# SECURITY NOW!

**Transcript of Episode #90**

## Multifactor Authentication

**Description:** Steve and Leo discuss the theory and practice of multifactor authentication which uses combinations of "something you know," "something you have," and "something you are" to provide stronger remote authentication than traditional, unreliable single-factor username and password authentication.

High quality  (64 kbps) mp3 audio file URL: http://media.GRC.com/sn/SN-090.mp3
Quarter size (16 kbps) mp3 audio file URL: http://media.GRC.com/sn/sn-090-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 90 for May 3, 2007: Multifactor Authentication.

Here we are already at Episode 90. At this rate, Steve, I think you’re going to catch up with TWiT any minute now.

**Steve Gibson:** I look forward to that day, you know it. One of my own personal benchmarks.

**Leo:** I’m starting to get emails and stuff on our 100th episode of TWiT, you know, people want to give away stuff, and I’m getting gifts in the mail and stuff. It’s kind of neat, I just hope we make it to #100 at this point. It’s getting a little iffy. There’s no question that in ten weeks we’ll be at #100 for Security Now!. There’s just no doubt about it.

**Steve:** Well, and I’ve got some plans for it. I have a special plan for the 100th episode, something I think everyone’s going to get a big kick out of. So, yeah.

**Leo:** Well, the Daily Giz Wiz just did its 300th, and we gave away a bunch of stuff.

**Steve:** See, now, that’s cheating because you do one a day; right?

**Leo:** Yeah. Well, and of course the first podcast I did was the radio show. And that’s up to Episode 350. But the Giz Wiz’ll catch up because we only do two a week of the radio show.

But I started – that was the very first podcast I did. I think it was one of the very first podcasts of any kind. And it was in September, I think, of 2004. Is that right? Yeah. Anyway, enough reminiscing. Time to talk about security. And today we're going to talk about something you call "multifactor authentication."

**Steve:** Yeah. It's something we've never touched on before. We certainly, of course, very early on this series we did some extensive coverage of what are good passwords and good password management habits and so forth. But increasingly what we're seeing is that what's called "single-factor authentication," which is just typically a username and password, is being regarded with increasing skepticism simply due to all the problems that we've talked about with passwords. So multifactor authentication is like the new thing. And I think it'll be fun to talk about it with our listeners.

**Leo:** So any updates from the last couple of weeks, Steve?

**Steve:** Don't have any errata. Although I do have sort of a fun...

**Leo:** You're perfect. You're perfect.

**Steve:** No, certainly not that. But nothing has come to my attention. But I do have sort of a fun, different sort of SpinRite anecdote to share, one that's not about saving a PC. I got a note from someone named Scott Freudenthal in Florida who's a Security Now! listener. And the subject of the note was "SpinRite saves more than just computers, thanks be to goodness," as he put it. He said, "Steve. I've used SpinRite for saving multiple hard drives in the past. I've always been impressed with the capabilities of the product. I always knew that I could count on SpinRite to save the day. I figured you had enough feedback on your basic day in, day out computer recoveries." No, that's not true, everybody. Send as many notes as you want. I love getting them, love reading them.

**Leo:** That's his reward, yeah.

**Steve:** Yeah, exactly. And he said, "So I have never sent you any feedback before. However, I was gone on a vacation recently, and during my absence our TiVo died. Now, we're talking about a real serious situation."

**Leo:** Yeah.

**Steve:** Well, because you can imagine, he was counting on his TiVo to be sucking up all of the programs that he wasn't able to watch while he was on vacation, planning...

**Leo:** And you've never seen anything so scary as somebody going through "Grey's Anatomy" withdrawal. I mean, that's just painful. Painful.

**Steve:** So he says, "I tried tech support when I returned and realized the only solution they were offering was a new TiVo. I vaguely remembered hearing on Security Now! that someone had used it to fix a TiVo drive. So I gave it a shot. Oh, happy day. Our TiVo is back, and harmony has returned to our little family unit. Thank you so much for making such an exceptional product. Keep up the good work and the great information provided on Security Now!. Signed, Ann and Scott Freudenthal in Florida."

**Leo:** That's neat. So now we've heard of people using it on their iPod hard drive and their TiVo hard drive.

**Steve:** Yeah. And I think I did once mention it sort of offhand about it being useful for TiVo. It has saved Greg, my tech support guy Greg's TiVo, a couple of my TiVos several times. Because those things sit there, they tend to run hot, and they just go day in and day out. And the TiVo actually is always running data through the hard drive, so the head is constantly moving back and forth between its recording area and its playback area, even if it's doing nothing. I mean, when you turn it on, you've got that 30-minute buffer delay always sitting there. So TiVo really does work these hard drives pretty hard. And sometimes they just get more than they can handle. So SpinRite is, of course, able to fix them. And I wanted just to renew that fact as a consequence of Scott's note.

**Leo:** It doesn't matter what operating system or what file system is being used on the hard drive?

[Talking simultaneously]

**Steve:** Well, SpinRite will, if it's present. But if it doesn't have a partition table that it recognizes, it goes, okay, well, we'll fix whatever's here, even if we don't know what it is.

**Leo:** Ah, interesting.

**Steve:** And of course it's interesting, too, because on the older Series 1 TiVos like mine, which I've stayed with because they're much more hackable than the newest ones, that's a PowerPC platform that has the bytes swapped in a different order than Intel. So the data recorded on the drive isn't even recognizable as a normal partition table, even though it has one. SpinRite looks for the signature at the end of the partition table, doesn't see the 55AA because on that system it's AA55, and SpinRite goes, okay, well, we'll just fix it even if we don't know what this is.

**Leo:** Interesting, yeah. All right. Let's see. No updates from last week, so let's launch right into it. This is something called multi...

**Steve:** Multifactor authentication.

**Leo:** Okay.

**Steve:** Yeah, now, okay. Let's step back a little bit and talk about authentication sort of generically. You know, back in the "olden days," like Andy of Mayberry days, you know, Opie would go into the drugstore and...

**Leo:** [Whistling theme to "The Andy Griffith Show"]

**Steve:** Exactly. And the pharmacist knew him. And so he'd say, oh, you're here to pick up, you know, Grandma – well, I don't remember what his grandmother's name was, but...

**Leo:** Aunt Bee.

**Steve:** Aunt Bee, exactly. Thank you, Leo.

**Leo:** Opie, go down to the drugstore and get my heart pills.

**Steve:** Exactly. And so there was known person-to-person authentication that you could rely on. And so people knew you, and just being known was real-world authentication. Or, you know, a restaurant you went in all the time, say you forgot your wallet once, it's like, oh my god, I don't have my money with me. Oh, that's fine, Leo, we know you. Come back in an hour or pay for it next time, whatever. So, I mean, in the real physical world we always had authentication.

Well, of course what's happened with the Internet, and to an increasing degree, that kind of real-world authentication is gone. But it's still very important, and in fact you could argue increasingly important, due to the fact that applications of the Internet are expanding, really important for authentication to be present. Now, we've talked about websites authenticating themselves using SSL and certificates, where a web server has a certificate signed by an authority where we trust the signing authority, VeriSign or Equifax or whomever. And so we assume that that signing authority has done its due diligence to verify the identify of anyone it's giving a certificate to. So our web browser trusts that signing authority. And when we get a certificate as we're setting up an SSL, a secure connection to the web server, we get the certificate, we see that it was signed by someone we trust, and so that creates a chain of trust that says, okay, we're going to trust the certificate owner because we trust the signer of that certificate. Well, that's authenticating from the remote end to us.

**Leo:** Single-factor authentication, really.

**Steve:** Well, that's actually, yes, that is a single factor. And in typical ecommerce mode the need, when you're wanting to authenticate yourself to someone else, is going in the other direction. And the classic example is online banking, where you want to make sure you can get to your personalized online banking account page, but nobody else can. And the bank would like to know that you are you. So originally, of course, passwords is what was used because people were kind of – they understood passwords immediately. You know, you have a username and password which you use to log on. Now, as we've talked about in talking about passwords extensively, you know, they're free, that is, they don't cost anything, and they're easy to use. But at the same time they have the downside of you get what you pay for, that is, you know, you didn't pay anything for them.

So you have the problem, as we've talked about, of user management, you know, if a password is easily guessable then that's a problem. If it's really complex, then you're in danger of not being able to remember it. And so what many people do is they'll write them down, and a written down password, of course, is a problem because then anyone else coming along can see it and find it. So that's bad. And we've also talked about the use of many different passwords, not using the same password all the time, but having a bunch of them. Well, having a bunch of them also puts a burden on the user of just having so many passwords. The other problem with a password is it can be deliberately divulged, that is, you could say, oh, you know, if you want to read, you know, my New York Times online magazine subscription or online newspaper subscription, just log on using my password, my username and password.

**Leo:** I've done that many times, yeah.

**Steve:** So it has both the problem of unintentionally being discovered and being deliberately divulged. So essentially there's this issue of accountability and plausible deniability. You know, weak authentication means that users cannot be held accountable. It's like, you know, the dog ate my homework problem, meaning that, say that your boss comes to you and says, hey, you know, you logged onto the server and were doing some bad things. But, you know, if all you're using is a weakly authenticatable password, you could say, oh, somebody must have stolen it. I didn't do that. So there isn't strong accountability there because, again, they're easy to use, but they have a problem.

So all of this discussion so far, the whole issue of passwords, is known in the security trade as "single-factor authentication," that is, you're only being asked to provide a single aspect for authentication. Well, generalizing factors, it's very cool, sort of the jargon that's been developed, I get a kick out of this because passwords are something you know. The next two types of factors are something you have, and something you are.

**Leo:** Wait a minute. Let me understand this. A password is something you know. Okay, that makes sense. So that's the first of a multifactor authentication scheme would be that.

**Steve:** Or I would say it's the most common factor in a multifactor authentication scheme is something you know.

**Leo:** Something you know. The other would be something you have, like, say, a Smartcard.

**Steve:** Or some sort of a token.

**Leo:** A token. And then something...

**Steve:** Something you are, which is to say your fingerprint, or the pattern of blood vessels in the iris of your right eye or something.

**Leo:** Or Opie's face when he walked into the pharmacy.

**Steve:** For facial recognition.

**Leo:** Right. He didn't need a password, and he didn't need a token. He had his face, so that was the single factor there.

**Steve:** Exactly, because of who he was, and that was uniquely recognizable. In fact, now, that's the key, is what we're really talking about here. And actually I haven't seen this anywhere in the literature, maybe because it's too obvious. But again, by making these obvious things clear, we end up with, I think, a deeper understanding. What we're really talking about is something only you know, something only you have, or something only you are. Because of course the problem with a password is maybe somebody else could know it.

**Leo:** Or if Opie were a twin it wouldn't only be him that looked like him.

**Steve:** Very good point.

**Leo:** Okay, I get it.

**Steve:** So things you know, something you know, as we've talked about, could be a password or a pin, like you use when you're using an ATM card. And in fact, an ATM card and the experience of authenticating yourself to an ATM machine is probably the most common and a perfect example of two-factor authentication because the something you know is the PIN number, and the something you have is the ATM card.

**Leo:** Got it.

**Steve:** And that's arguably much stronger than just a password. On the other hand, it, too, could be defeated. That is, you know, Mom could give her ATM card and PIN to one of her kids and say, you know, go down and withdraw $100. So the fact that the card is loanable represents some insecurity. That is, although the responsibility certainly falls on the card owner to manage that card correctly. So it's stronger than a password. But you could argue that the ATM machine is assuming that the actual owner of the card is the person who is present, when in fact that's not the case because there's no use of that third factor, which is something only you are.

**Leo:** Right, right, got it. So the more factors the better.

**Steve:** Well, yes. Except that – well, yes. In general, it's certainly the case that the more factors the better. The one downside of that is the more factors you have, the more tendency there is to trust the authentication. And what can often happen is that organizations or IT structures will put more trust in than they should because they get all worked up over how secure their solution is, and then if something comes along to break that authentication you can end up with much greater exposure and vulnerability as a consequence.

Now, of course, so we have the something you know is some information. Now, remember that there's also other classes of that. For example, there's the normal password and pin. Then there's like, you know, instances where you'll be asked for your mother's maiden name or the name of your elementary school or some, like, second-tier information if you're needing to further authenticate yourself for whatever reason. And of course, again, the problem with that is anything like that is easily sharable.

Now, we talked about the ATM card as something you have. Of course, what's becoming now popular are various sorts of dongles or tokens. RSA has something they call "SecurID" which is a little battery-powered LCD display with a six-digit number on it which changes once every minute. And so the idea is – and it's a clever solution, of course they've locked it up in patents every which way, the idea being this is a hardware token which you can stick on your keychain, and every minute the number changes. So if you need to log into any facility on the Internet which uses RSA's SecurID system, you will be asked for the current six-digit number appearing on this little token, which you just type in. And what's cool about is that it's obviously got a clock inside. Every minute it's changing this.

Well, we understand enough of crypto now that all that's required is for a number of seconds since whenever, since the year 2000, for example. The linear number of seconds is run through symmetric encryption with an individual key which turns that counter from just a monotonically increasing value into what looks like a completely random number. And in fact there's no way, without knowing what the symmetric key is, to guess, given any one number, what the next one will be. And an extension of that is there's no way to know at any point in time what number will be showing on that LCD. And the only people who know is the RSA server because there'll be some serial number on the key which is in no way related to the symmetric key being used to encrypt this monotonically increasing value. But what happens is, when you type it in, the server you're trying to authenticate to, like – I can't use Bank of America because they're not using this system. They have something else which actually...

**Leo:** SiteKey, which I hate.

**Steve:** And we'll talk – and it's bad, Leo.

**Leo:** I know.

**Steve:** It turns out it's already been hacked and broken. And they're all jumping around, talking about it as a powerful two-factor authentication, and it's not. But imagine some working system that has contracted with RSA to provide authentication services. So you enter this token into their server. Their server asks RSA – so you identify yourself by username and enter the current value on the token. RSA looks you up by the serial number on your token, which has been established ahead of time. They look up the matching secure symmetric key that is inside your token and unreadable using any practical technology, but they know what it is because they established that once. They look that up. They unencrypt, that is, they decrypt the six-digit number, which is just a number of seconds since some period in time. They decrypt that back to the counter value and verify that it's within a minute or two of the current time.

**Leo:** So that's why it's no good after that minute.

**Steve:** Well, exactly. So, however, they want to have some fudge window because the minute counter could click over, for example, your little battery-powered clock in the token could be running a little fast. And so what's very cool, though, is every time you use it, it resynchronizes with RSA a fudge factor. So, for example, RSA can see from the token you give it, it'll look at the minute before, maybe as many minutes before or after as necessary to find the value that you've entered. And then RSA can decide if that's close enough to the current time. And when they make that decision, they store a fudge factor so that next time they know, oh, this guy's dongle is running three minutes ahead. But as long as you use it every, like, three or four times a year, your clock in your little dongle isn't going to drift that far off from the last time you used it. So the RSA knowledge of where your little counter is stays synchronized. So it's a neat solution.

**Leo:** This is supposed to be the, I mean, for a single-factor security solution, this is really supposed to be it.

**Steve:** Well, it's a good solution. And in fact they've got it patented. They've also implemented the same thing, and you can imagine how, in a software-only version. They've just recently released it for the Java ME, the Micro Edition platform, that'll run on any small phone. It runs on Windows Mobile, Pocket PC, Symbian, the Palm Treo, the Blackberry. So you are able to use a regular PC-based or Smartphone and have the SecurID technology in there also because of course all of those things have a clock, and all you need is a clock and a counter and crypto.

Now, you could argue, and I would, that that's not as secure as using a specific custom token because anything that's basically in software is certainly hackable. On the other hand, it costs much less to download a piece of software than it does to make and carry this dongle. And people typically have their phones or their Treos or their Blackberries or whatever with them. So it's a good solution. By the end of this podcast we will explain why it's no good. I mean, why it's good, but it's not perfect.

**Leo:** It's not perfect, okay.

**Steve:** It's not perfect because it is susceptible to man-in-the-middle attacks, and many of these things are.

**Leo:** Right. But that's why multifactor becomes so critical.

**Steve:** Exactly, because you want as many as you can. Now, one of the other really interesting solutions which people have come up with is the actual use of a mobile phone, that is, not like I was just saying, using RSA security, but imagine that what it is that – you remember, one is what you know and one is what you have. Well, many people, if not most people these days, have a cell phone. So the fact that they're in possession of the cell phone is a strong aspect of authentication.

**Leo:** Ah, of course.

**Steve:** We're not going to overstate it.

**Leo:** It's not perfect.

**Steve:** Exactly. But it's better than nothing, and it's probably free. So imagine a service where you log in, and they send your phone, that is, the phone associated with your account, a short SMS message with a password, a series of digits which you have to type into the web browser as you're logging in. The only way you could know it is if you are in possession of the phone, and you look up the message you just received and key the number in.

And the real beauty of this approach is it's an out-of-band communication. That is to say, and we've talked before about in-band and out-of-band, the idea being that anything in-band is the main dialogue that we're having, for example, over a secure connection with a server that we're logging into, but doing something out of band is really nice. It'd be like someone phoning you at the phone near you and saying, Hi there, John, you need to enter this number into the web browser in order to authenticate yourself. That's an out-of-band communication. But using an SMS message on a cell phone is very easily automatable and provides the information out-of-band only to the person's phone that's associated with the account. So there's another sort of clean, clever approach for this kind of something you have authentication.

One other approach that I've seen is the use of a code sheet. That is, there's one company, and I can't think of their name, and again they've got patents protecting this, is when you sign up for this form of two-factor authentication, you print out a web page containing a little four-by-four, four-inch by four-inch piece of paper which is a chart of symbols and letters. And so the idea is, again, that's something you have that you can easily fold in half and stick in your wallet. And when it's time to authenticate yourself, their website will show you a series of icons, and you look them up on your chart and type in the corresponding letter.

**Leo:** That's cool.

**Steve:** It's very clever.

**Leo:** And simple.

**Steve:** And simple. And it's a one-time – basically it's like a one-time little crypto sheet. It costs nothing because everyone who can bring up a web page and print it is able to do that. So you don't need a hardware token. But it's a form of hardware token because every single one that's issued is different so no two people have the same two sheets. The issuing server remembers which sheet each person received. And then the web server is able to issue a one-time pattern of icons, that is, it will never issue those again. And so these things are trying to solve the problem of anyone collecting keystrokes. Because if you had a keystroke logger on your system that saw you entering this password, well, that password that you are giving back to the server in a classic challenge-response fashion, you've been challenged by being presented this series of icons. You look up on your unique sheet the corresponding letters and type that in. You will never receive the same series of icons again, never type in the same password again. So no one logging your whole log-on handshake protocol is ever able to reuse it.

**Leo:** And that's kind of the key to a lot of these systems is these one-time passwords, or one-time authentication.

**Steve:** Yes. And of course that is the whole RSA SecurID deal.

**Leo:** Right, it's not good again.

**Steve:** Since it's changing every minute, it's not good again. It's funny, because as I was thinking about this and doing some research, I was thinking, okay, well, what if you did a typo, and you had to enter it again? Well, that's okay. First of all, in a minute, or actually in an average of 30 seconds, you're going to get another one, and that will be okay. But if it's a typo, it's wrong anyway. So if you correct it, then the server goes, oh, you got it right. And in the process of getting it right, it will no longer ever accept that again. So it's very cool.

So some of these things involve hardware. There's even another company that has something called the iButton. Dallas Semiconductor produced something called an iButton which looks exactly like one of the little tiny hearing aid batteries that we're all familiar with because our calculators use them, or obviously people with hearing aids use them, the little round cell which is very small and has a sort of an inner circle contact on the other side. Well, Dallas Semiconductor realized, hey, we're able to provide power and data over two wires. So they build crypto chips into something that looks exactly like a hearing aid battery and then mount them on a little paddle that you can hook onto your key ring. It doesn't have a battery in it, but when you press it against their little socket...

**Leo:** It charges it.

**Steve:** Well, it powers it up, and then it’s able to do a little security protocol to say, give me your ID number, give me our serial number. So it is, it’s very...

**Leo:** That’s a very clever technique, yeah.

**Steve:** Yeah. And super small, lightweight, easy to have on your key ring, and it provides – again, it’s something you have that nobody else has.

**Leo:** But there’s problems with all of these. Something you physically have can easily be lost.

**Steve:** Yes, exactly. Or loaned. It can be loaned.

**Leo:** Yes, exactly.

**Steve:** So you tell your child, only withdraw $100. Well, or 20 or whatever to go to the movies. And certainly obviously there is accountability when the bill comes at the end of the month. But still, you’ve given up control. So the next factor in this hierarchy of authentication is something you are. Which is to say, some measurable physical characteristic. Fingerprints, of course, are very popular and have long been used. Speech patterns are also being used. You’ve probably also seen, Leo, over the years there was an attempt to use the typing rhythm.

**Leo:** Right. That keeps coming back. People keep talking about that.

**Steve:** Yeah. It’s interesting because no two people type with the same kind of rhythm. Many people don’t type with any rhythm at all. And of course now that requires something active on the computer that is down at a level where the rhythm of your typing can be received, for example, that would not work through a web form because you type offline, essentially, and press the button to submit. And then of course the more high-tech retinal scan or some other biometric.

In fact, just yesterday I wanted to grab something from my server cabinet at Level 3. And so I went there. They use, interestingly enough, they use the second two approaches and not the first. That is, something I have and something I am. I have a badge from them. And so I wave the badge near a reader which tells them whose badge this is. But it doesn’t tell them that it’s me, of course.

**Leo:** Who you are, right, yeah.

**Steve:** Exactly, because I could have given it to someone else. But then the next thing I have to do is stick my hand in this, it’s sort of a plate with some poles sticking up, little one-inch-high posts. So I put my hand all the way up into this so that the posts rest in between each of my fingers. And this thing measures the physical characteristics of my hand.

**Leo:** So it’s not exactly a scanner; or is it?

**Steve:** Well, no. And this is what’s interesting and is significant is that certainly there are probably people around, I’m sure there are, walking around who have a hand measurement that is indistinguishable from mine.

**Leo:** Ah, but they don’t have the card.

**Steve:** Exactly. They don’t have the card.

**Leo:** Or the likelihood of them having a card is highly unlikely.

**Steve:** Exactly. And the likelihood of someone stealing my card who has the exact same hand size that I have is low enough to make this much more secure than just using a card. And certainly just using my hand measurements is really not security at all. So the point I want to make about this, because it’s clever the way it interacts, and it relates to fingerprints in an important way that we’ll talk about in a second, is that just using a hand size really isn’t enough to identify me uniquely in the world. So I wouldn’t want to use that. But using my card first tells the system, okay, now check the one hand size we have on record. Or maybe one or two, whatever, depending upon how many people are going to use this card. In fact, with this system that Level 3 uses, it’s a one-for-one association. If I had Sue or Greg needing access to the Level 3 facility, they would have their own card and their own hand measured for that card. So there’s a one-to-one mapping.

But the point is, this thing never says, oh, look, this looks like Steve’s hand, let’s unlock the door. No. Instead, it’s, oh, we’ve just sniffed Steve’s card, and we know what Steve’s hand size is, so let’s see if the hand that’s about to be put on the scanner qualifies as being Steve’s. So it’s a really nice system that – and nothing to remember. Notice that I’m not putting in a PIN or a password in this system. However, I noticed that on the scanner there is a keypad. So probably, if Level 3 were even...

**Leo:** Third-level authentication, third...

**Steve:** Yes. All of the technology is there for me to have to enter a PIN and a card and my hand. And I guess they’re thinking, okay, let’s, you know...

**Leo:** Enough is enough.

**Steve:** The hard two out of the two out of three is good enough.

**Leo:** Right. Fascinating. And I think a lot of network operations centers use something like that. That seems to be very common.

**Steve:** Yeah. And in fact, as I was thinking about talking about this today, I realized that even if someone got my card who had my hand – and I don’t mean that literally because of course we’ve all seen the spy movies where guys...

**Leo:** Cut your hand off, right.

**Steve:** Exactly, cut my hand off and take it to Level 3 or...

**Leo:** Would that work?

**Steve:** Yeah, I think it would. I don't think it has to be a living hand.

**Leo:** They're not measuring the heat of the hand, okay.

**Steve:** I don't think so. And of course all of the spy movies we've also seen where you get your finger chopped off, and then they carry the finger and stick it up on the fingerprint scanner. Or on "Mission Impossible," you know, they're always peeling...

**Leo:** Plastic, rubber...

**Steve:** ...fake fingerprint, you know, impressions off of you. So fingerprints are interesting because they're sort of like hand size. That is to say, yes, you could argue that certainly a whole set of fingerprints, a whole hand, provide enough unique identification to be strong. On the other hand, given the opportunity to get DNA, which is absolutely unique, that's what forensic researchers and scientists and, you know, forensic criminologists use rather than fingerprints. And in fact, a fingerprint is vague enough and also susceptible enough to damage from, you know, an Xacto knife, and fingerprint databases tend to be huge, that they don't really work the way they show them on CSI, where it's like, oh, scanning and all these fingerprints and people's faces are flashing by. Because, I mean, those databases are way too huge.

The way fingerprint scanners work is that a number of characteristics of all fingerprints are found. There are swirls, there are breaks, there are places where, like, four ridges fade out or, like, merge into two, sort of like a Y intersection in the road. Well, those things are easily spotted by software, and they create position-dependent marks. That's then used to make a robust index. And that index is used to index this criminal fingerprint database in order to then sort of go to a successive level of refinement and decide if that's who you are. In the case of fingerprint scanners which are now appearing on PCs, they operate much more like the Level 3 analogy, where you have very few people who are logged in to this fingerprint scanner, you know, yourself, maybe a spouse or your kids or something. I was going to say your dog, but not in this case because it's not paw print scanner.

And so as you drag your fingerprint, or your finger, across this, one or maybe multiple times, it builds up an image. And then all it has to do is it says, okay, all I need to verify is that within sufficient level of certainly I recognize this fingerprint from among one or two or three. And again, the likelihood of some random thief having a fingerprint sufficiently close to yours is just diminishingly small. And since you are able to train this multiple times, and you are using it continually, it's able to adapt over time to slow changes that may evolve in your fingerprint, like, you know, a little cut appears somewhere. It's like, okay, we'll forgive that because all the rest of the finger looks good. So again, you're only really doing a comparison against a very small database of potential matches. So you can afford to do a very good job of that.

And the other cool thing about the newest scanners, you know, the original scanner was sort of a plate you put your thumb or your first, your forefinger on or something, and it would take a picture of the whole thing. There has been lots of work on how easily that is forged, just by using some rubber cement. It turns out you can very easily fool these because they're not checking for the likeness of the fingerprint. And so those things like the "Mission Impossible," you know, rubber finger really do work on those. But the newer scanners, which are even less expensive and so are becoming predominant, are more like a line scanner, where you have to actively draw your finger across that. Well, that's superior because it's a dynamic process. It's much less easy to make a rubber finger that is going to fool that than it is if you just had a Xerox copy of your fingerprint and just stuck it down on the window.

But it turns out that all of these things, while they are better, there are attacks that are workable attacks against these kinds of things. Now, it's interesting, you mentioned BofA and SiteKey. And it's certainly gotten a lot of press because it's a very sort of, oh, touchy-feely sort of solution. The idea with BofA's SiteKey is you give it your username first. It looks up your username and says, ah, Leo has registered a picture of his trash can with us. And so it sends back...

**Leo:** And but you give it a phrase, too, that goes with it.

**Steve:** Yes, exactly. A phrase and a picture. It sends those back to you. Now, what that's supposed to do is give them phishing protection, that is, it's supposed to verify that you're really talking to BofA because they know they have a picture of your trash can that you provided them when you set this up, and no phishing site would be able to do that. And so the idea is that it's like, oh, good, now I know I'm talking to BofA. And then they prompt you for your password.

Now, the problem is that they need this to be robust in the event that you want to sign on at the library – god help you, you don't want to sign on at the library. We've certainly covered that enough. But the point is, what happens if your IP changes? What happens if you're at a neighbor's who you trust, or you use a laptop, and your IP...

**Leo:** Well, that's what happens because I sign on on three or four different IP addresses, or I'm in a hotel, and I'm signing onto my bank.

**Steve:** Okay. You don't want to admit that here on Security Now!, Leo.

**Leo:** Well, now, wait a minute. It's an SSL connection, so I'm okay even doing it at the hotel; right?

**Steve:** Well, yes. Although...

**Leo:** It's my laptop. I'm not using their computer.

**Steve:** It's your laptop. And we know that you would be careful. But the way phishing sites still crack this, well, okay, wait. First, there's a non-phishing approach, or, that is, a simple way of breaking this. MIT did a study where they deliberately presented people who were used to using the SiteKey system with a page that did not challenge them with the picture.

**Leo:** Oh, I'd be happy. I don't want to see that stupid picture.

**Steve:** People complain about it all the time.

**Leo:** I don't want the picture. I just want to sign in.

**Steve:** Get this. 97 percent of people didn't notice that they weren't being given a picture of their teddy bear or their dog Spot or...

**Leo:** It was a relief.

**Steve:** Exactly. And the other problem is, and I'm sure you've had this happen, is there are instances where you will be asked for some additional information...

**Leo:** When I use a new computer, and I'm at a different IP address, then it wants a bunch of – and it's a real pain because it asks all these silly questions.

**Steve:** Well, and that, it turns out, is the weakness. Because if a phishing site creates a pseudo-site for BofA and gets you to click on a link, and we've all seen, and we've talked about how it's possible to obscure the URL so it looks like you're still on BofA...

**Leo:** I think I'm at BofA. Let's say I'm not paying attention. I think I'm still at BofA.

**Steve:** Exactly. So what BofA will do is they will see you don't have a cookie. Oh, and by the way, they use Flash cookies, which we will be talking about here before long.

**Leo:** Really. They're not using standard browser cookies, they're using Flash cookies.

**Steve:** Also, but I do know that they are a user of Flash cookies.

**Leo:** Interesting, huh.

**Steve:** And it says so in their privacy statement about the whole SiteKey system. So if they see that you're at a different computer that doesn't have a cookie, that's when they ask you some additional information. Well, it turns out that, if you were a victim of phishing, which is what this whole thing is designed to prevent, you log into what looks like BofA, and it's not. They ask you your username and your state. So you fill that in. The phishing site turns around and submits that to the real BofA site.

**Leo:** And gets the SiteKey.

**Steve:** Yes. Yes.

**Leo:** Give me your SiteKey. Now, actually it won't get the SiteKey, will it, though, because it's a different...

**Steve:** No. What happens is, the BofA site asks the question, which the phishing site turns around and asks you.

**Leo:** Passes on to you. So you fill in the answers.

**Steve:** So you fill in the answers.

**Leo:** And it fills in the answers.

**Steve:** It sends it back to BofA. BofA then finally says, oh, this must really be Leo in a hotel.

**Leo:** Here's the SiteKey.

**Steve:** Sends the SiteKey, which the phishing site bounces through, providing you with the picture you're expecting, and you give it your log-in.

**Leo:** So it's completely stupid.

**Steve:** It's completely stupid.

**Leo:** Which I always knew.

**Steve:** Yes.

**Leo:** Not so clearly proved. But it struck me as really silly. And the real problem is people who get fooled by phishing scams are not savvy enough to be figuring out how the SiteKey works, what to watch out for and all that. That's the whole point. They're trying to protect people who are not savvy.

**Steve:** Exactly. And we've just seen that a non-savvy person can still get fooled...

**Leo:** Easily.

**Steve:** Even with a picture of their kitty cat on the screen. It's like, oh, I know I'm using BofA. I'm going to transfer all my money.

**Leo:** Well, so there's no reason for me to go through this stupid – you know, the pass phrase I use with it is very insulting to SiteKey because it annoys me so much, even though I know nobody at BofA is reading it. It just makes me happy. Every time I see the SiteKey and the word stupideffingsitekey come up along with it, it just makes me feel better. I recommend this to everyone.

**Steve:** Well, so essentially the takeaway from this is that multifactor authentication is generally a good thing.

**Leo:** And that's what they're trying to do here. This is multifactor.

**Steve:** Yes, well, that's what it is. Well, technically, right, it's more than just you using a password. Actually what they're trying to do there is they're trying to provide the authentication which already really exists in the SSL secure connection.

**Leo:** Ah, yeah, that's true.

**Steve:** And in fact that's where it's really good. But again, that's what people aren't checking. That is, if you verify that you have a secure connection, and you take the trouble to check the certificate, and you see that it's issued from Bank of America directly from Equifax or VeriSign or somebody trusted, that's the way to know you've got a really good, nonspoofed connection. But exactly following your point, Leo, typical users don't do that.

And the problem is, this thing is trying to say, we're really BofA. Look, here's a picture of your kitty cat. And so it's actually providing a false sense of verification because it has been cracked. It has been hacked. There are phishing sites that are using man-in-the-middle attacks on the SiteKey technology. And unfortunately I guess Yahoo! has adopted it and a bunch of other companies have adopted it after BofA made such a big splash about it. And it's like, well, okay, this is better than nothing, you could argue. But once again, the fundamental technology of verifying the validity of your SSL secure sockets connection is a much better approach because it's real, and that cannot be, as we've seen, man-in-the-middle attacks are thwarted by SSL as long as you verify the authenticity of the certificate that you've got.

**Leo:** Well, and I think this is a good example of a company going for the appearance of security and not really the fact of security. This is such an obvious, intrusive technique that it gives you the feeling of, oh, they must be taking care of me. When in fact it has absolutely none of that.

**Steve:** I think that, I mean, I like the idea of simple biometrics, for example, using a little fingerprint scanner on a laptop I think makes a lot of sense, as long as the underlying – as long as what's done with the fingerprint, that is, the underlying technology to deal with the fingerprint makes everything else secure. For example, you really need this stuff stored in the BIOS and for that to unlock your hard drive, for example, rather than waiting for Windows to boot and then using that to log on. Because by the time that's happened, you know, all bets are off.

Now, all of this, this whole issue of multifactor authentication, this actually came out of the research I've been doing into something we'll be talking about soon, which is the Trusted Platform Module, TPM. Because that ends up being an interestingly robust, or at least robustable – there's a new word for us.

**Leo:** Robustable.

**Steve:** Another way of providing a very secure authentication factor which of course has a mixed blessing because, again – and you could argue that even these approaches are mixed blessing because, to the degree that they limit user freedom, there are people that are saying, well, you know, I'd rather just use my password, my username and password, because I want the flexibility of sharing it with some other people. So again, all of these things involve tradeoffs. And if someone's going to have their retina scanned, they might object to a laser being beamed into their eyeball.

**Leo:** You know, that's actually what Customs in the U.S. is using now is a retinal scan and fingerprinting. And a passport.

**Steve:** Yes, exactly. I have been fingerprinted as I've been going between here and Toronto.

**Leo:** Have you?

**Steve:** Have you had a retinal scan – yeah, actually, last year sometime, you know, stick your thumb here. And I was like, okay. Oh, wait a minute, no. I'm thinking of the DMV. It was the last time I renewed my driver's license.

**Leo:** They do, if you're coming from the U.S. generally into Canada they don't, or vice versa, I should say. But if you're coming into the U.S. from many other nations, they absolutely do. And now they're using, there's a kind of a fast pass thing they've got for – they call it CANPASS in Canada, and NEXUS, I think, is another word that they use in the states – that allows you to kind of breeze through with your passport and a retinal scan. And again, it's dual-factor authentication being the idea behind it. And of course the reason they let you do this is they've pre-interviewed you, and they've done a background check and so forth.

**Steve:** That sounds like a cool thing, Leo. I'll definitely sign up for that.

**Leo:** Yeah, I'm going to sign up.

**Steve:** I don't mind if some laser scans my retina.

**Leo:** And I've watched people do it. You know, in Vancouver there sometimes is a very long line to get into Canada. And I've watched these guys just whip through the NEXUS line. And they bend over, they get their eyeball scanned, and they...

**Steve:** Wait a minute. They bend over to get their eyeball scanned?

**Leo:** Well, they're bending forward, let's put it that way.

**Steve:** Oh, I see, bend down, right.

**Leo:** Very important, yes. The machine is a little lower than eye height. And it looks like then you do talk to a person, but very briefly, and you whisk right through while the rest of us are standing in line. So it’s kind of a neat thing. And the sad thing is that a lot of people coming to the states now from other countries are really put through the third degree, including fingerprints and pictures and passports.

**Steve:** Well, you know, the nice thing about that is it appears that retinas are very hard to spoof. I’m not sure how you would spoof someone’s retina. I mean, especially if you were under supervision by a guard. You could imagine you could make a fake eyeball, and then you wouldn’t have to bend over.

**Leo:** Right. Here, just do this.

**Steve:** But at the same time I like the idea of, for example, being known as somebody who’s a member of NEXUS. And if you could, like, check off "require me to have my retina scanned,” then that would provide – because I’m wanting stronger authentication, it would prevent somebody from spoofing me and coming into the country pretending to be me.

**Leo:** Right. It makes perfect sense, you know, it’s a good way to do it. And in this day and age of heightened security, it’s, you know, whatever it takes, I guess.

**Steve:** Well, and here we are talking about Security Now!, and the issue of authentication, which is certainly going to be an ongoing concern in the future because with everything moving electronic, with more and more services going online, the idea of establishing your identity to whoever it is you’re having a transaction with is increasingly important. And unfortunately the bad guys are seeing more and more gain from coming up with ways to circumvent whatever authentication schemes people come up with.

**Leo:** That’s true. If you want to know more about authentication, I bet you Steve’s got lots of information on his website, GRC.com. Am I right, sir?

**Steve:** Yes, sir.

**Leo:** That’s where the show notes live; the 16KB version of the show for the bandwidth-impaired; the transcriptions, the fine handmade transcriptions by Elaine; and of course Steve’s fine array of free security programs like ShieldsUP, SecurAble, DCOMbobulator, Unplug n’ Pray and on and on and on. And of course who could forget SpinRite, everybody’s favorite disk recovery and maintenance utility. So that’s a – I think that that’s just a fascinating subject, this multifactor authentication. And it makes me want to do more to authenticate myself.

**Steve:** Well, Leo, you are authentic for all of us.

**Leo:** PGP doesn’t seem enough.

**Steve:** We absolutely know you are you. There are various interesting offshoots of this that we will be discussing in coming weeks.

**Leo:** Oh, good.

**Steve:** Because authentication is a big issue, or should I say a big factor, in all things happening on the Internet now, and only more so in the future. I think ultimately the notion of username and password over time will end up being considered no longer sufficient for many applications; and people will, you know, maybe we'll all just have something like an RSA SecurID dongle or various schemes will become popular, and people will get used to them and appreciate the additional security that they do provide, even though you could argue nothing is perfect. Then again, you know, people aren't. So that's the...

**Leo:** It's actually amazing that we're able to do as good a job as we do, yeah. Hey, thank you so much, Steve.