Listener Feedback Q&A #19

Description: Steve and Leo discuss questions asked by listeners of their previous episodes. They tie up loose ends, explore a wide range of topics that are too small to fill their own episode, clarify any confusion from previous installments, and present real world "application notes" for any of the security technologies and issues they have previously discussed.

High quality (64 kbps) mp3 audio file URL: http://media.GRC.com/sn/SN-092.mp3
Quarter size (16 kbps) mp3 audio file URL: http://media.GRC.com/sn/sn-092-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 92 for May 17, 2007: Your questions, Steve's answers, #19.

Time for Security Now!, everybody's favorite weekly look at the world of security. Not just Windows security, but all kinds of security for technology for computers and the Internet. Steve Gibson is our genial host, the security expert at GRC.com, the creator of ShieldsUP and so many great free security utilities and, of course, the great SpinRite hard drive maintenance and recovery utility. Hello, Steve.

Steve Gibson: Hey, Leo, great to be back with you again for, boy, we're approaching 100 here.

Leo: Oh, we've got to do something for the 100th.

Steve: Yeah. Well, actually I do have a surprise.

Leo: Oh, good. That'll be fun. Well, that's not so far off. But meanwhile, as every fourth episode, every mod 4 episode, we do our listener questions and answers. We've got quite a few brilliant, inspired questions.

Steve: Actually I think this is going to be a really good collection of questions. I assembled them, and we ought to mention that I had said last week that we're going to begin doing sort of

a mailbag-and-errata section at the beginning because I'm getting so many just nice, you know, they're not questions for the Q&A, but they're just informed comments and opinions on things. So I don't have anything other than a fun little SpinRite anecdote to share this week. But in future weeks I'm going to take the time to read more of my mail and always have a couple fun things to share at the beginning.

Leo: Good, I look forward to that. That's great.

Steve: Someone named David Ward, who actually is with a computer company, WellMax computer, said, "I had a friend bring a laptop to me and tell me that the fan was failing. The machine would freeze after about five minutes into the Windows XP boot process."

And my first thought was, okay, well, if Windows is taking five minutes to boot, the fan may not be your only problem. But, you know, it does seem like...

Leo: Five minutes into the boot process sounds like it's laboring, yeah.

Steve: Five minutes into the boot, exactly. So he says, "He had taken it to three other techs who all said the same thing, that the fan was failing. I experienced the same problem. However, after booting into safe mode and having the machine stay on for an hour, I knew it was not a fan issue, but a hard drive issue."

So apparently he was able to boot into safe mode and not encounter whatever driver was being loaded that was having a problem with the hard drive and causing this trouble. So he says, "Sure enough, after running SpinRite" - get this - "for over 180 hours..."

Leo: Oh, that's dedication.

Steve: Yeah.

Leo: It slows down when there are disk defects; right? I mean, it has to work harder.

Steve: Yes.

Leo: Because it doesn't normally take that - I want to emphasize this. It doesn't normally take that long.

Steve: Oh, it's often a few hours, maybe seven or eight. But it is absolutely a function of the condition of the drive. What's going on is that while we're doing data recovery, I suppress the drive's normal intention to relocate a damaged sector somewhere else. Because once it does that, that sector is then unavailable. You can't ever get the data back. It's gone. So I do a bunch of low-level commands to say to the drive, okay, shut down relocation, don't move this away till I'm through with it. And the point is, I'm never going to have another chance to get the data back because ultimately I want to recover everything possible and then let the drive remove this damaged sector from use so that the data that I put back is being put back onto a swapped-in spare sector. So there's much more going on under the hood than is apparent from the fact that SpinRite just sort of scans over the drive. But it will sometimes stop and just crank

on a single sector for quite a while and then pick up speed again and typically finish much more quickly.

Anyway, of course, this story has a happy ending. He says, "It finally finished and found many bad sectors." Which is the point. SpinRite was, you know, working hard. He says, "However, it booted successfully then and ran very well. All the customer could do was shake his head and utter the phrase, 'Amazing,' several times." Of course I guess the customer had to come back a few days later. But at least he was amazed.

Leo: 180 hours later. That's like a week later.

Steve: So I just - I love the stories, so keep them coming.

Leo: Shall we start with the questions, sir?

Steve: You betcha.

Leo: Starting with John Jones from Willington, Connecticut. He's wondering about the future of IP, Internet Protocol. He writes: Now that we all know about IP addresses and networking, could you tell us about IPv6, what it means to the little guy? Do we have to go out and buy new routers? What is it? How's it going to affect us?

Steve: It's a great question. The good news is, I can't imagine anything that will be more transparent. Essentially, IPv6 does a number of things. First of all, it dramatically expands the so-called IP space, that is, the number of bits in an IP address. It takes it from 32 to 128. So, now, because every bit you add doubles the number of total combinations, because you have all the combinations you have with the bit off, and now all those combinations again with the bit on, so each binary bit you add doubles. So 32 compared to 128, I mean, even though it's "only" four times longer, I mean, it is - I don't even have the number on-hand, how many IP addresses that is. It's just a gazillion gazillion gazillion. So whereas 32 bits, of course, gives us 4 billion, 4 gig, IP addresses. And that's kind of a familiar number. That's, for example, the most RAM you can put in a lot of systems is 4 gigs, so forth, because they're 32-bit addressing. So, first of all, it solves the concern that many people had a decade ago that the world is going to run out of IP space.

Leo: I remember talking to Vint Cerf, the father of the Internet. And that's exactly what he said. He said we have to go to IPv6 - this was probably about almost 10 years ago - because we're running out. Turned out not, didn't it.

Steve: Yes, yes. What happened was NAT, Network Address Translation, that we've talked about so often, completely changed that landscape. I've got, I don't know, 13 computers here, Leo. And they - actually I have a block of IP addresses because I'm not a typical user. But...

Leo: Well, use me as an example. I have that Astaro Security Gateway and probably a dozen computers coming off of it with one IP address.

Steve: Yes. And so the idea that you could use port mapping in order to disambiguate data crossing a NAT router, that really dramatically reduced the pressure on the IP space. But again,

it really looks like the Internet is here to stay. It's not a passing fad. And so everyone figured, hey, you know, let's just solve this problem. And they solved it by making the addressing space so big that no one ever needs to worry about it running out.

Leo: I think Vint said that every molecule in the universe - or, no, the galaxy - could have its own IP address.

Steve: It's on that level.

Leo: It's a large number.

Steve: Yeah. So people will have no problem with IP-enabled refrigerators and washing machines and toothbrushes here, which, you know, Bluetooth stuff.

Leo: I imagine that'll happen, so it's a good thing.

Steve: So the first thing it does, the first thing IPv6 does - I don't know what happened to 5; 5 just never happened. Because we're on IPv4 now. And also 1, 2, and 3 sort of happened without anyone knowing, before the Internet all actually happened.

Leo: It's off with 802.11c getting drunk somewhere.

Steve: Okay, good, I believe that.

Leo: They never used us.

Steve: So in addition to a much huger addressing space, the only other real changes are that a number of things which were add-on sort of after the fact, things like IPSec security, which we've talked about extensively, which are not part of the IP spec, they were, you know, grafted-on protocol layers that would run on top of IP. That technology and additional authentication technology has been sort of subsumed by the formal spec. So good, useful things that no one back at the IPv4 era could have anticipated, with IPv6 they're brought into and made standard in a robust fashion. So that's really good.

Now, the reason I don't think this will ever be anything other than transparent is that we've still got a long way to go before IPv6 is what actually runs everything. And it can be transparent because the existing IPv4 addressing space just tucks into one little itty-bitty corner of IPv6, which is to say that IPv6 was designed as an upward-compatible extension on IPv4. So that even if you had an IPv6 for a so-called backbone, for example, running across the country, that IPv6 backbone can carry IPv4 packets with no trouble at all. Basically you just put a whole bunch of zeros on the front, and IPv4 lives in a little corner of IPv6. So packets can come in and go out of IPv6 with no trouble at all.

Similarly, when consumer routers need to be upgraded, the consumer router can simply have a firmware update that does the same thing, that is, it could be IPv6 aware on the inside and the outside, and perform that cross-version transform easily in either direction. So maybe at some future point, if you had an old IPv4 router and you were using some next-generation ISP, even

if they needed you to be using IPv6, you could upgrade your firmware or go out and spend another 40 bucks to get an inexpensive router. It's not like there's anything about IPv6 that is more expensive. And Vista already fully supports IPv6. That was an add-on in XP. You could add the IPv6 so-called stack into XP. In the case of Vista, it's right there, built in, and running all the time. So if you ever plugged a Vista system into an IPv6 system, it would just work natively IPv6 without even needing to perform the 4-to-6 and 6-to-4 translation.

Leo: Whatever you say, I agree.

Steve: So there you go. Basically it will have no effect on anyone.

Leo: It's just going to happen, and don't worry about it.

Steve: I was just looking at the last line of John's question. He says, "What is it, and how will it affect us?" So that's what it is, and it'll have no effect.

Leo: No effect. Rick in Atlanta says he has something to add about multifactor authentication, what we talked about last - actually two weeks ago now on the podcast. Listening to the multifactor authentication episode - that's Episode 90 - I wanted to bring attention to a method that was not mentioned, possibly because it may be unknown to you both. It's the method of using IP intelligence - oh, boy - geolocation, domain information, velocity checks and so forth to determine a risk factor for an online transaction. And it all works transparent to the user, which should make Leo happy as it is his biggest complaint of SiteKey. Do you know about this IP intelligence?

Steve: Yeah, it's an interesting additional factor. Now, of course two weeks ago we talked about multifactor authentication. And in a couple weeks we're going to have an episode titled "The Fourth Factor," which actually is different than this one. But this is an additional factor. Consider, for example, Leo, that you're using a system, as you were with SiteKey, to hook onto BofA. Well, we do know that they will prompt you if your IP address changes and if the Flash cookie that they're using doesn't match up.

Leo: Right, yeah.

Steve: But consider that they saw you were connecting from China. Now, I don't mean to be a...

Leo: That would be a bad thing.

Steve: I don't mean to pick on China. I mean...

Leo: I might be going to China. But in most cases it's somebody else.

Steve: Yes. And so what this IP intelligence is, is it's a really interesting service that a number of vendors are offering as sort of a third-party provider where they're able to say, do we believe this person is in Northern California? No, this person appears to be on some island in

the South Pacific somewhere. So it's like, whoops, wait a minute, we'll provide that information to the party that's interested in doing authentication, just so they know maybe to raise another red flag. So it's a very cool solution because, you know, we know that in general people are going to stay relatively local. And so even though you're not guaranteed of that, you could certainly factor that information into your decision about whether to trust somebody connecting based on where they're located.

Leo: He mentions velocity checks. Does that mean something like how long or how many servers it's going through or how long it takes the traffic to go from...

Steve: You know, I'm not familiar with the term. I saw that, too, and didn't take any time to look it up. So...

Leo: But it makes sense. If something's out of the ordinary about this transaction, put up a few extra questions. And that's exactly what Bank of America does.

Steve: Well, and the point is, why not take advantage of all the information you have? Certainly the IP of the person connecting to you, that is something, as we know, that cannot be spoofed because you need a direct connection to your machine. Now, you could be running through multiple layers onion routing, or any other kind of proxy server. So that's an issue. Although, if it's a secure connection, as we assume it would be, an SSL connection, that cannot be routed through onions because you need to have a matching certificate from the far end. So that's a non-spoofable IP address. Why not factor the connector's IP into the whole thing? Thus IP intelligence.

Leo: If I'm a good hacker in China, I'm going to be darn sure that my IP address doesn't come from China. But that's another matter. Listener Ian Williamson of Ottawa, Canada, has an important observation for us to share with our listeners. He says: Since making the changes to maintain WiFi radio silence - actually a term I think you coined.

Steve: I think that was, yeah.

Leo: I have noticed, he says, that my laptop has much more difficulty connecting to and staying connected to a weak signal from an access point. In other words, its wireless performance is poorer than the other laptops in the same situation. By rechecking the box for "Connect even if this network is not broadcasting," I find that the weak signal performance is better. Maybe there was some method in Microsoft's madness of not making these updates standard issue. You agree?

Steve: Well, I don't know, but I wanted to share this with all of our listeners, since back - I think it was Episode 86 we were talking about how to get a laptop to be completely radio silent so that it would not be broadcasting the fact that it exists. That's the checkbox you need to turn off after you've updated yourself to the very latest set of software from Microsoft, their WiFi update, which is not part of the normal Windows Update process. So it's by turning that off, and that's what you have to do to keep your laptop from having its own beacon where it's broadcasting its identity.

Now, I can't state why that would have an effect on connectivity. I mean, you can kind of see, well, okay, maybe the access point, if you're not sending out a beacon, the access point could be configured to drop you, except he's saying that it's a function of signal strength. So it

sounds sort of questionable to me. But if any of our listeners have experienced the same thing, I absolutely wanted to share it. And I'll be looking for any additional feedback from people to see if other people can duplicate this, or if there are people who are not seeing it. I know from my own experience I've not noticed that kind of issue. And I'm using WiFi much more now that we have WPA and I'm willing to use it at all. And it just works fine for me with all that stuff turned off.

Leo: Anybody who has anything to do with RF knows it's completely a black art. No one understands why anything works or doesn't work.

Steve: Well, Leo, radio can't work. I mean, figure it out. You have no wires.

Leo: There's no wires. And people, RF engineers, you know, and I've worked with them for years in radio, RF engineers, you know, these guys, they hear something new every week, you know, that just - oh, really, you're hearing the station in your teeth, eh? Okay.

Dave McGee of Toronto and various hotels, apparently, around North America, asks: I have a question about unsecured wireless networks. At home I use WPA with a 63-character PSK. Obviously a long-time listener. The issue is I'm never home. I travel for work and rely on unsecured hotel WiFi for all my work and personal Internet access. What would you recommend I do to secure myself when on these networks? I'm guessing Leo could use some advice on this, as well, as he travels a lot. P.S.: SpinRite rocks. It saved a drive at work. Thank you, thank you, thank you. Well, that's nice.

Steve: Well, we certainly understand his problem, and we've talked about this before, but I thought it was worth reiterating. If you're on the road, even if you were able to use WPA, for example, to connect to a hotel's wireless network, you're participating in a massive network which is fundamentally spoofable. The problem is, and we talked about this, as I mentioned before, is ARP spoofing, which allows one person in a network to potentially insert themselves as a man in the middle by sending packets to the various endpoints, tricking their ARP tables into believing that it is the holder of the IPs coming and going so that they send all their traffic through you. It is a trivial thing to do. It really exists. There are free tools now on the 'Net for allowing people in this situation to knit themselves in.

So the only thing you can do is make sure that your connections themselves are secure. Meaning, for example, if you were using Google Mail, you want to use secure Google Mail, https://mail.google.com, or you want to use some sort of a VPN. Ultimately that's the right solution is to use one of the free or commercial third-party VPN solutions, where you have a VPN client running on your laptop in the hotel room. It's connecting out through the hotel network to a publicly located VPN server. And we've talked about HotSpotVPN as one that Leo and I have both used with great success in the past. That way your traffic is going through an encrypted tunnel until it gets way out of the hotel LAN environment, which is so vulnerable, and out onto the public Internet. Then it's decrypted, and your traffic then emerges onto the Internet well away from the hotel. That's really - a solution like that is the best approach.

Leo: And if you're a business traveler, often you have a VPN through your business that you could use.

Steve: Back into your corporate network.

Leo: Right. And then you could surf out from the corporate network. But, yeah, it's a little scary, I mean, the idea of using somebody's open Internet access. And as we talked about, hotels are crazy.

Steve: Well, and there have been security researchers who have just sniffed the traffic in their hotel room, and over the course of an evening captured 180 log-on names and passwords and servers from other hotel guests.

Leo: Yeah, she's famous for that. Just for fun she does that.

Steve: Exactly. Remember that? Just sort of plugged in to see what was going on.

Leo: Moving on, A Student who loves Security Now! in rural Alberta wonders: On SQL injection exploits, do scripting languages like, say, Perl, PHP, or VBScript have the language constructs to allow sanitization of users' input? Yes, by the way, they do, in many cases.

Steve: Yes.

Leo: Where do I find the links that you alluded to during the podcast? On the dangers of cross-site scripting, would you be able to rate and compare the various popular scripting languages with regard to the degree of their insecurity, assuming it could be quantified at all? This is one for Randal Schwartz, who is a regular listener and a Perl guru.

Steve: Well, and it's a great question. The only comment I guess I would have is that, first of all, yes, as you said, any language which is doing this has string operators and the ability to process strings and could be used to purify the user's input data, which as we saw before when we were talking about SQL injection exploits, that's where the problem occurs. So any language can be used to do that. If I had to respond to which language is better, I would say, well, there's really no difference because every language is equally able to do this. The only thing I would say is, well, it's entirely a function of the programmer. The safer language might be the one that tends to have the more security-aware programmers.

Leo: It's all about the programmer, absolutely. Every language has the tools.

Steve: Yes.

Leo: Perl has some great - go to CPAN, and you look at the CPAN modules, there are great modules for doing this, like LibWeb. But, you know, a programmer has to use them.

Steve: Exactly.

Leo: I remember talking to the guy from w00w00. I can't remember his name. Like eEye.

Steve: You sure remember where he came from.

Leo: Yeah, great security company like eEye, and he's a young guy. He said, look, it's simple. Instead of using strcpy, use strncpy. I mean, it's little things.

Steve: And Leo, I forgot to mention this. I really haven't had an occasion to. But in reading some background that I was reading in the last few weeks, I ran across the formal statement that Microsoft went through, and in Vista they removed every single instance of strcpy.

Leo: Because that's clearly an invitation to a buffer overflow.

Steve: Yes. The idea is, for those who are not C programmers, or programmers with a similar API, strcpy stands for string copy, where you give it a pointer to the buffer containing a string, the pointer to a buffer of the destination for that string to be copied to. And C strings are zero-terminated, that is, null-terminated. So the strcpy command will copy all the characters from the source buffer to the end buffer, up to and including the final terminating null, or zero byte. The problem is, you could see, if you gave it an extra long source string, and the program was expecting a shorter destination, it was expecting a shorter string so it gave it a smaller destination buffer than a source buffer, you could easily - well, the program would just keep on copying right over past the end of the destination buffer. Strncpy has a third parameter. You not only give it a pointer to the source buffer and a pointer to the destination buffer, but a maximum character count to copy. So you give it the size of the destination buffer. And no matter what, even if it hasn't yet encountered a null-terminating byte in the source string, it'll stop copying at the maximum character count that you give it. So it's really interesting. I thought when I read this, it's like, bravo, Microsoft. I mean, that must have been a...

Leo: It shouldn't even be allowed.

Steve: Must have been a ton of work, though, to go back through and change all that.

Leo: But at least that you could kind of do with a global, you know, searching for strcpy and find them all. It was Matt Conover, by the way, from w00w00 who told me that. And by the way, you use a language that is probably the most difficult to do this in, Assembly language, because you have to do this all by hand.

Steve: On the other hand, since I do have to do it all by hand...

Leo: You know exactly what's happening.

Steve: Exactly. And I'm always having to be aware of that. So it's not like I'm tricked into, you know, lulled into a false sense of security by, oh, well, I'm not really going to worry about this. I mean, I've got to worry about everything all the time.

Leo: You range-check everything.

Steve: Yes.

Leo: Matthew Dwyer listens and writes from New Zealand. He's a Kiwi. I've heard you guys mention you don't run a software firewall at all, instead putting your trust behind your external router - or in my case my Astaro Gateway - as a hardware firewall. I have a properly configured external router, but I'm nervous about running without a software firewall. Should I really get rid of the software firewalls on my LAN?

Steve: Well, this is a question we've had before. But it keeps coming up. Like you, Leo, I don't run a local software firewall. Although frankly, after...

Leo: I turn on the Windows firewall. I leave it on. So that's - maybe I've changed my tune. But now that it's on by default in Vista, I just leave it on.

Steve: No, I absolutely agree with you. And in XP - it's on on all of my laptops where I'm running XP. There's no reason not to do it. The problem is that something like one of the big, heavy security suites is just so cumbersome now. I mean, I really hear reports from people talking about how they removed - and I don't mean to pick on McAfee or Symantec, but those are the big guys. They removed them, and their system, it's like young again. So I guess my point is...

Leo: ZoneAlarm has become like that, too, unfortunately.

Steve: Unfortunately it's not something that I any longer can in good conscience recommend. It's certainly gone a long way from 2.61, which was, you know, the version I loved so much. So a software firewall that does outbound blocking is really, I mean, that's the key. Everybody's got incoming blocking, whether from a router or from your existing XP or Vista personal firewall built into the OS. The real issue is outbound protection. And there are lots of people who just love having it on their system. They like the idea of needing to give permission to programs in order to get outbound connections to the Internet, which I fully understand. I'm taking the gamble of being really careful that nothing evil gets in because my whole theory is, once that happens, it's over anyway. I mean, it's too late. So the issue is that.

Now, the one caveat is intra-LAN protection. If none of your machines had any personal firewalls running, and something evil got on one of them, it could much more easily spread to the other machines on your LAN if you had no protection. So I would say at the very minimum use the firewall built into Windows, into XP or Vista. And if you wanted an additional layer of protection, add a software firewall. But I really think, if you protect yourself, you're careful about where you go on the 'Net, what you click on, and what you do, you know, I just - I've never had a problem.

Leo: So there are really two compelling reasons that you might want to run a software firewall. One of them is kind of dubious, and that is for outbound protection. But since many malware programs now look for the firewalls and around them, so forth, and because most people just go yes yes yes yes yes, or as our guest last week said, no no no no no, that's of dubious merit. The one thing that is of value, and the reason I turn on the Windows firewall, is this intra-LAN infection. So if you bring an infected laptop into your office, well, it's inside the router, and so it could infect all the computers unless they have firewalls turned on.

Steve: Exactly.

Leo: So there is a good reason to leave the Windows firewall on. And I don't think outbound protection is - fortunately, I don't think it's particularly compelling anymore.

Steve: Nope.

Leo: Dusan Maletic of Babylon, New York raises a great - and distressing - point: During your multifactor authentication discussion, one comment utterly surprised me. Utterly. Did I emphasize that sufficiently? Utterly.

Steve: He picked himself up off the floor and went over to send us this note.

Leo: You're willing to provide biometrics for such low-priority needs as boarding the airplane. Not only does this info contain essential personal identification - actually it's not for boarding the airplane. I'll explain what it's for. But anyway, not only does this info contain essential personal identification, but it will be stored at such low security level that it's almost certain this data will be stolen or manipulated at the servers for nefarious purposes. A good point. I won't gainsay that. I should mention that the - and by the way, it's not a retinal scan. I misunderstood that.

Steve: Yes, it's an iris scan.

Leo: It's an iris scan. A retinal scan is much more complicated. But an iris scan. And it isn't for boarding the airplane, it's for customs. It's for crossing the border. Although, of course, you're getting on and off an airplane often, but not necessarily. Leading to the ultimate identity theft. How to prove you are who you are if someone alters your biometrics data. Oh, that's - I didn't think of that.

Steve: Yeah.

Leo: Only the most - this guy's been reading too much Philip K. Dick. Only the most demanding security access should be associated with biometrics, if any.

Steve: So I think he raises a great point that I wanted just to discuss here. And that is, as always, there's the issue of risk versus reward. There certainly, you know, you and I were talking two weeks ago during the multifactor authentication episode that, hey, it's like, yeah, let's all give, you know, random people our iris print...

Leo: Good point.

Steve: ...so we don't have to wait in line.

Leo: Not a good idea, right.

Steve: And so the idea is - and again, I think he makes a good point. Consider the tradeoff, the

idea that, well, yes, now...

Leo: But which is easier to steal? My passport or my iris?

Steve: Well, you'd really miss your iris if it were stolen, Leo.

Leo: No, I'm just saying that I think the passport is in even less secure hands. Mine.

Steve: Yeah, well, but remember, that's physical, and physical is always a different domain than electronic.

Leo: True, but this is multifactor. They want your iris and your passport, so...

Steve: That's a very good point. Although I guess the idea being that, you know, somebody could - if you were known to have your iris print on the server, and somebody wanted to keep you from going...

Leo: They could mess with me, yeah.

Steve: ...to Vancouver, you know, they could put some dog's iris scan in there, and...

Leo: It wouldn't match.

Steve: ...chances are it wouldn't match.

Leo: I wonder. Nexus must have some procedure for that.

Steve: Well, and this guy talks about low security level. And of course it's going to get stolen. It's like, okay, well, yes, I mean, certainly after listening to Marc two weeks ago, I mean last week...

Leo: Everything is going to get stolen.

Steve: It's like, aagh.

Leo: Well, and the TSA - this isn't the TSA, again, this is Canadian government, Canadian Customs; and U.S. government, U.S. Customs. But the TSA did, just a couple of weeks ago, lose a hard drive with all of their employees' information on it - Social Security numbers, name, address, everything. So, I mean, the TSA is not so very different from what this is. So he makes an excellent point.

Steve: You know, if they had to replace all those TSA employees because of their personal data being lost, that would be kind of okay.

Leo: I have to go through the TSA checkpoint next week. I'm not going to say a word.

Steve: Okay, cut that from the recording.

Leo: You know, I'm sure there are TSA employees who listen. And, you know, for the most part they're very jolly, nice people. I don't blame them for the stupid laws that require me to take off my shoes.

Steve: True. And, you know, I actually - I say, look, there's no metal in them. Let me just walk through your scanner. It won't even bleep at all.

Leo: Do they let you do that?

Steve: Yeah, they do.

Leo: They still won't let you bring a bottle of water in, though.

Steve: No.

Leo: And, you know, that one baffles me. Don't get me started. Don't get me started. TV Indy of Muncie, Indiana has a question about TiVo operation. Steve.

Steve: Just to change the topic.

Leo: Just a little light, upbeat subject. I'm not sure what you said about TiVo in Episode 90 is correct. Does it really write to the hard drive 24 hours a day? My impression is the 30-minute buffer only exists in RAM. It is only when you choose to record a program in progress that the contents of the RAM are written to the hard drive. Unless the program's being recorded and stored for later viewing, I doubt that anything's being stored on the hard drive. My parents have a TiVo that's been running continuously for nearly five years, and it's the original drive. Would it even be possible for a hard drive to survive that long if the head stayed in continuous motion for that period of time?

Steve: I'm absolutely positive that the TiVo is reading and writing continuously. Our listeners may not know, but I know you know, Leo, that I have three heavily hacked TiVos. I've got original Series 1 TiVos with my own expanded Linux kernel that's able to go beyond the 137 gigabyte boundary. Each of them has a single 500-gig drive. And one of the hacks you can perform on the TiVo is to change that 30-minute buffer so that, for example, it's an hour or two. Especially the Series 1 TiVo has very little RAM, just enough to hold the Linux kernel, and an hour or two of video at high quality is many, many gigabytes of storage. So it's absolutely the case that TiVos are continuously writing and reading to and from this buffer on the hard drive. And, yeah, it does create wear and tear for the system.

One of the things that I do, because I'm sort of fanatic about hard drive life, is if I'm ever unplugging my TiVo, I will manually reset it and then pull the plug because I want to shut down the fact that it's continually writing. Otherwise you risk damaging the hard drive if it's writing while you pull the power. So I manually restart the TiVo, then pull the plug, just because, you know, why not.

Leo: It's really - it is amazing. That's probably one of the hardest environments. Not only is it writing all the time, but it's hot in there. They get very warm, those drives.

Steve: They do.

Leo: There's a little fan, but it's a little fan.

Steve: All of mine have big copper heat sinks with a secondary fan, a heat sink on the back of the drive itself in order to pull the heat out of the drive so that the little case circulation air has a chance to keep things cool.

Leo: That's a third-party add-on you did.

Steve: It's a me party, yeah.

Leo: He's the third party. What, did you cut up a car radiator? What did you do?

Steve: No, I actually took P4 heat sinks. And it's all jury-rigged, but it works great.

Leo: I'd love to see your TiVos. Just there must be stuff sticking out all over. Ted Hosmann of Monterey, California brought up a fantastic point: I just listened to your multifactor authentication episode. Loved it. The process of swiping a badge and using hand recognition to enter a building is the same process we use at my job. I work in a building that embosses over two million credit cards each month. Whoa. And we have about 15 million blank credit cards on hand. Oh, boy. Where is this? A variance to the secure entry that we use is when we swipe our card it opens one side of a tube that we have to stand inside in order to have our hand scanned. Once we pass the hand scan, the tube has another set of doors that open into the building. Oh, that way you can't have somebody follow on.

Steve: That's exactly it. And he's about to explain that.

Leo: One thing that most employees don't realize is, once they step into the tube, there's a weight sensor in the floor to determine if you're letting someone tailgate into the building on your scan. Wow. It would be extremely difficult to explain to the security people why I weighed 350 pounds one day and 200 every day before and after that. Wow.

Steve: Isn't that interesting? And it brings up a point that I had not mentioned. But when I was setting up my account at our server facility at Level 3, coming up on two years ago - it was two

years ago after Memorial Day - they specifically said, you know, you want to keep this place as secure as we do because your crown jewels are inside here, too. So, you know, even if somebody's getting out of a car and walking over with you at the same time, do not let them tailgate. You know, you do your card, do your hand scan. Then, you know, turn to them and say, I hope you understand, you're going to have to do this yourself. Walk through the then-unlocked door and close it firmly and lock it behind you...

Leo: It's hard to do.

Steve: ...to force them to - it is. But Leo, it's happened a couple times, and they understand. They appreciate that I'm keeping them safe and their equipment safe, just as I hope they will do the same for anybody else.

Leo: Yeah. Ryan Sullivan in New York is worried about and apparently having some trouble with WiFi security. He writes, I listen to your show regularly. And when you started talking about wireless security and how unsafe WEP really is, I frantically tried to change my wireless security to PSK-AES encryption that you were talking about. That's a form of WPA. But I have one big problem. Nothing can connect to my router when it's encrypted in AES. My devices can detect the network and then allow me to type in the new password, which I got from your passwords page. But then after that they won't connect to the 'Net. The main device that is annoying, since I use it most often, is my Nintendo Wii. Oh, well, the Wii is notorious for not supporting WPA. I always use it to watch things like YouTube videos on my nice big TV. I'm paranoid about security and would hate to revert back to WEP now that I realize how unsafe it really is. I have no idea what to do. Any help, if it can be spared, would be greatly appreciated. Well, I'll just add that that is a problem with some hardware, including the Wii.

Steve: Yes. Now, one of the things he said, though, which is the reason I wanted to put this in here, is that there are two versions, or essentially flavors, of WPA security.

Leo: I'm sorry, the Wii does do WPA. I was thinking of the Nintendo DS.

Steve: Ah, okay, exactly. But there are still two flavors. I don't know if the Wii does both. Because the original move from WEP to WPA, the people who architected this were very conscious of the fact that they wanted to fix WEP in the worst way because WEP is, you know, the worst WiFi security. But they were conscious of the fact that low-end WiFi cards or hardware might not be able to support AES encryption just due to the substantial processing burden that AES imposes. It is a lot more math to be doing compared to the relatively and extremely lightweight RC4 encryption.

Now, we've talked about RC4 encryption, how it's a pseudorandom sequence generator which is part of the reason WEP is so badly broken. This episode that he was talking about was, you know, how WEP is even more badly broken than we knew, and it's now possible to basically, don't even worry about the WEP key, just crack it in 60 seconds, and you're onto the network. So but the point is that the original WPA spec had something called TKIP, Temporal Key Integrity Protocol, which fixes the problem with WEP while still using the lightweight RC4 pseudorandom number generator to generate random bitstreams for XORing into the user data. Which means that it's very possible that lightweight older hardware and things that this guy may not be able to get to work with the stronger version of WPA, sometimes called WPA2, it would probably still work with TKIP. So what he'll probably find, if he's got a device, is that he may have a choice, and certainly his router will, his wireless router, between TKIP encryption and AES. Some would argue...

Leo: The Wii does both, by the way. The Wii does AES, WPA2, and TKIP.

Steve: Okay. So one thing I would recommend, then, is - okay. So first of all, just to finish that thought, his wireless router will have TKIP and AES. There is absolutely nothing wrong with using TKIP.

Leo: Go ahead and use it, okay.

Steve: Yes. It, too, was designed by the geniuses, the security guys who finally came along and said, okay, we really need to stop using WEP. Let's fix WEP completely. And they did this in WPA, even in that first version that did not support the arguably stronger AES. Again, TKIP is all you need. It's all I use. And you know how frantic I am about the whole issue of WiFi security.

So the other thing I would tell Ryan is try using a simpler key first. That is, I love it that he's grabbed one of those humungo-jumbo 33-character monsters from my passwords page. But it could be that some of his devices are not properly hashing that into the WPA key which is actually used. So, I mean, try using your dog's name for ten minutes. It won't hurt you to just try it for ten minutes. No one's going to do a brute force attack on you in that length of time. Or just make seven or eight random characters. That's even better. You know, your initials, your three best friends' initials or something. Because that's certainly going to be plenty safe. Try setting things up with a simple key just to get everything working. Then experiment with strengthening the keys, making them longer, making them contain more complex characters, until you discover what's making one of those devices break. And then back off to the best key you can use for that device. And you're still going to be safe enough.

Leo: So the most important takeaway here, though, is it's perfectly fine to use TKIP.

Steve: Yes.

Leo: The key is WPA. You don't have to worry.

Steve: There's no known weakness in TKIP. People like AES because it's, ooh, it's new.

Leo: Super secure.

Steve: It's fancy. But it's like, okay, RC4 actually, once you allow it to get going, and it spits out 256 random bytes, from then on it takes something like 2 billion bytes of analysis of what RC4 is generating even to know it's not truly absolutely random, even to be able to detect that it's a pseudorandom sequence generator. I mean, it's really good. It just needs to be used right. And these problems were solved in WPA, even with TKIP.

Leo: Russell Gadd in the U.K. writes: Steve, you said you carry around your WPA WiFi password in a little file on a USB dongle, which you then cut and paste into Windows or your WiFi router when it asks for your password. As you are exposing this file to your system for a while, and also passing the password through the clipboard, isn't there a risk

of it being cached somewhere on the system which could later be found by an attacker? Isn't this just as risky as storing your passwords in a plain text file on your PC? I ask you, Mr. Gibson, maybe you regard this as no real risk, as someone getting into the files on your PC has already compromised your security? To me it just feels a little less secure to open the file with some text processor and do this cut and paste, rather than remembering some difficult-to-guess password and typing it in directly. What do you say to that, Mr. Gibson? Mr. Security Expert?

Steve: I mean, he's got a point, that I am using the clipboard. Now, as it happens, I always overwrite the clipboard after I've done a cut and paste. That's just sort of something you learn from - I picked this up, I think actually it was TrueCrypt that first alerted me to the notion because those guys really thought of everything. And they deliberately cleared the clipboard as one of the things that they're taking care of.

So, yes, I am using the clipboard, rather unavoidably, in order to cut and paste my 63-character WPA password into whatever OS or router I'm configuring. There just isn't a way around that if I want to use a really hairy password like that. And you could argue that, even if you type something in, I mean, we're hoping that the other software does the right thing with the password you give it. Sadly, it turns out there are many instances of security software storing on the hard disk, in the registry or one place or another, unencrypted passwords. So again, this process is all moot.

Now, he brings up the point that, you know, am I not worried about this because, once something's inside, then all bets are off anyway? It's like, well, yes. I mean, the point of wireless is that you're using your wireless password to protect your wireless domain so that nobody on the outside can connect to your network. So there is no way for anyone to get your password unless they're on the inside. Now, backing off from that a bit, as Marc told us last week, Marc Maiffret of eEye, he's all about all these basically internal exploits. He recognizes that that's the next frontier for exploitation, which really is what we've been talking about, and I have been saying, harping about browser scripting for so long. I mean, clearly the problem now is people inviting these things into their systems unwittingly. So you would like to not have software that got inside able to find your password and do something with it.

So all you can do is all you can do, which is, you know, get a good password and eliminate it from the clipboard because there have been exploits, in fact, where malware will export your clipboard, see what's on your clipboard and send it out, just because oftentimes there's something tasty that's been sitting there for a few hours.

Leo: All right. But I think the real point is that, when we're talking about wireless encryption, we're really talking about protecting the broadcast. If somebody has physical access to your machine, there's a whole host of other problems you've got.

Steve: Yes. And if something's already on the inside, sure, you'd like to keep your wireless password secret. But on the other hand...

Leo: They've got everything else.

Steve: ...that becomes the least of your problems.

Leo: The least of your worries.

Steve: And also notice that it's not like having your WiFi password lets the guy in China - again, I don't mean to pick on China, but that seems to be where all this is happening. The fact that some Chinese guy has your WiFi password probably is not a problem.

Leo: Because he'd have to drive to Irvine to get...

Steve: He's a long way away.

Leo: Right. Or the U.K. in this case.

Steve: It's a much bigger problem that he's rummaging around inside your network from your computer.

Leo: I have in my hands the last question. Joseph Melanson of Syracuse, New York has a great question about what you've called, another coinage, another Gibson special, "Internet Background Radiation."

Steve: IBR.

Leo: IBR. You've mentioned in past episodes about IBR - worms that are no longer a threat, junk floating around the 'Net. Does this stuff cause slowdowns and bottlenecks? Does this stuff cause potential damage to packets crossing the Internet, even if it doesn't infect those packets? I was wondering if there's a way to build some kind of system to attract this stuff, trap it, and keep it from getting back on the 'Net. Could programs be written to seek these worms and viruses out and destroy them in the wild?

Steve: It was a great question because it is an interesting thing, you know, anybody who is looking at the logs from their routers, if they still haven't turned them off, will see just all this junk coming down on to their IP. For the most part, unless somebody is really targeting them directly, this is just Internet background radiation. It's some stuff somewhere emitting packets onto the 'Net that happen to have your IP address. And it could well be some old Blaster-infected server located somewhere. It could be some random PC that just, you know, an innocent homeowner has some sort of Trojan installed on, and it's out there hunting for other machines with the same vulnerability it has that let it get into this first person's machine.

So the problem is, while most of the time the source IP will not be spoofed, meaning that they are actually trying to find you, they need a response back to them. So it's not like a denial of service attack, where just the act of sending a packet to you is the point. The point is getting a response. So they're probably not spoofing their IP, meaning that if there were some sort of Internet background radiation cop, you know, somebody whose job it was to go track this down, and if they could somehow get the legal authority to do this because that's the problem, and also you'd need international treaties that allowed this because you'd have to go wherever this IP led you, in theory, yes, you could backtrack all the way up, find the machine and, you know, pull the plug on it. Then everybody else on the Internet would breathe a little happier because there'd be one less source of junk on the 'Net.

At this point it's really not causing a problem. If you look at the rate of noise coming into a random IP, you know, it's a few packets a minute. It's not overloading routers. It's not causing jams. There are certainly heavily loaded routers that are, because of their centralized location, they're dealing with a lot more of this junk than not. But on the other hand, they're much more

powerful, and they have lots of bandwidth. So it's really, at this point, it hasn't grown to the level where it's a huge problem.

And I think we've probably even reached - I bet if you were to chart the amount of this gunk on the 'Net over the last two decades, it would be an exponentially flattening curve. That is, we're probably to the point now where new sources of radiation come onto the 'Net at about the same rate that old sources leave. Somebody finally updates their machine, the machine just gives up and dies, they lose Internet connection, who knows what. I mean, certainly there's going to be some churn in the total source of this kind of radiation. So I would imagine it's about as bad as it's going to get. I don't think it's ever going to go away completely. But I doubt that it's probably going to get much worse. And it's not really at a level where anybody is motivated to go around and clean it all up because new sources are being created all the time.

Leo: Right. I suppose you could have some sort of scrubber worm. But that could cause other problems.

Steve: Well, yes. It's illegal also. Marc was talking last week about the problem he faces or foresees in trying to perform security analyses of web-hosted services on the Internet where he's able to check Outlook Express, for example, to see what problems it has. But if, for example, Microsoft did an Outlook Live version where people were actually using their web browser as their Outlook client, he's unable legally to go probe Microsoft's farm of servers because they're not going to be happy that he's out there looking. I mean, you could imagine, Microsoft sees the work that eEye does, finding and reporting Windows vulnerabilities, as a mixed blessing. It does blemish them, and but now they're forced to acknowledge that there's a problem and fix it in a few months and give us our Patch Tuesdays on the second Tuesday of every month. So the fact that Marc foresees this future where they're not going to be able to fix these things, and there are still going to be problems, that's a little bit chilling.

Leo: Yeah, yeah. Well, we have completed our 12 questions for today's episode. I want to thank you for all your emails and thank Steve for his insight. We can be very grateful to Elaine, who will make a transcription of this so you can read along as you listen. And of course there are 16KB versions along with the transcripts and notes available on Steve's site, GRC.com. And that's also where you'll find his program, SpinRite, the world's finest hard drive maintenance and recovery utility for the last decade or so.

Steve: No, Leo, 20 years.

Leo: 20 years?

Steve: Yes.

Leo: So you have updated it.

Steve: Oh, yeah. We're at version 6.0. Yeah.

Leo: I'm just teasing you. I don't want anybody to think it's a 20-year-old program.

Steve: Okay. No no no.

Leo: GRC.com. He cleans it up all the time. Always working to make it finer. And by the way, while you're there, check out Steve's free security updates and utilities. There are so many good little programs there, including ShieldsUP, which will be at its 50 millionth user in a day or so. There's SecurAble, the new one, and DCOMbobulator, Shoot The Messenger, Unplug n' Pray, many, many more. GRC.com. Any final thoughts, Mr. G?

Steve: I think we're set. I'm looking forward to, as I'm sure our listeners are...