SECURITY NOW!

Transcript of Episode #154

Listener Feedback Q&A #46

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

High quality (64 kbps) mp3 audio file URL: http://media.GRC.com/sn/SN-154.mp3
Quarter size (16 kbps) mp3 audio file URL: http://media.GRC.com/sn/sn-154-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 154 for July 24, 2008: Listener Feedback #46. This show and the entire TWiT broadcast network is brought to you by donations from listeners like you. Thanks.

It's time for Security Now!, Episode 154, our sesquicentennial plus four. I'm just going to keep saying that, Steve. I just like saying that word.

Steve Gibson: I think it's plus two now.

Leo: Oh, plus two, even better.

Steve: We're zeroing in on it, yeah. Two episodes to go.

Leo: So Steve Gibson is here. He is the man at GRC.com, the Gibson Research Corporation, where you'll find such great things as ShieldsUP!, which has become world-famous as a way to test your firewall or your router when you first get it working. Everybody goes to ShieldsUP! at GRC.com. He's also, of course, the creator of SpinRite, the world's finest disk recovery and maintenance utility. How are you today, Steve?

Steve: I'm great. It's great to be back with you for approaching the end of our third year, Leo.

Leo: Wow. Questions and answers today, as always on the even-numbered shows. And we've got some really good ones, including, I might add, the Sad and Disturbing Truth of the Week and the Creative Writing Award of the Week. I love how you get these awards, Steve.

Steve: Well, they're deserved, as our listeners will hear when we get to them.

Leo: Oh, good. I can't wait.

Steve: We do have first some security news.

Leo: So, Steve, what's happening in the world of security this week?

Steve: We've got three interesting things. First of all, there are new updates to Firefox, both v2 and v3. 2 went to 0.16. Seems that every time I talk about this there's another subversion of Firefox 2. Last time it was .15; now they're at .16. So I wanted to let our users know that they want to check in with Firefox, those who are using it. I have to say, Leo, I'm very impressed with the people who use GRC. I don't know if this is an overall trend, but more than half of GRC's visitors are using Firefox over IE.

Leo: I think we're, like, almost 60 or 70 percent on TWiT. It's great.

Steve: Yeah, it is. The Firefox browsers, especially v3, I'm very impressed with. They fixed some bugs that are still in v2 and that I think there's very little chance they're going to fix in v2 because they're more architectural problems, not little cosmetic patchy things. So 3, as soon as people are comfortable with the move to 3, I never want to push anyone to move to a new major version of something...

Leo: Although I just saw a very interesting study - oh, where did I see it? - that said what a surprisingly few people were using the most recent version of their browser. And I think you and I would both agree that that's a pretty important thing to do is update.

Steve: Well, for example, the reason these were both fixed was to fix that blended threat we talked about a few weeks ago that involves having both Safari and IE or Firefox on the same system. Remember that Safari by default downloads things to your desktop. And unfortunately Windows searches for DLL files in a really brain-dead sequence which involves your desktop. So it's possible for someone to cleverly, by hoping that you've got Safari and a non-Safari browser both on your system at the same time, to get Safari to download something to your desktop which then Windows will discover when you're using a non-Safari browser and get it to run this code. So that's what was fixed, that and a few other little less significant things, that was fixed in this version of - in these most recent versions of Firefox.

Leo: This study was from the Swiss - I found it now - Swiss Federal Institute of Security, IBM, and Google. And 60 percent of people use up-to-date, fully patched web browsers. But that means there's 40 percent out there who are running kind of vulnerable. I like IE7, though. I think IE7's pretty secure. Do you have an opinion on that? I know you've always been an IE user.

Steve: I have, and I have to agree with you. I think that, I mean, it is - my annoyance is that it takes Microsoft so long to fix this stuff. I mean, it's years, years, years. I mean, and I think the greatest comeuppance for them is that they have given an alternative browser like Firefox such a huge window of opportunity to come in, arguably with a much more secure solution than IE. I mean, it's Microsoft's own fault that Firefox has the huge market share that it does. And you've got to know that Microsoft is not happy about that. They fought like crazy to get IE to the position it is now over Netscape back in the day. And here it's because people have left IE because of all the problems it's had for so long. It's just shocking to me that it's taken Microsoft so long to fix these things. But yes, it is substantially more safe. The fact that you get - you've got good built-in pop-up blocking; the fact that you get a notice when an ActiveX control attempts to run. And so, I mean, many problems are now no longer problems under IE. But it's got such a bad reputation which is now going to haunt it for the next who knows how long. They may never recover from that.

Leo: Well, yeah, it's true. And I think a lot of people, once they use Firefox, like the extensions so much, like the capabilities so much, they don't even go back to IE. So it really doesn't matter that IE is now safe. It's too late.

Steve: And Firefox is a - as I was just saying, Firefox v3, and even 2, but 3 even more, is a beautiful solution.

Leo: Oh, yeah, I agree. Things like NoScript, which really I know you love, Adblock, some of the things that you can do in Firefox you just can't do in any other browser, really makes this a good choice.

Steve: Right, right. A report that came out on the 'Net from a security monitoring organization had something to say that I thought was rather humorous, not really good news. It was a report on the changing structure of cybercrime organizations. And this report noted that there has been a huge fall in the price of compromised financial details.

Leo: It's cheap now. We're having a fire sale.

Steve: And guess why? It's due to an overabundance of supply.

Leo: Yeah, yeah. Oh, boy.

Steve: There are so much now, bank account details including PIN and so forth, that the price has fallen because there's just so much of it available. Used to be that the bad guys would pay about a hundred dollars for bank account information including the PIN number. It's dropped now between 10 and $20 just because there's so much of it around. And the bad guys are now, like, well, that's sort of more commodity. They're now looking for other sorts of information that they can get higher margins on because they would like something that's more valuable because bank account information, ah, that's just, you know, not a big deal anymore.

Leo: Anybody can get that. That's amazing.

Steve: Also in the news - and we may cover this more, in more detail if necessary. But Bruce Schneier, the well-known security researcher at Counterpane, founder of Counterpane Security, worked with a bunch of students. And it's actually the students he says who did the heavy lifting. They demonstrated that systems which are not fully encrypted, that is, that do not have whole-drive encryption, but which instead encrypt portions of the hard drive, actually don't succeed in the plausible deniability, for example, which TrueCrypt attempts to offer.

It's just a fact that the OSes have not been designed to properly sanitize the debris. So things like funky drive letters that are lingering in the registry and in various history lists and things, if a forensic research acquired a machine which, for example, had one of these - one of the features of TrueCrypt, for example, is that you're able to create like a hidden partition in the unused space of an encrypted partition, and the idea being, oh, you can say no, no, no, here's the keys to the partition that I've got encrypted. And you have plausible deniability that there's actually another partition hidden behind that one.

Well, it turns out, unfortunately - and again, this sort of makes sense when you think about it. Accessing that hidden partition brings it into currency in the OS. And there are enough little traces left behind in the OS that a good forensic researcher would be able to determine that, uh-oh, there was another partition there that now has disappeared. And then I guess they twist your arm harder and make you give them the second set of secret keys. So just a little heads-up. Again, a fully encrypted partition, because the entire thing is encrypted, doesn't have this vulnerability. On the other hand, the vulnerability that is created is a feature that some users might want. It's worth noting that it's now been demonstrated that that plausible deniability, it's broken.

Leo: Interesting.

Steve: Also, I didn't intend to go into great detail, and I still don't intend to go into great detail, in another one of these heinous ISP-sponsored third-party spying outfits. We've talked of course now for several weeks on our non-Q&A episodes and even in some of our Q&A episodes about the Phorm system. One of the other ones we mentioned in the very first week of this was NebuAd. Well, some of the people in the GRC newsgroups mentioned that they have received updated terms and conditions from their ISPs notifying them that they're going to be using NebuAd. And in reading through postings, submissions from our listeners, in order to select the questions for today's Q&A, I ran across another couple Security Now! listeners who wrote that when we talked about NebuAd, that kind of rang a bell in their memory. And they went back and looked at some updated stuff that they - fine print that had been sent to them. And sure enough, their ISPs were notifying them that they were going to be using NebuAd.

So I did want to just mention the technology very briefly that NebuAd uses because it's way annoying. It does not perform the script-free multi-browser cookie dance that I described a couple weeks ago where remote sites are being faked by the equipment hosted in the ISP's facility which intercepts your attempt to access a remote server and instead bounces you to a NebuAd server and makes your browser dance around a few times using redirection in order to basically salt everywhere you go with their own cookies. Instead, this does what Phorm was trying to do in the '06 and '07 pre-release testing with British Telecom, BT. It actually inserts script onto the pages you download. It inserts JavaScript.

Essentially what it does is it spoofs a final packet into the stream coming back from a web page. When it sees the web page ending with a backslash html closure in the html code, that triggers it to inject its own script reference at the end of the pages you download. So it is actively modifying the pages you receive from websites you visit. And that JavaScript includes a URL to a site owned by NebuAd that causes your script-enabled browser to then go and fetch whatever scripting code they choose to be inserting into your browser page at that time. And of course your browser then runs that script, and that's the way they succeed in planting cookies on your browser and enabling tracking.

So it is really bad and annoying also. Does it in a different way, but essentially achieves the same thing. And they've got the same opt-out mumbo-jumbo. But here we have an instance where ISPs are sending out fine-print boilerplate that people are not reading, not giving people the opportunity to explicitly opt in, but rather requiring them to opt out if they don't want this kind of third-party tracking going on.

Leo: They just keep trying, don't they, Steve. It's amazing. They won't give up.

Steve: It's a bad trend. And lastly I wanted to mention that, since we last talked, Service Pack 3 of Windows, XP SP3, switched from being a separate sort of called-out notice, if you've got your system set for downloading but notifying you for - since the release of SP3 there was like a separate notice that was offering you SP3. And as many people know, probably everyone who's been listening to the podcast, there have been lots of problems with SP3. Many people have no problems, which of course is why it's still out there. But there have been all kinds of instances of selective problems occurring in XP after the installation of SP3. I had two different problems. My tech support guy Greg had a problem. I think I heard you mention that you had a problem.

Leo: Yeah.

Steve: With a post-service pack...

Leo: I have one machine it just, I mean, to its credit, it tries to install it, can't install it, rolls back. I'll have to figure out why, though, because it's going to keep doing that now, I guess.

Steve: Well, that's why I'm bringing this up. It turns out that Greg, my tech support guy, brought this issue to my attention. And I put into Google - thank you, Google - "service pack blocker tool" or something like that.

Leo: Ah-ha.

Steve: Turns out Microsoft themselves offer a very small, simple, lightweight executable, downloadable from their site. Just put in "service pack blocker tool" into Google or into Microsoft's own search. It's easily findable. What this does is simply set a bit in the registry. So you don't even really need the tool if you are, like, a registry hacker kind of person. But part of this is a very small, a little 10K, I don't think I've ever seen Microsoft do anything for 10K, it doesn't have a fancy...

Leo: It's probably just a registry modification.

Steve: It's just a little command-line EXE which basically puts a DWORD variable into the registry telling Windows Update not to update this service pack. So I wanted to let individuals know, also people who are responsible for, like, corporate updates or even small office groups, that it is possible - first of all, that this Service Pack 3 has switched into this mode. And if they feel as I do that I don't know, it's still not time for SP3, you can keep Windows from sneaking it in behind your back.

Leo: We should point out that anybody who does that, though, should be a security wiz and know what to do to protect themselves against the patches that SP3 is adding; right? I mean, it's...

Steve: Yes. So far the patches that have happened since have patched around SP3, yes, and haven't required it. There was some question about whether this most recent DNS patch might be requiring SP3.

Leo: Oh, did they put out a patch for that DNS flaw?

Steve: Yeah, Microsoft actually beat everyone else to it. It was in the last...

Leo: Oh, that's right, they did it Tuesday, yeah.

Steve: Right, it was in the last round. So that's been out. Of course that is our serious propellerhead, really fun, techie topic for next week is how DNS spoofing works.

Leo: There was an interesting post, I don't know if you saw it on Slashdot, on the IT part of Slashdot, with a guy explaining - because you know Dan Kaminsky hasn't really revealed the details of this DNS poisoning exploit. Somebody apparently figured it out, and it's actually been published now, which means people will be using it now.

Steve: Oh, absolutely. So it'll be interesting to see. And lastly, I've got a great, fun SpinRite story to share with our listeners. This is from, I think, a Security Now! listener, I'm not sure, a Steve Balaam. He wrote just recently, says, "I bought SpinRite 3 around three years ago and made a CD from it, and then completely forgot about it. Just recently I was composing music for a game..." - so he's a game music composer. And he said, "...and all of a sudden my brand new, non-networked, non-backed-up machine started to make some very strange, irregular noises from the hard drive. Fearing the worst, I grabbed my high-capacity portable drive and tried to back up all my files. Unfortunately, the hard drive then froze."

Leo: Oh, boy.

Steve: Uh-huh. "And when I tried to reboot, I could not then even get past the POST sequence, the so-called Power On Self Test the BIOS does. I feared the worst, as it seemed to give all the indications of complete hardware failure, and was mentally preparing myself to write off six weeks of hard work. Then I remembered SpinRite, grabbed the CD, and set the BIOS to boot from it. Amazingly, SpinRite found the hard drive. And so with my heart in my mouth I selected to recover the data. Six hours and many alarming graphs posted from SpinRite later" - you've seen that Dynastat...

Leo: I was going to say, that's Dynastat, yeah.

Steve: Yeah. He says, "And many alarming graphs later, it said it was done. I took out the SpinRite CD, pressed the reset, and waited. As if by magic, the PC booted. I logged on and found everything was as it should have been. I have since added another drive for resilience and will now periodically run SpinRite as preventative maintenance. This program truly is a lifesaver, and probably the best money I've spent in a long time. Thanks for a great product." So thank you, Steve, for the great report.

Leo: Isn't that nice. Steve, are you ready for some questions, laddie?

Steve: Aye, captain.

Leo: We've got some good ones for you. Starting off with John R. Baskwill at Penn State Harrisburg. He has all the details. He says: I was listening to some past episodes of Security Now! recently, when I heard a question from a Penn State student concerning ShieldsUP!. Since I work for Penn State - wow, now this is getting the answer from the source here - I thought I'd try to clear things up a little bit here for you. He says the student was concerned that the IP address on his computer was the same as the IP address that ShieldsUP! was showing, which means there must be no NAT being performed; right?

Steve: Yup.

Leo: Right, right. Well, during your answer you said that they, Penn State, must have a big network where they can afford to give individual public IP addresses to the students in their dorms. You were absolutely correct, sir. When I first started working for Penn State I was surprised by the large size of the address pool the university owns. Each student living in the dorms must register his or her MAC address with the university. A lot of universities do something like this. Obviously they want to keep people off of their network who aren't students. But it's a tricky thing to do. So here's how they do it at Penn State. So they register their MAC address, which is locked to a specific port, interestingly enough.

Steve: Yup, so there'll be a data switch which is intelligent, and you're able to say only allow this low-level Ethernet traffic from this MAC address.

Leo: That's really good. And each MAC address is assigned its own IP address via DHCP. That's a public IP address. So that's why the address on the computer is the same address ShieldsUP! displays. A little different than a router at home. It doesn't use a private pool of addresses. It's actually giving up the public address to each - you've got a static IP address if you're at Penn State. Also during your answer you and Leo said that since no NAT is being performed to provide stateful incoming security, the university could have a firewall which is blocking all inbound traffic. That is true. All data ports in the dorms are behind a firewall.

You also indicated the setup as described by the student made you a little nervous because a student could put a NAT router in his or her room for additional protection, or should put a NAT router. That's not true. He says the only device a student is allowed to connect to the network is that single personal computer. That makes sense because the router would have a different MAC address. They may not connect hubs, routers, print servers, terminal servers, or other network devices. There are other limitations the students must adhere to, including bandwidth limitations. The students are encouraged to read the Housing Connection Agreement before signing it. Additionally, the students must watch a short video that details what they may and may not do while on the Penn State network. Any inappropriate or illegal activity is traced, if circumstances dictate, by the IP address. That's why they use a unique IP address, because it belongs to that individual student. Does that clear things up a little? Go Nittany Lions.

That's great. I mean, I think most universities have to deal exactly with these problems. And this sounds like a good solution.

Steve: I think maybe part of the reason - I don't know how Internet-savvy the students are. But if they, certainly like our own listener who originally wrote in with this question was asking, if they realize that the IP they've got is static and assigned to them, then it seems to me they're going to tend to be more responsible if they don't think that they're hiding behind a NAT router, no one can trace them back behind the router. If they're doing things, like things that are in breach of the university's policy, for example the MPAA finds their IP address out sniffing around, downloading lots of movies, and the MPAA wants to stomp on them, it's like, okay, this IP address owned by Penn State has been found to be downloading this content. Well, that's directly traceable back to a given student. So maybe the students understand that.

Now, it's certainly the case that this is a pretty much locked-down system. It seems to me, well, I don't know, maybe a little overkill because this means, for example, that a student can't take their laptop from their dorm room to somebody else's dorm room and, like, both be online on that other person's connection. So, I mean, I understand certainly universities are crazy about student behavior on the network. This really does create accountability, which I'm sure is good. But one of the things I'm sure John who posted this knows, and many of our listeners probably know...

Leo: And that they're all saying right now in their heads...

Steve: Yes. And that is, you can certainly change the apparent MAC address of a router to emulate the MAC address of a PC. In fact, that's a feature that all NAT routers now offer, specifically for this reason. Some ISPs lock their subscribers' MAC address to a given machine because in the old days they were sort of fighting this whole idea of NAT routers and, like, well, no, we want you to buy five IPs from us, if that's what - you're going to be using five machines. Well, good luck. I mean, from a networking standpoint, a PC with a firewall is indistinguishable from a NAT router. And so all you would have to do, and I know there are Penn State students who are aware of this, is just copy the MAC address from their PC into a router, plug it in, the university is going to be none the wiser.

Now, the university I'm sure knows, one of the things this certainly allows is for, again, enforced accountability. It is probably the case that you could detect, if you really were concerned about it, whether a router was at the endpoint or a PC because there's going to be different behavior outward facing from a router than a PC. But again, that would be - that would require some very sophisticated equipment. And I think mostly what the idea is, a student would have to be clearly violating the agreement that they signed as part of getting access or agreed to as part of getting access to the university network if they were doing this. And again, it certainly enforces accountability. But we never really talked about DHCP, as I'm listening to this, and he's talking about assigning specific IPs by MAC, DHCP is a far more powerful protocol than most people are aware. I think we're going to have to talk about that sometime soon.

Leo: Oh, I'd love to because we all use it. But I think we use it in a very kind of simple way.

Steve: Yeah, I mean, it does, actually can do many cool things more than just giving you an IP address.

Leo: Oh, I'd love to know more about that. That's great. Yeah, I wonder how many kids, first thing they do, they check into the dorm room, put a WiFi access point in there, spoof the MAC address, and provide bandwidth for the whole dorm?

Steve: And I guess my feeling is that really the - and I'm saying nothing negative about Penn State. I mean, we've got John, a Security Now! listener, I don't want to upset him. But what they've done seems like so onerous, I mean, so restrictive that you're almost forcing students to break out of that jail because you're not letting them, for example, have their friends over and all plug their laptops in, in order to do their - get together and do a little homework.

Leo: And as you know, the more onerous you get, the more likely somebody's going to be figuring out ways around it.

Steve: Exactly.

Leo: Let's get to another question. Christos Kirst in Huntington Beach, California wants the stat of his net. Okay? Okay. Steve, first off I want to say I own SpinRite. I've talked at least four people into buying it, as well. It's amazing. Also your podcast keeps my drive interesting. Always great info that's making my network more and more secure. My question is this: When I run the DOS command - actually you could run it on any machine - "netstat -an" on my Windows 2000 server, I get the normal info. But then what gets me worried is this UDP thing. It shows UDP 0.0.0.0:65518 *.*. Should I be worried about this? What is it?

Steve: Well, okay. What that says, first of all, netstat is a command universally available from the early days of UNIX machines, which were the first machines to be doing networking like this. It's a command, very useful command that I often run in a DOS box. It is a DOS-style command, so a command line-style command. Netstat -an actually happens to be my favorite prompt, my favorite command line...

Leo: What does it do?

Steve: ...switches also. It gives you just exactly the display that you want. It shows you all of the various ports which have something going on, that is, that they're open and listening, you've got established connections, they may just have been shut down, in which case they're in time-wait mode. You can really sort of - it's a snapshot into the networking status of your machine. Now, this one command line, or this one example that Christos cited, where it says UDP 0.0.0.0:65518, what that says is that something, some process in the system, has opened and is listening for incoming traffic on that very high-numbered port. Ports only go up to 65535. So this is - and he's at 18, which is just below the upper ceiling. So this is not something that just said give me any UDP port. It apparently said, I want one kind of way up out of the way, maybe sort of trying to be obscure. So especially on a server machine, which tend to be rather lean and mean and don't have a whole bunch of random applications running, this is something to kind of be worried about.

Now, the problem is that Windows 2000's netstat lacks a feature that was added in XP which is very useful. In XP there are additional commands. There's a "-b" command-line option which will show you the name of the process which has opened the port, which would immediately allow you to determine what it was in your machine that had done this. The concern is that something may have gotten in there, and it's set up shop. I mean, this is what a trojan does is traditionally they would open a listening port and wait for someone to come by and connect to them, discovering that they were installed on that machine.

So I don't want to be alarmist. But this is certainly something that says - that 0.0.0.0 is essentially wildcard. It's like *.*, you know, like ****. For example, when I do a netstat on my XP machine, I'll see a whole bunch of things that are 127.0.0.1. That's a sort of a special reserved network block, the whole 127 block is. But 127.0.0.1, that's sort of shorthand for this machine. So what that says is that something has opened a port on the local machine, that is, it's listening for other things on this machine that want to use the networking system to talk to each other. And this is something that UNIX machines have done since day one. And it's a practice that has become very commonplace, even on the Windows side. So those are not something to worry about. Those are, for example, I saw that Firefox has opened such ports. IE has opened such ports. Many other things that I've got running have said, like, okay, we're going to use my system's own networking system to talk among these processes. But the fact that it's 127.0.0.1 means that that port will not accept incoming connections from other IP addresses. But this 0.0.0.0 will. It's basically saying, I'm open for business, I'll accept incoming data from anybody.

Now, having said that, if this machine is behind a NAT router, as we know, that provides incoming protection. So that won't be accessible from the outside. And hopefully, if you're running a server, you are absolutely behind a firewall. And rather than in the old days, where you would block things that you knew were bad, the only way to configure a firewall these days is to specifically open incoming ports that you know you want. So, for example, he probably has 80 open for allowing web traffic, maybe port 443 if he's also allowing SSL secure connections to a web server. I don't know what kind of services he is serving on this Windows 2000 machine. But my point is that, even if this were something bad that had, like, snuck itself up into the top of the port space and was hoping to receive word from mission control somewhere out on the Internet, there's nothing going to get into that port because certainly any sane configuration today will have blocked that.

So part of the good news is, even if this were something bad, there's not anything it's able to do. But you do - I would say this is suspicious enough that you ought to go about figuring out what it is. Now, there are some free utilities available for Windows 2000 that do this kind of port mapping. In fact, this is one of the features that one of the Sysinternals apps offers.

Leo: Autorun. Autoruns.

Steve: No, autoruns is things that start up a system.

Leo: The process program, what's it called?

Steve: No, it was a networking-based tool. But one of the things that...

Leo: Current ports. Is that it? I'll look it up while you talk. You talk, I'll look.

Steve: Anyway, there is something that the Sysinternals guys, I know that Mark

Russinovich added that feature that would map open ports back to the processes. It turns out there was no solid way to do it under Windows 2000, so it took lots of reverse engineering and system hacking in order to do that. But that's of course what Mark Russinovich was known for, and that's how he acquired his rep, and ultimately why Microsoft acquired him. So anyway, I don't know the name of it. But it's easy to find. Just...

Leo: If you Google Sysinternals, you'll go right to the site. And I'm just trying - there's some process tools. There's Proc Explorer, PsTools, PsFiles, see what files are opened remotely. TCPView? Is that it?

Steve: There you go, that's it. TCPView.

Leo: Active socket command-line viewer. Oh, that's cool. Look at that. So that tells you what sockets are open, what's going on. That's very cool.

Steve: It's very much like what Microsoft added to XP and Vista. And I'm sure that Windows 2003 has it, too, because of course Windows 2003 is really just XP moved forward further. I'm sure that's a feature they added to their stack. But Windows 2000 doesn't have it. So you need to use a tool like TCPView in order to do that.

Leo: 94 kilobytes, too, in the true Steve Gibson style.

Steve: Ah, very good. Nice...

Leo: Mark's good. Mark's real good. T.O. Galloway in Prince Frederick, Maryland wonders who's watching. I just finished listening to Episode 152 of Security Now!. Yeah, I'm behind. Not that bad.

Steve: But not far.

Leo: Not far. You'll catch up. And I got to thinking about the guy in South Africa that had the router admin sign-in that was open on port 80 of his router. Oh, yeah, we talked about that.

Steve: And by the way, I never mentioned this, but other people have that, too. So our asking people to use ShieldsUP! to check to make sure that port 80 was closed, a lot of people discovered that they've got bad UI in their routers.

Leo: That's terrible. T.O. says: I have a couple of network security cameras that monitor my home. I travel a lot. It's nice to monitor what's going on while I'm gone. But these cameras use port 80 to stream the video out. By opening port 80 on the router, am I exposing my router's admin remote sign-in or anything else on my computer to the possibility of having it hacked? Could I use some other oddball port, say something like 41 or 41,587 to stream the video out through and get the router's sign-in still hidden? How high up do the port numbers go that are commonly used? Thanks to both of you and Leo for the valuable information you provide. It's necessary. And thanks for SpinRite. It's saved my bacon more than once. And P.S. to Leo: To kill two birds with one email, on a recent episode of The Tech Guy you made some recommendations for network cameras. The ones I use are from 4XEM. Oh, good. Their quality, I think, equals or exceeds Axis, and the software they provide for free download is excellent, just for future reference. I will check those out. Thank you, T.O. 4XEM. So port 80 is open because those little cameras are really web servers.

Steve: They are web servers. However, okay, here's what's happening. In order for him to be connecting through his router on port 80, he had to have set up port forwarding on his router, so that any incoming connection requests from his browser when he's out wandering around somewhere would be forwarded through his router and either go directly to his webcam hardware, if the cams themselves are network-enabled; or he's forwarding it to a computer, where he's got a bunch of these webcams hooked into USB ports on his computer and running the server software on a PC behind the router.

So there's two issues. First of all, he's safe from his router's admin because by forwarding port 80 to something, to hardware, either his webcam hardware or to the PC, it's not having a chance to touch his router. So that problem he need not concern himself with. And in fact the solution that we suggested for people who find that their administrative interface on their router is open to the world is to forward port 80 to the twilight zone, send it off to some IP behind the router, 192.168.0.222 or something, send it to nowhere, and that ends up just essentially stealthing that port. So that's a nice workaround for people who've discovered that their routers are not properly blocking incoming connection requests to the router on port 80.

However, it's worth also noting that he mentioned this 4XEM company and likes their software. The vulnerability is that maybe their software has a problem. And so basically you're running a web server which is listening for connection requests on port 80. It's dicey. I'm not saying, you know, to be overly concerned. But you are - you're trusting a service which you are exposing to the Internet. And the universal rule is hackers will find vulnerabilities in exposed services.

Leo: That's true anywhere. Anytime you're running a service of any kind, you're relying completely on whether that service is safe. And look at this BIND service that we were just talking about, the DNS problem.

Steve: Right.

Leo: Lot of people run their own DNS servers. Well, when you're running a server, you have to trust the server software. And if there's a problem, you're at risk.

Steve: Yeah. And so the advantage of sticking with, like, IIS, although believe me it had its problems...

Leo: But it's been banged on, anyway.

Steve: ...in the heyday. Yes. Yeah. Or a big name - or like Apache, for example, that has just had all the dust beaten out of it. The advantage of one of those mainline servers is you get all the benefit of the pounding on that it's had. It's trivial to write a web server. And so the concern is that they hired some random programmer off the street whose job was to get this thing working in the afternoon because they're going to ship it tomorrow...

Leo: Now, we're not saying this is the case. It's just the risk.

Steve: Exactly. And but it's always the risk when you're dealing with any service that hasn't had the benefit of being seriously pounded on. And, I mean, look at the troubles that real servers have, and imagine the challenge of just someone writing one in the afternoon and saying, oh, look, you can see yourself wherever you are.

Leo: Now, Mark Thompson has written a little web server. But I trust Mark.

Steve: I don't.

Leo: You don't?

Steve: No. I mean, he would never do anything malicious. But it is just so...

Leo: You might make a mistake, yeah.

Steve: It is so difficult to - you have to, I mean, literally every line of code you write has to be written with security in mind. I mean, you really have to be challenging yourself step by step. And frankly that's just not most people's orientation. If you're in the security business, if that's where you live, that's how you think. I mean, when I'm doing stuff for my site, that's all I'm thinking about.

Leo: So what's the safest thing for him to do?

Steve: Okay. What I would do if this were me, if I had to run unknown software, that is, whose fundamental security I couldn't vouch for, I would give it its own machine. I would give it its own computer, not have it, like, talking to my main desktop. And then I would isolate it to the best degree possible. For example, maybe route it through - put it through its - put it behind its own router and map that port through so that basically it's in jail. So the computer is essentially as isolated from the rest of your network as possible, so that if something, if there was a security problem there, and somebody did exploit a buffer overrun in order to install code, the code couldn't do anything. It would be sitting there going, well, okay, what kind of a network is this? There's nobody here.

There's nothing else on the machine, no valuable information on that machine, and no exposure of the rest of the network from that machine.

Leo: Well, I guess it's something for us all to be aware of because more and more we're using these kinds of cameras and little devices on our system.

Steve: Oh, just wait till you plug your dishwasher or your refrigerator...

Leo: I know, exactly.

Steve: ...into the network.

Leo: We talk about that all the time, the refrigerator that calls home, you know, I'm out of milk. Well, that's a server.

Steve: Yup.

Leo: Oh, boy. Oh, boy. All right, moving along to, let's see, our next question is from Henry Cocozzoli in Troy, Michigan. He wants to make sure he doesn't have a Trojan. If you live in Troy you should have a Trojan.

Steve: I'm glad you caught that, Leo. Because I was chuckling to myself. Okay, he's in Troy.

Leo: As a long-time Security Now! listener and SpinRite customer I wanted to try your Wizmo - which is a great little tool that Steve gives away for free at GRC.com. I was listening to Security Now! 151 at the time, but my Trend Micro Office Scan 8.0 Antivirus flags it as a virus. It says "PAK_Generic.001." I thought you'd like to know. Oh, I bet he knows. If you need more information, please let me know. Keep up the great podcast. Not the first time we've heard this, Steve.

Steve: No, and not the last, unfortunately. There's a - generally what happens is this is certainly a false positive. I'm very sure Wizmo has no viruses. Nobody else has reported one. So some little window of time goes by during which the increasingly annoyingly heuristic nature of AV products will say, oh, look, here's a byte sequence that might be bad because we found a similar byte sequence in something else that we know was bad. So this is a virus. Okay, well, it's not a virus. It's just the problem is there are now so many viruses that there is, you know, a phenomenal number of signatures. It's like a million a year new viruses, or a million total, I think it is, and three quarters of them in the last year. It's just it's been an explosion of this. And now of course the viruses are becoming more polymorphic than ever, so that they're deliberately trying to hide themselves, which means that in terms of this cat-and-mouse game the antivirus companies are having to, like, increase the generality of the windows that they use, these scanning windows. And they're getting an increasing number of false positives.

So it's not just my stuff, it's everybody's stuff that is increasingly being accused of being infected when in fact it's just not. And this will go away. In another, you know, the next time Trend Micro updates their signatures, it'll go away because it never was the case.

Leo: Right. But I think you're right, I think we're going to see more and more of this. Viruses use two techniques - or antiviruses use two techniques to find viruses. The signature technique, where they're just dumb, they're just searching for a matching string that occurred in the virus, that can cause false positives. That seems less likely. But you mentioned heuristics, and that's where they try to be a little smarter and say is this virus-like activity or is this string kind of like the other string.

Steve: Well, yes, they try to be more general because for whatever reason a match doesn't work. Now, my software in the past has had some false positives, even in signature matching, because I've been in the antimalware business, because I've, for example, been, like, in - I think it was like there was something I did with the RPC server where I was shutting that down. Well, I had code that was working in a similar fashion to what the virus was doing, although I was doing it for benign and beneficial reasons, where the virus was doing it for malicious reason. There actually was a valid signature match because we were sort of in the same area of the sandbox, essentially. So those problems went away, too. People - basically people will report those to the virus makers, and they'll go, oops, sorry. And then they'll, like, tweak their signatures to narrow that or specifically put in a special case saying, okay, we know Steve's stuff is not malicious, so we're going to make sure we don't false positive on this.

Leo: Well, anyway, nothing to worry about, I guess.

Steve: No.

Leo: I guess you can kind of tell from the generic nature of the virus.

Steve: I was going to say...

Leo: That clearly doesn't see a virus, it's just...

Steve: Even has it in the name, "PAK_Generic.001."

Leo: That's basically saying, I don't know what it is, but it could be a virus. It looks a little virus-like. Danny Richardson in Fremont, California wonders about whitelisting. Hi, Steve. I saw an article on CNET, thought it might make for an interesting discussion on Security Now!. Briefly, it's a discussion on the future of antivirus software.

Steve: Well, what do you know.

Leo: Well, there you go. And how some companies are moving from a blacklist to a whitelist model. What do you think? Love the show. What does that mean?

Steve: It's interesting. Well, what's happening is that traditional - this is the virus companies' ultimate reaction to the fact that they're beginning to realize this is getting out of control. That is, their existing model has always been find the badware. Now they're beginning to think, okay, that's not working any longer because there's so much badware, there are so many tricks that the malware authors are performing, that we're having a hard time, even with everything we're doing, separating the malware from the good ware. So now they're saying, okay, and this is especially true in a corporate environment. We're going to flip it around. We're going to specifically whitelist software that we're going to allow to run on the machines, and just assume anything not known to be good is bad. Now, this is a problem, of course, for end users because look at all the software out there.

Leo: That's pretty much everything is going to, yeah, I mean, that will limit what you can do very much. Although I guess it might be ultimately the only way they can keep up.

Steve: You can see, you can imagine that in a corporate environment where there are inherently more restrictive policies, it makes sense. I mean, you can imagine, in fact, corporate IT just saying, yeah yeah yeah yeah, let's go for whitelisting. Because essentially it means anything not whitelisted will be absolutely blocked from running. And then of course you've got to go to your IT guy with your tail between your legs and say, uh, why can't I run the Martha's Vineyard mapping application?

Leo: We never heard of that. You can't run it.

Steve: Exactly.

Leo: It's too small. We're not going to support that. Yeah, that's too bad.

Steve: Yeah. So I do see this happening. And it's going to be rough for end users. I don't think that model works in end-user cases. I think - I know that lots of end users feel better having something there sifting through their data. And every so often, I mean, if nothing else, a false positive or two lets them know their antivirus scanner is awake and paying attention.

Leo: It did something, yay.

Steve: Hey, good, maybe it'll find the bad stuff next time.

Leo: Bill Barnes in Charlotte, North Carolina worries about his provider's DNS service. So what do I do if the report at DoxPara.com tells me I'm vulnerable? Remember we talked last week about Dan Kaminsky's discovery that many DNS servers were flawed, and he set up a site, DoxPara.com, to test your DNS. He says: I manage several clients with different ISPs, and a number of them are not yet patched. For most of them, Kaminsky's page displays a different DNS than is configured in my router or was given to me by the ISP. I don't have any leverage with the ISP and certainly don't want to make a public show of the fact that they're putting me at risk. At least my EarthLink DNS did get patched a couple of days after the first time I checked.

I have to say I did it as an experiment on the air, I gave people the URL of DoxPara.com - this is when the flaw was first announced, about a week and a half ago - and asked them to come into the chatroom and tell me. And a surprisingly large number of ISPs were not patched at the time.

Steve: Yeah, it's going to be interesting to watch this. I noted that when I went back and looked just this morning, Dan had put in a "don't panic" message. I think what happened was his little DNS tester freaked out people because he was saying, yeah, you're vulnerable. And they're like, aaaagggghh. Now what? And so it's like, okay, calm down. Literally, like it says something like that, words to that effect. It says - I'm sitting here, I've had it brought up here. He says, "Do not be concerned at this time. IT administrators have only recently been apprised of this issue and should have some time to safely evaluate and deploy a fix."

So, and I will tell our listeners the same thing. This is, even at its worst, it is a highly targeted attack, and it's not like suddenly all of DNS 'Net-wide is going to be poisoned. The idea is that it's possible to confuse a DNS server so that while its cached data is non-expired, it's routing your browser to a bad server. Now, that's not a good thing. That's why everybody in the industry is patching this. But it does mean that it's sort of a focused attack.

Now, what you can imagine the bad guys doing, however, because I've been giving this some thought in preparation especially for next week's podcast, where we're going to really go into this and get our propellers wound up, is that it is very easy to discover servers that are not patched. So as ISPs across the 'Net get their servers fixed, that will bring a heightened focus upon those ISPs that haven't gotten their DNS servers patched. And you can expect a higher incidence of games being played against those servers.

So anyway, we'll be going into this in detail next week. My advice for people who are truly concerned, I mean, I haven't done anything at my end. On the other hand, I'm running a fully resolving server myself, so I'm not dependent upon anyone, like any ISP's DNS server. You could certainly, however, as we said when we brought this up last week, is just aim your DNS at the OpenDNS servers, which have always been well designed and are spoof-proof as much as is possible. So...

Leo: Yeah, they do a good job.

Steve: Yeah. Yeah.

Leo: Nick Cody in Ipswich - that's in the U.K. - wonders why ZoneAlarm broke GRC, darn them: Hi, Steve. I'm a long-time listener to Security Now!, and I think I may have identified a slight problem for ZoneAlarm users like me trying to access your site, GRC.com. I needed one of your fabulous passwords today, so I used IE7 to browse to www.GRC.com. It looked like the site was loading, and then it stalled before ever getting to the main page. I tried it a couple of times, and the same thing happened. It could only get so far and never showed your main front page. So I tried Firefox 3. Same thing. Always ended up hanging while trying to bring in something from GRCtech.com. I wondered if it could be ZoneAlarm throwing a spanner in the works, so I looked at my privacy settings in ZA. He's using v7. Ad blocking was turned off, so it couldn't be that. But cookie control was set halfway up at medium. Turned cookie control all the way off, cleared my cache, restarted Firefox 3, tried again, and bingo, your site loaded straightaway. I closed Firefox 3, cleared the cache, and used IE7 to navigate to GRC. Again, right away, no problems.

I repeated the experiment a few times using both browsers. There's no doubt I can't reach GRC.com when ZoneAlarm's cookie control is set to medium. And this is where my concern lies because according to the settings, "medium" blocks third-party cookies and removes private header information, and you can't reach GRC.com unless these privacy features are disabled. What's going on? Surely there are no third-party cookies at GRC.com. A quick mention of this issue might help those of us who use ZoneAlarm and want to look at your site. All the best. What's going on?

Steve: [Sighing].

Leo: I have a feeling, because that GRCtech.com, that would be a third-party site, wouldn't it.

Steve: Exactly. For the last two years, since I first deployed my third-party cookie testing technology, one of my servers, GRCtech.com, has deliberately been offering browsers cookies from non-GRC.com. This is part of the system I am just on the verge of releasing publicly to notify people when their browsers have third-party cookies enabled. There's a very cool page, Leo, if you take a look at it right now, go to GRC.com/cookies/stats.htm. You'll get a kick out of the - there's a 3D bar chart there. GRC.com/cookies/stats.htm.

Leo: This is based on incoming stuff on your site?

Steve: This is, yes, this is the history of all of the GRC visitors in the last week. And it shows a by-browser profile of which users, by browser, who's got third-party cookies disabled. And it really shows the effect of Safari's disabling third-party cookies by default. It's alone in the industry in doing so. It also shows that our Linux visitors, who are using Gecko browsers rather than Firefox 2 or 3 for Windows, they're also very security and privacy...

Leo: And Opera.

Steve: Yes, Opera has a lot of third-party cookies disabled. Very cool stuff there. So anyway...

Leo: I'm surprised that 2 percent of Safari users have turned on third-party cookies.

Steve: I know.

Leo: I don't know what that means.

Steve: I think that's exactly what's happened is they don't understand the implications of doing so. But anyway, so what happened with - and I don't understand what ZoneAlarm is doing. I mean, it seems onerous to entirely block a site which is...

Leo: Just block the cookie. You don't have to block the site.

Steve: Exactly. It just, I mean, literally, no one can get to GRC.com after upgrading to ZoneAlarm 7. It's been a serious problem. And it's like, okay, I'm really glad I helped ZoneAlarm out in the old days.

Leo: You mean that's the default setting?

Steve: Apparently, because it's zapping everybody.

Leo: Oh, that's terrible.

Steve: Although the days of ZoneAlarm 2.6, which was the last one I ever used and liked, you know, those are long gone. And ZoneAlarm has become a kitchen-sink product that I, you know, I'm not loading it. But I'm going to have to run a copy to figure out what's going on one of these days. Apparently it's also doing script page injection, which is another annoying thing. It's modifying the pages you download. I guess since it's your software, though, and not some third-party gunk at the ISP, that's very different. But I'd like to know what it is that it's installing into my browser pages.

Leo: Wow, that's pretty serious stuff.

Steve: It's gotten pretty heavy-duty, yeah. It's invasive.

Leo: Okay. Okay. Samir Talwar in London, England - by the way, you could presumably say, instead of disabling the, you know, not use the medium setting, but say site-by-site, okay, allow third-party cookies here; right? I presume that you could set it.

Steve: Oh, yeah. Well, yes. Apparently they're, I mean, I know that ZoneAlarm is aware of this because some users have called them, and I've seen reports from them saying that - like them saying that they, like, whitelist GRC and GRCtech. Apparently it doesn't work.

Leo: Oh.

Steve: I know. I don't know what's going on.

Leo: Oh. Samir Talwar in London, England does need third-party cookies: Hi, Steve. There's a reason for third-party cookies. I'm not recommend that you enable them by default, just enable them one by one as you need them. That is, in fact, what I do. I have two set up: RememberTheMilk.com, I use that, it's a very good to-do list manager; and Disqus.com. I also use this. This is the commenting system I use on my blog at Leoville.com. So Samir must be reading my mind.

Both of these hook themselves into other sites, making third-party cookies necessary. In the case of Remember The Milk, it has an iGoogle widget and a plug-in for Google Calendar; and Disqus loads itself into a lot of different blogs. It's true. Although I haven't noticed that. Oh, that's interesting. You would have a cookie from Disqus when you visit my blog. That's where the comments live. Again, not saying they're definitely a good thing, but they're not always bad, either. Oh, and one more thing, why don't browsers treat JavaScript loaded from external sites as third-party code, thus making any cookie access appear to be from that site? Oh, that's an interesting question.

Steve: Yeah, I thought that was a great point, too. Okay. So it is certainly the case that with Web - what are we up to now, 7.9 or something? We're beyond Web 2.0; right?

Leo: I think so, somewhere out there.

Steve: There are beginning to be valid uses for third-party cookies. The so-called "mashups" where you're running somebody else's site's code in, like, an IFrame or in some sort of a window in some other site. Or like Remember The Milk. It makes sense. And so wholesale disabling of third-party cookies I think still probably works for most people. But if there were something like, exactly like Samir suggests, like RememberTheMilk.com, where there's a reason for your - you know, you're cookie and privacy conscious. In general you want to deny third-party cookies. Most browsers allow you to whitelist cookies by site and allow RememberTheMilk.com and whatever site, you know, whatever site you know you need third parties enabled. Normally you don't because you only need to exchange cookies with the site you're currently visiting, not sites you're not visiting. And that's of course what we mean when we say "third party." And we'll be getting into all of this in really interesting, painful detail not long from now. But anyway, so I did want to acknowledge the fact that not all third-party cookies are bad. And sometimes they really do have a use.

Leo: Yeah. In fact, I use them, apparently. Michael Tiller in Canton, Michigan loves his YubiKey. But Steve, he says, I got a YubiKey to play with. I agree the concept's quite nifty. But I think there is one thing that makes it impractical, and that is the fact that it uses a symmetric encryption scheme. The key issue here - pardon the pun - is that for somebody to be able to verify your identity, they must implicitly have the ability to steal. In other words, they'd be able to fake your credentials. It seems to me that the killer version of this device is one with asymmetric encryption. That way I could give out a public credential and use my YubiKey to prove, using a private credential, that I am the same person. This issue came up in the YubiKey forum, and it seems like the summary here is that asymmetric encryption would require more sophisticated, read more expensive, hardware, and that it would require far more information to be communicated in order to be practical. We've talked a little bit about this before; right?

Steve: Yeah. I completely agree with Michael. So what he's saying is that - and this is, I mean, this is an issue with the YubiKey. We've talked about it with this notion of acquiring the private, the secret symmetric key from YubiKey. You could write to them and say I want my YubiKey's secret code because I want to provide it to somebody else to authenticate, I want to be able to authenticate it myself, or whatever. In the process of doing so, and liberating it from them, then you're responsible for it. And anybody else who got it could fake your credential. So the problem is asymmetric encryption is not just a little bit more time-consuming or difficult or number-crunching intensive. I mean, it is so much more than symmetric key as to be in an entirely 'nother order of magnitude. I mean, there are SSL accelerator hardware add-ons for servers that offload that SSL setup process because there is such a burden.

Now, I've talked about how it's much less today than it used to be because CPUs are so much more powerful. I mean, you know, a lot of the connections to GRC are SSL. I force an SSL connection to anyone starting to use ShieldsUP! because I want to avoid ISP transparent proxies that we've discussed before in order to get a direct connection to the user's machine so I can obtain their actual IP address. But, you know, and so I'm not using any hardware acceleration. And that's - it's fast enough. But the point is that there's a little tiny little processor on the YubiKey that is easily able to do standard symmetric Rijndael AES encryption. Rijndael was designed, in fact, with hardware implementation in mind. It was designed so it's very simple to implement in hardware or in a little firmware-programmed chip of some kind, which is what I presume is in the YubiKey.

But to do full asymmetric, public key-style encryption is dramatically more compute intensive. And when you really think about it, it's not exactly clear how that would work because you would - you probably need bidirectional communication. I haven't sat down and thought this all the way through. But the YubiKey, one of the things that's so cool about it is it just pretends to be a keyboard. You don't need to give it something and have it encrypt it in order to prove that it is what it is. It maintains its own internal state and merely increments that in order to, well, I guess it could, as I think about it, it could simply use its private key to encrypt that state and send that out. It would be a lot larger. It would be a lot larger. It would be the length of the modulus of the public key. So that would be a longer string to type. And then you ought to be able to use the public key in order to decrypt what it encrypted. Of course that would give you access to the encrypted contents, which might be a problem. Anyway, it would work...

Leo: It's not a simple thing.

Steve: It's not a simple thing. But more than anything you would press the button and probably have to come back tomorrow in order to get your results from it.

Leo: And that's why PGP uses a symmetric key initially, doesn't it?

Steve: Well, it's why any of these systems...

Leo: SSL uses symmetric keys.

Steve: Yeah, well, any of these systems do not encrypt the bulk payload. What they do is they generate a random number, a big random number. That's the symmetric key. Then that's the only thing they encrypt using the asymmetric encryption is they encrypt the symmetric key. That's what they send. And then using the other side of the asymmetric cipher, they decrypt with the other half of the asymmetric key that gives them the symmetric key. And then you decrypt the whole bulk payload. And now we're all confused.

Leo: What he said.

Steve: Yes.

Leo: Jez Goldstone in Manchester, U.K., has been thinking about the Phorm browser dance. Talked about it extensively last week and a couple of weeks ago. Steve, I really don't want to give these nasty guys any ideas. But your recent discussions about Phorm have centered around how they bounce a user's browser around to set a cookie for profile purposes. Since they can see the IP address of any request, the ISP can simply tell them which unique subscriber has that IP address at any given time. They can then associate the current IP address with their unique reference and know who's sending the request and what profile to use. They can then inject a cookie into the requested page between the user sending the request and it being passed on to the required destination. They could then pick it up using their advertising service on a third-party site that they're providing advertising for, in your example CNET.com. There would be no way to take anti-Phorm measures apart from using TOR or JAP or some other SSL anonymizing service. I know that this approach does not work for multiple users behind a common NAT router, but then neither does the cookie-based approach when multiple users share the same log-in account, for instance within a household. Did I miss anything?

So he's saying the ISP says match this Internet address to this unique subscriber, associate that address with a cookie reference, and then you'll know who's sending the request, what profile to use, and then inject the cookie. It's complicated.

Steve: Yes.

Leo: Does it work?

Steve: Well, it works. Bu the thing he missed, if you can put it that way, is that requires clear collusion between the third party and the ISP.

Leo: In every case the ISP is saying we're not revealing personal information.

Steve: Yes. And even so, everyone is up in arms about Phorm and NebuAd and their ilk of these third parties where they - I mean, and I have to say, I mean, the Phorm folks, I can't really say that one way or the other about the NebuAd people. But certainly the Phorm folks, because I've gone through their technology, they go to extreme measures to anonymize the data. You know, that's not satisfying people. They still don't like the idea that they're being profiled, even in that fashion. But at least, I mean, Phorm can argue, look, we really have a hands-off approach. Sure, we're sniffing and modifying the ISP's data stream. But we're doing it in a completely hands-off fashion. We're getting no account information from the ISP, and we are completely independent of IP. So I think, if somebody actually could tap into the ISP's database and say who's currently the customer on this IP, oh, boy, I mean, that...

Leo: Well, yeah.

Steve: You'd just go down in flames.

Leo: Right. So in other words, they've thought of this. They know they can do this. They're just...

Steve: They'd love to be able to do that, but they dare not.

Leo: They dare not.

Steve: And even so they're in deep doodoo.

Leo: Yeah, exactly. All right, Steverino. Are you ready, my friend?

Steve: The final two.

Leo: The final two questions. And they're good 'uns. Steve's given them special awards, awards of special merit. First, Pat Kugel in London, Ontario with the Sad and Disturbing Truth of the Week: Hi, Steve. I've been listening to the podcasts over the past several weeks and thought I'd pass along a little information from Canada. One of the two primary service providers in Canada is Bell Canada. They provide a PPPoE-based carrier service - notice this is not an ISP service; that would be Sympatico - for a large number of smaller ISPs. So they're basically the carrier, and then there's an ISP on top of it. They're being sued by a Quebec organization in addition to being audited by the CRTC - that's the Canadian FCC - due to their installation of DPI, Deep Packet Inspection servers. Their intent, to deep scan all PPPoE packets so they could target advertising and block websites they don't agree with.

Under CRTC regulations, this is illegal as the PPoE packets are considered private, and as a carrier Bell Canada does not have the legal right to open the contained packet. They're only allowed to examine the PPPoE wrapper for a destination. I've been reading and hearing that this kind of thing is happening more and more by ISPs and by carriers. If memory serves me, wasn't Comcast just given a warning by the FCC for the same thing? As a netizen I'm concerned that we're slowly losing the 'Net Neutrality war. Are we sacrificing too much in the name of profits and advertising? Or am I just being paranoid?

Steve: That is the sad and disturbing truth of the week.

Leo: They're all doing it.

Steve: I know. It is really becoming distressing. I mean, I think that Pat expressed this perfectly. And that is that a common carrier is - they have obligations under the law. They're certainly not able to spy on the data that they're passing and essentially censor it based on its content.

Leo: It'd be like listening to your phone calls and saying, oh, no, you can't call that person.

Steve: Yeah, we don't like what you said.

Leo: No, no, no, sorry.

Steve: No. And unfortunately the technology is here, this Deep Packet Inspection, basically DPI, it was nice when it just meant Dots Per Inch instead of Deep Packet Inspection. Unfortunately, I'm afraid that DPI is ending up being renamed because unfortunately, as the cost of doing this has come down, as processing power has gone up and networking has matured, it becomes virtually inexpensive for this kind - for carriers to examine, deeply examine the traffic that they're carrying. And, I mean, this is like sort of the next level of problem from the Phorm and NebuAd type companies. Here Bell Canada themselves are doing this.

Leo: It actually seems to happen a lot in Canada. In fact, Sandvine, which is the service that Comcast uses for DPI, is a Canadian company, I think from that area, I think from London, Ontario, as a matter of fact.

Steve: Yeah.

Leo: "Jake" - I put that in quotes - writing from some time in the future, wins the Creative Writing Award of the Week. Subject: A Big Thank You from the Future. Dear Steve and Leo: Since back in 2003 I've been working on a time machine - for 10 years. I was so buried in my work that I didn't pay much attention to issues like hard drive health until recently. With the current state of electronic security in 2013, I came upon Security Now!, which is really now Security Then!. And I listen to it for the nostalgia of the good old days when security was comprehensible.

One day my computer came to a halt. With all of my data and designs at risk, I immediately bought a copy of SpinRite 7, and in hours I had recovered my data. By the way, v7 makes great toast. SpinRite allowed me to complete my time machine, so I took my first trip back to 2008 just to thank you. Keep up the good work. You're our only hope, Obi-Wan Kenobi. Best regards, Jake. P.S.: Please visit us in 2013 for a demo of my perpetual motion machine. But don't count on free energy. That's a pipe dream. Very cute. Very cute.

Steve: So I got a kick out of that.

Leo: I love that.

Steve: Ever since 2003, for the last 10 years, he's been working on his time machine. And of course we do not have SpinRite 7 yet, folks. We're selling SpinRite 6.

Leo: You're going to get emails now, people saying where's SpinRite 7? I want it.

Steve: I know, what Jake talked about. Well, okay, Jake is from 2013, and he traveled five years back in time to 2008. So we just want to let you know, SpinRite 7 apparently makes great toast. I didn't have it on my list of things to add.

Leo: Now you do.

Steve: Don't have that at my update. But, you know, I'm not arguing with Jake. Apparently he knows more about SpinRite 7 than I do at this point.

Leo: Toast is good stuff.

Steve: There you go.

Leo: Oh, Steve, it's been fun. It's been a great 12 questions, as always. How can people send you questions or suggestions?

Steve: GRC.com/feedback. And thanks to you, Leo, they need add no extension to the end of that.

Leo: No .htm.

Steve: And no wwwwwwwww on the front. Just GRC.com/feedback. 435 people sent me their questions and comments and notes and things in the last two weeks. So I really appreciate those. I just have such fun plowing through them and finding 12 to share with our listeners. So by all means keep them coming.

Leo: Thank you for doing that. And by the way, if you want to go to GRC.com for other reasons, there's lots there. Wizmo...

Steve: And more coming, too.

Leo: More, something neat Steve's working on. But ShieldsUP! is there, and of course don't forget SpinRite, the world's finest disk maintenance and recovery utility. It's all at GRC.com, along with 16KB versions of the show, transcripts, show notes, it's all there. GRC.com. Steve, thanks a lot. That was a great batch of questions, some fascinating stuff. Next week what's on the agenda?

Steve: Another great 90 minutes spent with you, Leo.

Leo: Yes, sir.

Steve: Next week I will say again to our listeners who do have propeller-head beanie caps, this is going to be one of our great techie episodes. We're going to delve into, in detail - and this is, I mean, again, everybody's going to understand this. I promise you, no matter how much of a neophyte our listeners consider themselves, they're going to understand how DNS protocol works at the level required to spoof your own ISP's server if they haven't patched by that time.

Leo: Okay. Strap on your pseudorandom number generators and head into the future. We'll be back in a week.

Steve: Strap on your pseudorandom number generator? That's right.

Leo: Put that right on.

Steve: Make a few passwords, while you're at it.

Leo: Steve, we'll talk again next Thursday on Security Now!.
