Palm because I wanted to try falling back from the Sony to the Palm that I was using so extensively for years, and I decided I can't.

Leo: Oh, interesting. You're spoiled

Steve: Yes. Well, first I picked it up on the Sony. And I found myself frustrated by the low contrast of that screen.

Leo: I know, I know.

Steve: And that's the problem with...

Leo: They've really got to do something about that.

Steve: Yeah, that's the problem. And it's like, gosh darn it, I mean, the Palm screen is just vivid black and white, and of course you have a backlight, so it's really bright white. Because I read inverted. I read white text on a black screen on the Palm and love that. But the problem is now I can't have a smaller screen. The E Ink, the Sony screen is so much nicer and more page size that it's even worth tolerating the lower resolution, I'm sorry, the lower contrast of that medium.

Leo: The contrast is bugging me, though. I have to agree with you. I can't wait till they get a paper white screen. Then that's a product that I would, I mean, already, right, it's a good product, and I've been reading a lot of books on it. But I just - my eyes as I get older, I need more contrast, I really do.

Steve: Well, it's not just your age, Leo. Well, actually I guess maybe we're the same age, so...

Leo: We don't know, we're the same age, yeah. So maybe younger people don't have this

issue. I don't know.

Steve: I did notice also, when I went back to Sony to poke around to see whether "On Intelligence" was available from the Sony Connect site, that the price has fallen on the eReader. I think it was three something, and it's now 299.

Leo: Maybe they've got another one coming.

Steve: It might have been 349. Anyway, it has had a price reduction, which to me sort of speaks of, oh, maybe they weren't selling so well.

Leo: Yeah. No, I'm sure they're not, despite how much we've talked about it. It's too expensive. It's not quite the technology it needs to be yet. But we're early adopters, you know.

Steve: Okay. Mailbag. I got two pieces of interestingly related mail. Remember Justin Gerard, the neat 13-year-old kid who had somebody from an outside service come in to fix his computer and discovered that they were using SpinRite. And I don't remember if he was going to try to get his dad to buy him a copy or what. But I found a note from him saying, "Hey, Steve, I found out that Nintendo has partnered with Astaro to provide..."

Leo: What?

Steve: Yup, "...to provide optional content filtering via a proxy for the Nintendo DS web browser. Just thought you might like to know." And apparently Justin, by the way, has a podcast called Gamer's Edge podcast.

Leo: Oh, I know that podcast. Okay.

Steve: Well, that's Justin, yeah.

Leo: I'll be darned. All right.

Steve: And then we also got a note from Terry, it looks like Terry Sheltra of Charlottesville, who writes, "I just thought I would share something interesting with the two of you about the new Nintendo DS browser. Included in the package is an insert that offers content filtering provided by none other than Astaro, free of charge. Since they're a big sponsor for your podcast, I thought I'd share the news with you."

Leo: So that wouldn't mean that they put Astaro on the DS.

Steve: No, it would mean that somewhere there's a proxy server that Nintendo is offering that people can route their Nintendo DS web browsers through, and Astaro is the content filtering and protection technology that Nintendo chose.

Leo: Makes perfect sense. That's what I'd choose. Of course I'm a little biased. All right. So that's the mailbag out of the way. Remember again...

Steve: Almost. I've got one last little blurb here. A David Lee, who's an MCP Small Business Specialist in Ottawa, Canada, he wrote to me with a little fun blurb about SpinRite. And I bring it up because he asked two questions, one of which I hear a lot, so I wanted to answer it on behalf of our listeners who are now SpinRite users in case they encounter this. He says, well, his subject was, "Hello, Steve and Leo, another testimonial to hitch to your belt." And he said, "I've been an owner of SpinRite for about a year now and have used it a few times to improve performance on my machines, but until yesterday had not had any catastrophic failures from which to recover.

"Enter the BSOD," you know, the Blue Screen of Death. "Upon rebooting after installing Adobe 8.1 Reader update, my laptop provided me with the gut-wrenching 'unbootable drive' message. Neither safe mode nor last good configuration boot options would work, either. Knowing I had SpinRite in my 'back pocket,' however, gave me some immediate comfort in the knowledge there was a better than good chance it would get me back up and running. I ran it on Level 2, and 45 minutes later (on my one-year-old Fujitsu 80-gig 2.5-inch drive) it was stuck on an obviously bad area of the drive. With a couple of red U's on the screen, I had confirmation the issue was hardware related, as I had suspected. I left it to run overnight and am writing to you this message from my now working laptop." So, success, thank you. And then he says, "Two questions for you. What is the best practice for replacing drives that exhibit unrecovered data errors?"

Leo: That's a great question because we talk about recovering these drives, and everybody says, oh, we're working along, and it's just fine. But it always makes me queasy. If you had a problem before, should you keep using that drive?

Steve: Exactly. He says, "In your SpinRite episode on Security Now! this was not addressed, i.e., should a drive that has had these types of errors be replaced ASAP? Or with the sectors now marked as bad, is it likely safe to continue using for the foreseeable future? Anticipating a couched response to this," he says...

Leo: Couched? Couched?

Steve: And there's a little smiley face here, "...what would you do if it was your hard drive?" he says.

Leo: Okay, no couching, Steve. I want you to say it out. Sing it out.

Steve: Well, there are a couple of reasons that SpinRite will give you a red U. And we can divide it into two categories. First of all, the U means that no matter what SpinRite tried to do - and, I mean, it will sit there and crunch on a sector for a long time - it was never able to completely recover the sector's data. Now, that can be a defective sector, which should normally not surface on the drive; or it could be one that was miswritten, for example, if the drive lost power during a write. Then you will cause an unrecoverable sector to be created...

Leo: Because is it a head crash or...

Steve: Well, no, it's not a head crash. It's just that it never had a chance to finish writing the sector and update the error correction code, the ECC, which is sort of like a very powerful checksum. And so it just sort of half wrote the sector. And so when you then run SpinRite over it, no matter what SpinRite tries, it just cannot get this sector recovered. It says, look, there's nothing I can do to tell you what this data is supposed to be because in this case it was only half written. So it doesn't necessarily mean that the drive is having a problem.

Now, another non-power failure event can occur, which is that in general defects grow over time. And as anyone who's looked at the videos that you and I and that I and Patrick Norton made, where I was on yours and Screensavers TV shows showing SpinRite work, drives are doing data recovery sort of silently all the time. So-called Error Correction Code is being employed constantly because the data densities have gotten so high on our drives that it's no longer the case that sectors read perfectly. So what happens is the drive is making corrections just because the density is so high. And the drive is watching how long the error bursts are, that is, if it's a few bits, the drive's like, okay, fine, this sector's still okay. But over time these tend to grow.

And so what happens is, at a certain point a threshold is reached where the drive becomes uncomfortable with its continuing future ability to perform on-the-fly correction of this sector. And so what it'll do at that point is it will get a correct read, that is, correct the errors in a sector, and autonomously remove that sector from service, swapping a spare into its place, and then rewrite the data that it correctly read back onto the swapped-in spare sector. So what can happen, if you don't allow the drive to see given sectors often enough, you haven't given the drive the chance to recognize that some sectors are approaching its tolerance of its ability to correct them.

Which is one of the reasons that just running SpinRite at Level 2, which is a relatively quick read pass, essentially what that does is that forces the drive to read all of its sectors. And even though SpinRite won't show you that it did anything good - and this is one of the sort of the dilemmas, the mixed blessings of using SpinRite, is people will run it all the time, and they'll go, well, you know, I've never had a drive give me any problems, and I run SpinRite every few months, but I don't know that it's really doing any good. Well, one of the reasons they've never had a problem is that SpinRite is actually letting the drive assess its own sectors and swap them out before they become problematical. So if someone didn't do that, they might find that SpinRite would show them a red, uncorrectable sector because the sector had gone too long without being read, so the drive wasn't able to say, whoops, this is getting to be problematical. I can still read it, but barely. So while I can, and while I can read it and correct it, I'm going to swap in something that's in better shape. So that's all sort of going on behind the scenes.

And then the third and final problem is drives can run out of spares. And at that point, if you actually have a defect which is not readable, and there's no spares left for the drive to use, it has no choice but to just say, sorry, here's a problem sector, and I'm out of spares. And that does happen. So at that point you certainly would want to say bye-bye to that drive.

Leo: Okay. But how can you tell that?

Steve: Well, running SpinRite a second time after you've had - see, once SpinRite gives up on a sector, marks it has uncorrectable, shows it that way, it will then rewrite the sector with whatever data it was able to recover. That's why I don't give up easily. I try 2,000 times - well, I in the guise of SpinRite - 2,000 times, and use every trick in the book, moving the head different distances in each direction and then coming back at it so that I'm arriving in slightly different positions, do all kinds of things to really, really try to read that sector. When I finally can't, after 2,000 attempts, I will rewrite the sector with what I was able to read. And that process fixes its unreadability and then allows the drive to maintain it from there on. But if you then run SpinRite a second time, and you've still got a problem, that means the drive was not able to even correct from a rewritten, correctly written sector, and it's time to take the drive

out of use before it really goes belly up.

Leo: So you didn't couch too bad on that one. The bottom line is that there is some slack in all drives, and some room for error. But when it gets to the end of that rope, that's when you have to replace the drive.

Steve: Yeah, actually I would say a lot of slack and room for a lot of error.

Leo: It's normal. There's no drive made that doesn't have some problems.

Steve: Not anymore, unfortunately. And then his second question, which I got a kick out of, was "Why hasn't anyone bought your company?"

Leo: Security Now!, you mean?

Steve: No, no, bought GRC or bought SpinRite. And he says, "A la the old Chrysler commercials, 'I was so impressed, I bought the company.'"

Leo: I bet you've had offers.

Steve: Well, Peter Norton offered to buy SpinRite shortly after I created it, about a year and a half later, I think. He said, Steve, I've got the Norton Utilities. Everybody I talk to just wants me to add SpinRite capability to our stuff. I want to buy it from you. And the best thing I ever said to Peter was no thanks, Pete.

Leo: This is your life's work.

Steve: Yeah, so far.

Leo: Can't sell your life's work.

Steve: So far.

Leo: On the other hand, everybody has a price. If somebody came along and offered you a billion dollars, you might take it.

Steve: Yeah, yeah. I'd still do the podcast with you, however, Leo.

Leo: I would hope so. Because at that point you could do it from your yacht. In the Caribbean.

Steve: Exactly. And I'd create my own satellite network so that we had low latency.

Leo: All right. Let's get to the meat and potatoes of our discussion today.

Steve: I forgot what we're talking about.

Leo: Are you human.

Steve: Oh, that's right, that's right.

Leo: Are you human.

Steve: What I love about this is that this is a type of authentication other than what we've talked about before. Of course we've really clearly covered the issue of are you a specific human. That is, you know, the multifactor authentication like what we were talking about earlier with PayPal and so forth was which human are you. And so what I like about this is that this is not who you are. This is are you a who.

Leo: Okay. Are you a who?

Steve: Are you a who, not...

Leo: Sounds like a Dr. Seuss story somehow.

Steve: Exactly. It's not who you are, it's are you a who.

Leo: And that's important because...

Steve: Well, what we've had, of course, is the rise of the Internet robots. Back in the early days of the Internet there were just people using web browsers and using services and things. But what began to happen was we introduced - probably I guess the first real instance was free email accounts. It became clear that it was possible to do browser-based email, where companies like Yahoo!, who may have been the pioneer of web mail, they said, hey, you can create an account, and it'll be susiejones@yahoo.com or whatever, just to make up a name. I hope Susie forgives me for using her email address.

Leo: She's probably pretty used to it by now, with a name like that.

Steve: And so everyone said, hey, this is cool, then I don't have to be at my computer, I don't have to use an email client. Any web browser anywhere I can log onto my email account and check my mail and send stuff. Well, the bad guys, the spammers said yes, and so can we. We can create endless email accounts at Yahoo! with random names or made-up names or names composed of words in dictionaries or whatever, and use it to send our spam because who would

know that we were spam. And that way, you know, the beauty is - this was back in the early days where we had the blacklisting of spam servers. So the problem was those servers that were blacklisted could no longer send spam with sufficient reliability because they were becoming known. But by using Yahoo! sort of as their third party, as their intermediary, using the web interface, they were able to cause Yahoo! to be a spam forwarder, and so that's what happened.

So, and then of course even more recently now with all of this Web 2.0 stuff, now there's so much more, this whole notion, and we've talked about it with regard, for example, to cross-site scripting vulnerabilities and various problems associated with accepting input from users. Now the web is much more bidirectional than ever before with Facebook and MySpace and all the blogging that's going on. And one of the newest trends you see are articles online that then want feedback from readers. They have posting comments at the end of articles. And so what's happened then is that, if high-profile sites like NBC and ABC and CNN and so forth, that is, sites that are highly ranked in search engines, spammers will now - that is to say, website spammers will want to put their own links into blogs and into comments because then search engines will say, oh, look, here's a highly ranked site that has a reference to this other site, which turns out to be a yucky site that people really don't want to know about. But that's a way of them elevating themselves in search engine rankings.

So it's the same story as, like, virus/antivirus, malware/antimalware, spyware/antispyware. There are useful services that have unfortunately been abused by people on the dark side who have said, hey, we can pretend to be human and abuse the system in various ways. So...

Leo: But they can't prove that they're human if you can come up with a scheme.

Steve: Well, yes. What's really interesting is that the first time I encountered this notion of needing to prove that I'm human, it's like, oh, what an interesting, kind of cool problem. And your first thought, my first thought as a computer guy, and any of us are to some degree, well, to probably a complete degree computer users. Any user sort of thinks, huh, how would I solve this problem? And at first it doesn't seem to be such a hard problem. It's like, well, it's got to be easy to figure out the difference between a computer and a human. But it turns out it's less easy than people might think because so much of the sorts of things that are feasible to do, computers can be programmed to get around.

So this notion of some way to solve this problem turns out to actually have a deep history, which is sort of interesting. There was something known as the Turing Test, T-u-r-i-n-g, which was originally proposed by an early researcher in computers named Alan Turing, who was also the father in some senses of modern computing, or at least of algorithms.

Leo: He also helped crack Enigma, the German codebreaking device, World War II.

Steve: Yes, exactly. He worked at Bletchley Place, or is it Plaza or Place?

Leo: Place.

Steve: Bletchley Place over in England, and was involved in the codebreaking of the German Enigma code during World War II. And he was, well, first of all, he was recognized at a young age as being a genius.

Leo: Brilliant guy. Brilliant guy, yeah.

Steve: Yes. He really is regarded as that. He tackled the question of computability, that was, what did it mean for something to be computable, and that is to have, like, an algorithmic solution. And in fact he formally laid down this notion of an algorithm. And the formalization of it took the form of something called a "Turing Machine," which he formally proposed and laid out in a paper that he wrote early on.

Leo: Couldn't be built at the time, in the '40s.

Steve: Well, and in fact it's more of a theoretical machine. I mean, people have built them for fun. But the idea of a Turing machine is that it, sort of using the jargon...

Leo: Bletchley Park, by the way.

Steve: Oh, that's right, Park, right. Using the concepts at the time, the idea of Alan's Turing Machine was that you'd have a paper tape which was theoretically infinite in length. It had a starting place, but it never ended because, you know, this thing, in order to do useful work, might have to have a very long tape. Because the idea was that it was a series of very simple steps. Basically you had a read head that was positioned at some location on the tape. And each cell in the tape could contain only a one or a zero. So he had this notion of binaryness back then. And based on - oh, and the machine also had some state. So it had a well-defined state which was not specified, but it was just some number of bits or something that defined the current state. And the way he defined it was you had the current position of the head, which was able to read or write to a one or a zero to the cell, and then cause the tape to move to the left or the right and change the state, that is, update the state to state plus one. And basically that was the definition of this machine. And he managed, through a series of very careful mathematical proofs, to show that this machine could solve any problem that any computer could solve. And in fact there was something known as the...

Leo: At the time a computer was a human.

Steve: Exactly.

Leo: Was a woman, usually, who was doing computations. So he's talking about a machine that replaces a human computer, basically.

Steve: Right. And then there was this notion of the UTM, the Universal Turing Machine, where he said, okay, you could define a machine on the tape, which on the tape would then be followed by a problem. And so this created this notion of a Universal Turing Machine because you could then - you could use it as, like, a universal emulator for any other machine. So he basically said, okay, this is how computers can be simplified down to this.

Well, amid all of this, actually toward the end of his life, which wasn't very long, unfortunately, he only lived - didn't quite make it to the age of 42. So he was quite prolific during his first 41 years. He ended up saying, okay, thinking about intelligence, how can we determine if a machine is intelligent? How do we define intelligence? And he had a very 1940s, early 1950s definition. He said - and this was the so-called "Turing Test." He said, if you put a human in one

room and a machine in another, and you ask them both questions, that is, a human being asks both of them questions but doesn't know which room contains the human and which room contains the machine, that is to say, the AI, the artificial intelligence, if the person asking the questions cannot reliably determine which room contains which entity, then that is said to pass the Turing Test for determining whether this is a machine or a human. So again, it's a sort of a thought experiment. It's a simple formulation for asking the question. But that's deep in the history of computing.

Leo: Just some little footnotes on that, in "On Intelligence" he'll talk about - you haven't gotten there yet, but he talks about the Turing Test and why it essentially will never be solved. That's kind of in some ways the premise of "On Intelligence."

Steve: Well, I think he also talks about John Searle's Chinese room thought experiment. Anyway, there's a bunch of stuff in this book that I really recommend to our listeners.

Leo: A second footnote, before you go on. It's appropriate to mention, and it's very sad, that he in fact committed suicide after being prosecuted for being gay.

Steve: I know.

Leo: It's just a great tragedy. I won't belabor it, but read the article in Wikipedia about Alan Turing's life. And there's some excellent books on Turing. And he actually appears in "Cryptonomicon," another favorite book of ours.

Steve: Yes, he does. Right, I forgot about that.

Leo: Yeah, Bletchley Park is a big part of that whole book. But anyway, so that Turing Test is the first case of a human-machine challenge, I guess.

Steve: Well, it is. And it bears on what we're talking about because the rather lengthy acronym for the whole class of currently most popular approaches to differentiating people from computers, that is, who you are or are you a who, involves this Turing Test phrase. What happened is, about seven years ago, in 2000, Carnegie Mellon University, a researcher there, Luis von Ahn - and for Elaine's sake, who's transcribing this, von Ahn is two words, v-o-n, and the second word is A-h-n. He got serious about CAPTCHAs. And actually that's their acronym. And believe it or not, Carnegie Mellon has a trademark on the word CAPTCHA.

Leo: Really, it's an acronym? C-A-P-T-C-H-A. I always thought it was just "capture" mispronounced.

Steve: Well, get this. CAPTCHA stands for Completely Automated Public Turing Test to tell Computers and Humans Apart.

Leo: I'll be danged. So Alan Turing is memorialized in the CAPTCHA.

Steve: Yes. And now people know, if anyone has the memory to remember Completely

Automated Public Turing Test to tell Computers and Humans Apart, now you know what a Turing Test is and who Turing was who came up with this test. So that's what CAPTCHA stands for. And the university, CMU, has a trademark which they're obviously not being jealous about. I'm sure they trademarked it so that, well, because they had come up with it. They thought it was cool, and they wanted some credit for it, but not to prevent others from using the word because it's widely used, and no one from CMU seems to be causing any upset about that. So the most familiar CAPTCHAs, and I would imagine by this point everyone around, that is to say everyone listening to this podcast, has encountered some. You and I have in fact talked about them from time to time briefly in prior podcasts, Leo, where they're these wacky-looking images where you are asked to basically decipher a warped and deliberately obscured chunk of text or letters and numbers or something, you're asked basically to solve some sort of problem, the idea being that this is something that would be easy for humans and hard for computers. Well, it turns out that computers have gotten so good that, in order for it to be hard for computers, unfortunately more often than not it also has to be hard for people.

Leo: Right, unfortunately, yeah.

Steve: And, I mean, I have encountered CAPTCHAs, visual CAPTCHAs, that I have not been able to properly enter the proper code for, which is really annoying.

Leo: I do it all the time. You know, I have a lot of difficulty with Digg's. I just sometimes cannot read Digg's at all. They invoke the CAPTCHA if you try to enter your password and fail once. Then they want to make sure you're not a machine.

Steve: Okay, that sort of makes sense. In fact, one of the things I want to talk about here is this notion of the cost of CAPTCHAs because it's very interesting. I want to make sure that people know that the show notes for this episode has a bunch of links. Wikipedia has an excellent page on CAPTCHAs, lots of coverage, lots of good references. And on the show notes for this episode, Episode 101, I found some interesting CAPTCHA-cracking sites that have really good technology for and pictures of many popular CAPTCHAs, showing basically what it is about them that isn't hard enough. And there are some CAPTCHAs that have solved at 100 percent, some at 88 or 89, meaning that algorithms have been created which are specifically designed to crack the visual complexity that people have put into these things in order to, again, differentiate between human and computer.

The problem, of course, is that - as I said, this is the best set of puzzles or problems that we've so far been able to come up with. But the problem is computers have gotten so powerful and so good that there's a very fuzzy line between computer and human. Looking at these CAPTCHA-cracking sites, and specifically at the complaints the authors of these crackers have, they say, for example, using a constant font or even a small number of fonts allowed them to do a better job of cracking the CAPTCHA.

Leo: Interesting, huh.

Steve: Or using all aligned or almost aligned characters allowed them to sort of analyze the characters and lock on to the relative alignment. And that was an aid to them. Or using constant character position. There are some CAPTCHA generators that produce images where the characters are, like, scattered around, but they're always scattered around in exactly the same location from one CAPTCHA to the next. So somebody trying to deliberately crack a CAPTCHA-protected site would take advantage of the fact that, yeah, the characters are all kind of "cattywumpus," but they're always in the same "cattywumpus" position. So once the author of a cracker saw that by looking at several CAPTCHAs being generated by that site, they'd go,

oh, well, sure.

And what's interesting is there's a perfect example of a mistake by the CAPTCHA maker because here you've made it really more difficult for the human because we're just encountering that CAPTCHA probably once. But here a spammer has - essentially what you've done is you've made it harder for the human by scattering the characters around; but someone cracking the CAPTCHA, writing code to crack it, once they recognize that the characters are always in exactly the same position, with the same orientation, well, you've made it no more difficult for the code, while keeping it difficult for the human. So that was a bad idea not to move them around. Also constant horizontal position, constant character rotation, no deformation, non-textured background, use of constant colors or weak colors - that's easier then for the cracking author to lock onto - or no overall perturbation of the image. So essentially what this means is that everything that is being done to make these things difficult for us does to some varying degree make it difficult for a program. But unfortunately we're seeing now that CAPTCHAs are to some degree being able to be broken.

Now, other sites have analyzed what it is that is the most difficult for computers to solve, that is, in terms of cracking these images. And it turns out that dividing the individual glyphs, you know, "glyph" is the font term for an individual character, essentially. So dividing these images into individual characters is something that we do pretty well. But that's one of the problems that computers have. And so, for example, running lines through the characters really does bump the level of difficulty for automated recognition of the characters up rather substantially. And so one of the things you will see in the most modern CAPTCHA technologies are words or phrases or sequences of letters and numbers which end up being obscured with lines running through them, and maybe not so much other wacky stuff that makes it really difficult for people. I mean, again, Leo, I've run across CAPTCHAs where I'm thinking, okay, am I going to be punished if I get this wrong? Because I don't know if this is a Q or a bent-over R. I mean...

> **Leo:** But you make an interesting point, that sometimes what's hard for a human is actually not hard for a machine.

Steve: Right, exactly. It's not always the case that what some clever programmer has actually done something that is making it more difficult for a machine because what you're generally having, and this is a function of value, which is one of the key points I want to make today also, is the CAPTCHA protecting Yahoo!, because Yahoo! has a high cracking value, you might very well have somebody who is willing to spend a great deal of time and effort writing a Yahoo! CAPTCHA-cracking technology. Whereas some obscure, off-the-beaten-path blogging site, which has never had a problem with malicious posting to it, well, putting an extreme CAPTCHA on that site is...

> **Leo:** Is extreme.

Steve: Is extreme, exactly. Here you're hoping somebody is going to put a comment on your blog posting. Well, raising the bar up too high for that poor human who is like, okay, do I have to stand on my head in order to decrypt this? What is this CAPTCHA trying to say?

> **Leo:** I have to say, though, it's so frustrating when you get spammed that sometimes you just say, screw it, I don't care if it's hard for humans, I just - I'm not going to let the spammer in.

Steve: Right.

Leo: You wouldn't believe, I actually have turned off registrations on TWiT.tv because these guys will go to the extreme trouble of creating an email account, signing up, waiting for me to send them the password via the verified email account, and then spamming. So it's very, you know, it's a lot of effort they're putting in, compared to how hard it is for me to block them. It's not very hard for me to block them, I just have to constantly do it. And I finally just gave up, they were so...

Steve: Persistent.

Leo: ...relentless, yeah. So I understand why people might put really difficult CAPTCHAs.

Steve: Well, there have been some very, very clever hacks. And I think one of my very favorite most clever hacks, and this was done as a proof-of-concept actually by somebody at CMU who was working sort of on the opposition team, I mean, he was on the side of CMU, and these have been major CAPTCHA innovators over the years, and we're going to talk about a recent very cool concept that they came out with. But in any environment you want somebody trying to break your solution. And in the academic environment there were some people who said, I know how to break the whole CAPTCHA system.

What they did was they created, or maybe they just borrowed or used, but anyway, there was an adult website that had very high traffic. And what they did was they created a system which would - imagine going to Yahoo! and wanting to create an account. You're presented with a CAPTCHA. On the fly, this bot took that CAPTCHA and stuck it on the log-in page on the adult website, where a human would solve the CAPTCHA problem.

Leo: Oh, no.

Steve: So the bot would then take the human solution and feed it back to Yahoo!, thus solving Yahoo!'s original CAPTCHA problem. Isn't that just the cleverest thing? I just...

Leo: So they basically used unwitting human helpers by just borrowing Yahoo!'s CAPTCHAs and keeping track of them.

Steve: Exactly. There was enough traffic on the site, that is, on the adult website, that no CAPTCHA would remain unsolved for long. And so it fit within the window of time that Yahoo! was allowing that CAPTCHA to be valid. So on the fly, Yahoo! CAPTCHAs would be - or, you know, I would imagine the way it would work is this. You would actually be - I'm just thinking of this on the fly here. You would actually have your bot on the adult website. Oh, no, I was going to say you would wait till they were presented with a puzzle. Or maybe, yes, you go the adult website, and it says "click here to log in."

Leo: Right. And meanwhile it goes off to Yahoo! and gets a CAPTCHA.

Steve: Exactly. When the person on the adult website clicks here to log in, it's about to present him with a CAPTCHA. So the bot goes to Yahoo!, clicks to create an account, receives the CAPTCHA, forwards it to the adult site, that then presents that CAPTCHA to the human, who solves the CAPTCHA problem. And then the bot turns around, solves the problem on Yahoo!, and is able to create an email account. So I just, you know, that's super clever. And I just

thought that was a neat solution.

Leo: It's also depressing because it really means it's hopeless. I don't know how you'd get around that. I mean...

Steve: Yeah, exactly, now what are we going to do?

Leo: Now what do we do, yeah. Because you're essentially, I mean, you can't get around the fact that they can enlist humans, as long as they're willing to enlist humans to do this.

Steve: Yes, exactly. And that, by the way, is known in the CAPTCHA community as the so-called "relay attack," where you're relaying the CAPTCHA through a site where you've got enough traffic in order to generate a useful number of CAPTCHA solution events on the fly. And again, what are you going to do about it?

Now, CAPTCHA is a little bit controversial, also. And I have a link on our show notes to the W3C's, you know, the W3 Consortium's anti-CAPTCHA page. They just don't like CAPTCHA because they feel that it very unfairly discriminates against people who have any kind of disabilities. And they're all about, and they want the web to be all about, being a nondiscriminatory system. And they make the point that even if you have - and you talked about this before, Leo, the notion of an audio CAPTCHA, that is, the idea being your main CAPTCHA is a visual CAPTCHA, where you've got to figure out what this phrase or expression is. But for blind people, of course, they're not going to be able to do that. So you have a backup solution, which is click on this to hear some...

Leo: I'm going to play - this is the one - this is Digg's.

AUDIO CAPTCHA: J, P, N.

Leo: The truth was, that wasn't - I wouldn't think a machine would have that much trouble.

AUDIO CAPTCHA: F, F, J, P, N.

Leo: It's actually easier than doing it looking at it. I'm going to use the listen to it from now on.

Steve: It actually is. Now...

Leo: Now, I played the Carnegie Mellon for you before. You want to hear that one?

Steve: Sure.

Leo: This is the audio - attempt to make an audio version of a CAPTCHA.

AUDIO CAPTCHA: [Tone], 4, 1, 6.

Leo: That would be tough for a machine.

Steve: Was that 416?

Leo: I guess.

Steve: I think. And, see, here's the problem, is that we also have speech recognition. I mean, we've been working on that for a long time in the computer community, in the same way that we've been working on optical character recognition and vision technologies. So unfortunately the audio backup for the visual CAPTCHA also has to be deliberately obscured in order to make it, again, difficult for a computer. And unfortunately there just isn't enough difference today between computer recognition of audio and human recognition of audio for this to be a really great solution. But the W3C page makes the point that even doing this discriminates because it's a privacy problem. Now you're forcing blind people, people who are not sighted, to declare themselves as such on a site. And you could argue that that is some loss of privacy because they're not able to solve the visual CAPTCHA, they're being forced to solve the auditory CAPTCHA. Anyway, the...

Leo: I'm not sure, I think that's going a little - that's a little scrupulous, but...

Steve: I know. But the W3C page does - it's very anti-CAPTCHA, and it makes a number of points, one of which...

Leo: I would be more aggressive in defending CAPTCHA; but it sounds like, from what you've just described, it doesn't work that well anyway. Or it doesn't necessarily work that well. So maybe it's not worth defending.

Steve: Well, there are problems with it. Now, one of the coolest things you turned me onto a couple weeks ago, you just sent me a link in email, Leo, and you said "Isn't this neat." And it is just, again, I love clever stuff, and I love sharing cleverness with people. And this of course is known as reCAPTCHA. And this again is from our same guy, von Ahn, at CMU. He recognized that there's a huge effort underway to digitize books. We are wanting to get books into digital form, rather than just scanning them and then, like, presenting scanned pages. People can read scanned pages, more or less. The problem is that a scanned page is a huge amount of data compared to a typewritten page. And if we actually have it converted to text, then it could be eBook, it can be - you can use text-to-speech in order to automate the process of reading the book audibly, I mean, it's just so much cooler to have these books in electronic form, I mean, in textual, you know, ASCII-style electronic form as opposed to scanned images. But many books were published, I'm sure it's probably most books were published before we had the technology to do it all electronically. So we just - there is no original DOC file or TXT file to back up these books. So here we've got this huge archive of scanned images. Well, the cleverness of reCAPTCHA is of using single-word snippets from scanned texts and presenting those to users. So in the process of solving the CAPTCHA puzzle, you are part of the solution of digitizing these books.

Leo: I just think it's so elegant. And by the way, I use that, if people want to send me email - now, here's an example of why you might want to use a CAPTCHA. I'm not going to put an email link on my front page, although at this point it's pretty moot since I think I'm on every spam mailing list in the world. But I thought, well, why make it any worse? So I do have a contact link. But in order to get the email address you have to fill out a CAPTCHA. And I'm using the CMU CAPTCHA. So if you go to Leoville.com or Techguylabs.com, and you click "Contact Leo," you'll get a CAPTCHA. And in the process - so do you want to describe how it works? It's very cool. There's two words you get.

Steve: Yes. What they do is they show you two words. These were both...

Leo: They're both English words.

Steve: They're both English words. They were also originally flagged as unrecognizable by the OCR software.

Leo: So these are books they're trying to scan, but they've had some trouble with.

Steve: Well, these are individual words from books they've tried to scan. One of the cool things about OCR, contemporary OCR software, is it knows if it's not sure. That is, it knows, if it's, like, eh, I really don't feel confident, you know, you now get a confidence factor per word from OCR software, or at least this particular OCR software that's being used. So here's the cool thing. The word starts out as being unrecognizable with high confidence by the best OCR software we have. So that's a good start at something which no one else's OCR software is going to be able to recognize, either. Then it is deliberately warped, and a wavy line is run through it. And as we learned, a wavy line, it now means that instead of having little islands of glyph which are easily identified and then can be chopped up and recognized individually, this turns it into one big blob of black. And it turns out that it really hurts recognition, but not for humans. Humans don't have a problem with that.

And so the idea then is that two words are presented to you, and you're asked to type them both in. Well, you're not the only person who receives this pair of words. Enough people receive different combinations of them that by comparing what people put in, the back end software that is doing all this is able to arrive at a high enough confidence level that, you know, I don't know what their numbers are, but maybe ten people all put in the same text for each of the two words appearing at different times in different contexts and in different pairs that they finally said, okay, we now know for sure what this word is. So not only have you proven yourself to be human, but you've also helped OCR software digitize books.

Leo: And I think it's just so cool.

Steve: I love it.

Leo: In fact, even if I didn't really need it, I would still use it just to do my part; right?

Steve: Yes. And, now, it is estimated that 60 million CAPTCHAs, visual CAPTCHAs, are currently being solved by humans every day.

Leo: Wow.

Steve: And that about ten seconds or so of time is required for the typical CAPTCHA to be looked at and typed in.

Leo: Oh, that's a lot of computing time.

Steve: So ten seconds times 60 million a day is 600 million thought seconds per day, which is to say 166-plus thousand hours. So 166,000 hours are being spent now solving CAPTCHAs.

Leo: Each day.

Steve: Each day. So by using reCAPTCHA, that can be turned into 166,000 hours of people distributed over the entire world, digitizing books.

Leo: I think that's great. I would love to see what the results are so far with this project. I think that's just wonderful.

Steve: So absolutely, I want to commend people to take a look at reCAPTCHA. If you go to CAPTCHA.net, www.captcha.net, it now recommends reCAPTCHA. And reCAPTCHA's just the same with an "re" on the front of it.

Leo: Trivially easy to add to a website. You'll generate it, it'll create a link, and boom, it's done. And by the way, they talk about the pornography attack in the other - this is really interesting how they're doing it. They point to a lot of the information that you've been talking about.

Steve: Right. Now, one other - there are other things that have been done, in general solving - that is to say, have been done as non-visual means, or non-CAPTCHA - well, I guess CAPTCHA actually would encompass everything because it's computer blah blah blah, whatever that acronym was. But so anything that is trying to differentiate a person from a computer. So we have the visual CAPTCHAs we've been talking about. But, for example, you could imagine some sort of puzzle-solving solution. There has been JavaScript created which asks simple, English-language problems, like what is one plus one, as a trivial example. The problem is, again, it wouldn't be hard to cause a computer to have, you know, there would be a limited enough vocabulary of permutations of questions that different numbers would get plugged into that you could write some code that would understand that limited subset of questions and be able to answer them. So that's not very exciting.

Then, and you mentioned this, I think, last week or the week before, there was this interesting notion of something called "KittenAuth." KittenAuth was created a couple years ago. The idea was it would show you a grid of photos, some of which were kittens, and you'd have to select those which were kittens and then click okay, the idea again being that only a person would be able to actually recognize what this was a photo of, and whether or not it was a kitten, and be able to answer it correctly. There again, the problem is you would have a limited number of pictures. And given somebody who was motivated enough to crack this form of CAPTCHA, they would simply capture all the pictures, flag them as kittens or non-kittens, and then be able to hack right through. So that's not going to work very well.

And the other thing is that CMU comments that that actually isn't a CAPTCHA by their definition. They want it to be a fully automated creation, that is, you're having to take actual photos, and you're going to have a limited database of those. You're not actually synthesizing something from scratch, which is what these visual alphanumeric CAPTCHAs are doing is they're building the image from scratch, and you don't have a limited domain of images from which to choose or, again, that would be very easy to hack.

So then another thing that has been suggested is somehow involve live operators, where you involve a human in the loop, sort of harkening back to the original Turing Test, where you did have a human making the determination. And then there's a whole domain of what we were talking about when we were talking about multifactor authentication, where you have specific ID authentication, that is, you're no longer determining whether the person, an anonymous person, is a person or a human. But you're saying - you're using single sign-on. You're using authentication like OpenID or CardSpace or something to say, okay, this is exactly who you are. Maybe you're using public key infrastructure for doing, for example, using certificate authentication or biometrics. So it's like, okay, computers don't have thumbprints. Although you could certainly imagine that computers could just create a fake thumbprint. So those things seem not so useful; but, you know, they've been experimented with. And then there's also simple questions being asked, variations on, like, five words, and the question is which one of these does not have a sail. And the problem again is you're probably going to have a limited number of questions, which after a while all of those can be automated in order to crack the response.

So it turns out that this is a hard problem, like I said. You think right off the bat, the first time you encounter it, it's like, oh, how interesting. Shouldn't it be easy to tell people and humans apart? But over an Internet connection it's not so easy.

Leo: Not so easy. Yeah, very clever.

Steve: And then when you add this notion of a relay attack, where you've got people solving the same problem for a different reason, then it's like, oh, very clever, and now what do we do?

Leo: Yeah. Well, they claim that the reCAPTCHA is not susceptible to that. I haven't read it, so I'm not sure what...

Steve: They've got a bunch of crypto involved, where they use multiple servers, and it's all free, but you get some tags that you use as part of this. And so they are specifically working to liminate - to liminate - to eliminate and limit - I was trying to say eliminate and/or limit that vulnerability.

Leo: Yeah, very interesting. Well, that's CAPTCHAs. That's how to tell if you're a human. Are you human? Hmm.

Steve: Okay, Leo, now you've got to do this. https://www.paypal.com/securitykey.

Leo: Ah, okay.

Steve: See if that comes up for you. I put in PayPal security key into Google. It's the first link that comes up. Because this is just so cool for five bucks. And it just, I mean, $5 to have...

Leo: Oh, there it is, yeah. All right. Order your security key. I don't know why it's not showing up on my front page. But that's great. Oh, I've got to log in. All right. And I'm - so, now, you use this every time you use PayPal now?

Steve: And I forgot to mention, also eBay. It provides security for eBay, as well.

Leo: Yeah, because they're owned by the same - see, five bucks, what a deal. I'm sending it right now.

Steve: I know, $5. I mean, who doesn't want to have one of these cool crypto tokens for $5? Anyway, I just - I wanted to make sure people knew because then, after you authenticate this to your account, you simply add, and you must add, those six digits, the six digits that currently shows, to the end of your password in order to log on to PayPal or eBay. And I'm just so jazzed about this.

Leo: That's just brilliant. Well, we use PayPal heavily. And I am always a little nervous because I keep, you know, I keep all of our receipts from the TWiT Network in there. And I'm almost using it as a banking account because you earn interest on it. So I feel like, well, that's a safe place to keep it. But, boy, we could be wiped out if I weren't careful with the password. So this makes me feel a lot, lot better.

Steve: It is safer now.

Leo: Yeah. Because since our hosting draws directly from that account, all of our overhead pretty much comes directly from that account. So if it were to suddenly go to zero balance, I don't want to think about it.

Steve: I would not want to encourage people to create a bogus PayPal account just to get a $5 token.

Leo: No.

Steve: But you could. And it's very cool.

Leo: Well, you know, if you're going to - here's a way to do it. Make a donation to TWiT. As part of that you can set up an account. You don't have to tie it to your bank account. You can use a credit card. So you don't have to feel like you're somehow giving them something private. I think that's one thing that scares people. Just sign up. Give us a little donation. Help the podcasts to stay afloat. And then your little benefit is you get that $5 security key and be more secure. I like it.

Steve: It's just too cool, Leo.

Leo: I just ordered it.

Steve: Cool.

Leo: All right, Steve. Boy, this is a long episode. We had a lot to say.

Steve: Mailbag is next week. So I want to again say [ADDRESS DELETED DUE TO SPAMBOT DISCOVERY]. That is mail that comes directly to me through no spam filtering or anything else. The web form at the bottom of the Security Now! page at GRC.com sends email to the same address. It does so anonymously unless you provide your name and location and email and so forth, which does allow me to respond when I'm able to, and I like to when I can. But anyone who just wants to do an easier way...