SECURITY NOW!

Transcript of Episode #148

Listener Feedback Q&A #43

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-148.mp3
Quarter size (16 kbps) mp3 audio file URL: http://media.GRC.com/sn/sn-148-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 148 for June 12, 2008: Listener Feedback #43. This show and the entire TWiT broadcast network is brought to you by donations from listeners like you. Thanks.

This is Security Now! with Steve Gibson, the Wizard of Security, from his lair in beautiful Irvine, California, where it must be about 100 degrees once you get aboveground, out of the Fortress of Solitude; right?

Steve Gibson: It was drizzling this morning, actually. We've been having a really weird, like, I don't know if summer's ever going to get here. I got a piece of solicitation in the regular paper mail yesterday from some air conditioning company that had over-purchased air conditioners. That was their come-on, anyway.

Leo: We bought too many.

Steve: We thought we were going to be installing air conditioners in '08 summer, but we're not, so we're going to give you a thousand dollars off. It's like, okay, well, that's kind of believable. And they said they had 11. They had 11 too many.

Leo: Send them my way. I could use an air conditioner. It's hot up here.

Steve: Not down here.

Leo: It's got to be in the 90s, believe it or not.

Steve: Beautiful day out.

Leo: It is, it's a beautiful day except I'm inside. So let's talk a little bit about our listener questions and answers. We're going to do Episode #43 Feedback. We've got some great questions from people all over the country.

Steve: Yeah, we got a neat variety today. Some are long; some are short. It probably balances out to about what's usual. But there was a lot of good stuff in my mailbag. People went to GRC.com/feedback and sent us stuff, as I always want to encourage people to do. I read as much as I can.

Leo: Including a "Looming Threat Observation of the Week." But first, before we do any of that, let's get an update on the news, security news, around the world.

Steve: Some interesting things in security. First of all, we're recording this on the monthly Tuesday, the second Tuesday of the month.

Leo: Patch Tuesday.

Steve: Patch Tuesday, Microsoft's Patch Tuesday. So unfortunately the update details are not available. I do know because they now release a week ahead of time, they tell us sort of a broad stroke what's coming. There are three critical updates, and I'm most interested in one which involves Bluetooth because, of course, being radio and with many Bluetooth adapters on laptops, that's of a concern. So, well, not only laptops but, you know, portable Windows devices of various sorts. So there's three critical updates: Bluetooth and IE and DirectX. On the other hand, when wasn't there a critical update for IE? And even DirectX has been having lots of problems. The good news is the BSA, the MBSA, the Microsoft Baseline Security Analyzer that we talked about last week, will inform any of our listeners whether their updates are needed. All three will require rebooting your system, so it's something that you may want to plan for. By the time you hear this it'll of course be Thursday, and this'll be two days ago for you. So the information will be there. Windows Update will probably know. But specifically because there is a radio-based critical remote code execution problem somewhere. I don't know as we're recording this where. But it does sound like something that our listeners are going to want to make sure they get patched.

Leo: All right. Run your patch system if you haven't already. And I turn mine, automatic patch is on. Do you do that? I'm just curious.

Steve: No, I don't. I've got mine set to download and hold. So download and ask me. For example, I don't want SP3. SP3, Service Pack 3 for XP, has caused me two separate sets of problems. And my tech support guy, too, Greg. And by removing them in all those cases the problems went away. So I've confirmed that it was SP3 that did it to me. It's like, I'm not in a hurry to go there. Although Microsoft, every time I check they're saying, hey, you don't have SP3. It'll be good for you. It's like, uh, no, I don't think so.

Leo: I installed it on one machine, and it was actually smart enough to back out. Said oh, problem, and then backed out very nicely. And then I put it on a - I had it on a - XP SP2 on a virtual machine on my Mac, and it installed just fine, flawlessly. Then I immediately made a safe point in VMware saying might want to go back to this point. So...

Steve: Well, and it's certainly the case that not everyone is having problems. These things seem to be very anecdotal. But I have this, as I said a couple weeks ago, items in my Start menu stopped responding to the mouse. And it's like, okay, going into Windows Explorer and double-clicking on the EXE is really not very user-friendly. So I decided I don't need Service Pack 3 for the moment.

Another very important problem for Skype users is an interesting one. Apparently it's only Windows versions of Skype, although I didn't see that explicitly reported. Although I say that because the Mac version of Skype I've got is down at 2.7.something.

Leo: Yeah, yeah, it's way behind, isn't it.

Steve: So this is relative to any Windows version prior to 3.8.0.139. There has been a vulnerability discovered which allows some Skype user to send another Skype user a link which is executable, but it bypasses Skype's executable verification, which would normally warn you before you were going to click on an executable. So this would allow someone to send you a link masquerading as something else, causing you to run the program on your machine. So, I mean, again, it's as long as you're not using Skype, or if you only Skype to people you really know and trust, probably not a big problem. But it is something that has come to the security community's attention. And it's been fixed. So as long as you're at .139 and later, 3.8.0.139, you're going to be okay.

Leo: Good to know.

Steve: Now, the last thing is really interesting. Probably by now people have heard Microsoft saying not to use Windows Safari.

Leo: Yes.

Steve: Yes. And the reason for this is just freaky. Okay, so there's a problem, a known problem with Safari on Windows. Apple knows about it, and their initial reactions were that's not our problem. Okay. So what Safari does, Safari has always, both on the Mac and this behavior moved over, well, stayed with Safari when it was ported over to Windows. And that is, it will allow web pages to download programs without user involvement or confirmation. So, I mean, to me that doesn't seem very secure. But...

Leo: It doesn't execute them, it just downloads them.

Steve: That's what I'm saying. So that means you could go to a web page, and that web page could, without your user involvement, download a file to your machine. Now, that seems like a bad idea. But that's the way Safari has always been. Okay. The default directory for downloading on Windows versions of Safari is the user's desktop. So the first thing that happened was someone realized that - a security researcher called it "carpet bombing." You could go to a malicious website, a Windows Safari user could go to a malicious website that would "carpet bomb," to use his term, your desktop with files. And, you know, which is certainly annoying. Okay.

Then it turns out that a sort of a mistake in the design of Windows interacts with this. Therefore in a security standpoint, or in security jargon, we call it a "blended threat" because Safari by itself can just download the file without user involvement, but won't execute it. Unfortunately, Windows will. So what happens is that, when Internet Explorer runs, it loads a number of DLLs into itself, which it finds on the system in various places. Now, the problem is, and this is an old security problem with Windows from years ago which was known as the "DLL search order," that is, the order of subdirectories that would be searched when DLLs are loaded.

There are three DLLs: sqmapi.dll, imageres.dll - image resources - and schannel is a secure channel .dll file. Well, it turns out that if you don't have Windows updated with so-called "secure DLL search order" - and it wasn't until Service Pack 2 that this was set by default to be on, and it's not clear to me if you upgraded to Service Pack 2 whether it would turn it on. That is, remember there are many things that Microsoft left alone when you went to Service Pack 2. For example, if the firewall was off, it wouldn't turn it on. But if you installed from scratch the Windows with Service Pack 2, then it was turned on. So there were a number of things like that.

So the problem is you could use Safari to - a Windows user using Safari goes to a bad site which puts, for example, schannel.dll on your desktop. And you don't noticed that it's arrived, for example. Many people have cluttered desktops. So one more little icon it's like, oh, well, I'll get around to cleaning this up later. Then you later run Internet Explorer. Well, it turns out that the non-secure DLL search path includes the path where it's first - the first place that DLLs are searched for is the directory that the program itself is loaded from. So it would look in IE's own directory where the IE executable is. However, these things live in the system32 directory, that is, these various DLLs do. So if it's not found there, it goes through a series of directories. The second directory which is checked is the directory from which the app was launched. Well, most people launch IE from their desktop. So the second...

Leo: Oh, that's interesting. So it would look on the desktop.

Steve: Yes. And it would find the schannel.dll which was deposited there by someone using Safari.

Leo: Is that why Apple says it's not our fault, it's Windows's fault? Is that because it shouldn't be doing that?

Steve: Yeah.

Leo: It shouldn't have the desktop in the path or whatever?

Steve: It's very creepy. I mean, yes. The DLL search order problem, again, it was something that - Windows has always done it. At some point someone said hey, you know, Microsoft, this is not a good idea. And so they said, uh, you're right. But they were worried, as they always do, and we understand this, that if they change something fundamental to the system like the order in which DLLs were loaded, lord knows what side effects that might have. So...

Leo: But I have to say Apple's - that's passing the buck because this is not good behavior regardless of that.

Steve: Correct. So the workaround, Microsoft gently says we sort of recommend maybe that you not use Safari.

Leo: I think this is a little sniping, don't you think?

Steve: Well, you know, and I was curious because the various news reports were, like, in bold print.

Leo: Microsoft's known...

Steve: Yeah, Microsoft warning people not to use Safari. Okay, so I looked at the actual jargon, the actual verbiage on Microsoft's site, and it's very gentle.

Leo: It doesn't say don't use Safari.

Steve: Not at all. I mean, well, it gently says that. All you have to do if you're a Safari user on Windows is have your files go somewhere else. Create on the root, for example, SR, you know, Safari download, or put it wherever you want to, under My Documents or something. And then change the default in Safari so that that's where files are downloaded by default. And then if anybody - if you go to a malicious site it'll just go into that directory, which will never be searched for DLLs. So it is the case, this is a little bit of a - it's blended, and so is the responsibility. I don't like the idea of a browser not prompting me to verify that it wants - that I want it to download a file. I'm glad that that has been added in IE7 very clearly, where you're really prevented from that. But at the same time, you could say, okay Windows, that's just bogus that you're going to run files that I might have on my desktop.

Leo: Shouldn't do that. And Safari shouldn't be doing what it's doing. And one of the researchers said you can use this for a, quote, "carpet bomb" attack, which is not a security risk, just really, really annoying. You can go to a web page that just loads up a ton of files onto your desktop, you know, hundreds and hundreds of files on your desktop. That's just annoying. But I think that's a clear flaw.

Steve: Yeah, I agree.

Leo: They should fix that.

Steve: Yeah. I think it ought to be, well, I would like to see Safari changed so that you have to give it permission before loading a file on your computer.

Leo: You know, Safari has been a problem on the Mac side as well because it would open files that it considered non-dangerous files. And so somebody showed how this could also cause a security flaw. And they changed that behavior. I think it's just a matter of time before they fix Safari on Windows.

Steve: Yeah, well, as a consequence of Apple not having been in the crosshairs of hackers as long as Microsoft has been, there are some...

Leo: Right, they get away with stuff.

Steve: Well, yeah, they don't have the maturity of understanding...

Leo: The paranoia.

Steve: ...that anything that can be done, will be done.

Leo: Right. Some would say that's paranoia and not maturity of understanding. But it's exactly the right point of view. Now that the new iPhone is out I think that this maybe makes us understand a little bit better why Apple was so anxious to get Safari on Windows because the new iPhone has enterprise-grade kind of synching for Windows, as well. And it looks like there's features that you're going to want to use Safari for. So and I think Microsoft probably also resented a little bit Apple's pressure on Windows users to install Safari, remember with that update. You've got to update Safari even if you didn't even have it.

Steve: Yeah. Well, and there were, as I understand it, there was, like, they're trying to get you to load more of their Windows stuff ever you do anything. One last comment about the iPhone and the iPod Touch both. I wanted to mention to any of our - I don't know if I want to call them "fat-fingered listeners." But ThinkGeek just came out with a stylus that works on the iPod Touch and the iPhone.

Leo: Really. Because, now, that's a capacitive screen, so you need to use your - normally you need to use a live human finger to do it.

Steve: Exactly. For example the Palms use a film screen that is a resistive technology, so you can tap it with your fingernail or the stylus or whatever you want. The iPods are a capacitive technology, so you can't use, for example, the eraser end of a pencil. It won't work. And recognizing this, some enterprising company came up with a stylus that does work. And so it solves the problem of typing on the keyboard, which is my main gripe with those is that the keyboard's just unusable. Elegant as the result is, it's like, okay, I want my little two-thumb keyboard, please. Like the Treo and the Blackberry have.

Leo: And that's really - that was really the reason I use a Blackberry, not the iPhone, is the keyboard.

Steve: Yeah, it's impossible.

Leo: Any other tech news?

Steve: I have one interesting, well, an interesting SpinRite report. This is not a testimonial. And the reason I chose it was that he asks a question at the end which I think our listeners will find really interesting and which I remember, believe I remember feeling was really interesting when I first heard it. This is Philip Le Riche. He said - the subject is, "SpinRite Fails Again." And it's like, okay. And he says, "Hi, Steve. SpinRite failure stories seem to be the flavor of the month recently." And I'm thinking, okay, I don't know if that's such a good trend; but, you know, that is the case. We've been having some fun with the "SpinRite didn't work for me" stories. Although typically it's not SpinRite's fault.

Leo: Well, there's a good reason, and there's always a happy ending, I might point out.

Steve: And he says, "And now it's SpinRite floppy tales." He says, "Well, here's one that combines the two. I'm one of those people Leo can't understand who still uses floppies."

Leo: I can't understand that. I just don't get it.

Steve: "In fact, because I need them for data transfer between two systems for which USB memory sticks are prohibited by policy."

Leo: Oh, interesting. However, what a silly policy if they still allow floppies.

Steve: Yes. Well, actually he even anticipates you saying that at the end of his note.

Leo: Okay.

Steve: He says, "A few weeks ago I popped a floppy into my trusty old USB floppy drive and got the dreaded 'This diskette does not appear to be formatted. Would you like to format it now?' message. Another floppy gave the same result. However, only a month or two previously I'd finally caved in, after some 60 episodes of Security Now!, and bought my own copy of SpinRite."

Leo: Yay.

Steve: "So I was glad for the chance to try it out when in need. I hibernated my PC, booted into SpinRite, and set it to work on the floppy. But I was disappointed. Almost immediately SpinRite came up with a message, the precise wording of which I have since forgotten. But the sense of it was that my floppy was not just dead, but as good as fossilized."

Leo: Well, that's why I don't like floppies, because that happens a lot.

Steve: Well, SpinRite is normally effective in this case. Well, we'll find out why it didn't work here. He says, "Rarely do I miss a chance of taking something apart. So I got out my screwdrivers and opened up the floppy drive. The problem was immediately apparent. One of the heads had fallen off its carrier and was just dangling by its leads."

Leo: Okay, now I understand.

Steve: He says, "Expecting SpinRite to cope with that..."

Leo: That's a little much.

Steve: "...would indeed be a bit much." He said, "It looked relatively easy to remove the carrier, superglue the head back on, and put it back together again, provided there was some kind of detent to ensure correct head alignment. Unfortunately, there wasn't, with the result that..."

Leo: Floppy drives cost $4. I'm sorry.

Steve: "...with the result that SpinRite still reported my drive as fossilized. So I had to order a new one. I'm left with one question." And here is his question. "With CD drive speeds now at around 52x, how is it that it's taken 20 years for me to find even a double-speed floppy drive like this smart new one of mine?"

Leo: Yeah, that's a good question.

Steve: Then he says, "Thank you for SpinRite and for Security Now!. Never before I started listening had cutting the lawn been such a pleasurable experience." Then he signs off, "Best regards, Philip. P.S.: I can almost hear Leo asking, what's the point of a policy that disallows USB memory sticks, but..."

Leo: He knows me too well.

Steve: "...but permits USB floppies?"

Leo: Yes, and?

Steve: He says, "Well, at least with a floppy disk containing sensitive data, if you want to dispose of it, you can open it up and drop the floppy bit in a shredder. And not many USB sticks have a read-only reset switch so you can be quite sure no sensitive data leaks when you import from a less trusted system."

Leo: I should point out, and he even kind of raised this issue, that most CD-ROMs now are 20 times faster than floppies. You just have a read-write CD. And you can get shredders that will shred CDs, so. [Indiscernible] do that myself, personally.

Steve: Now.

Leo: Yes.

Steve: Why don't floppy drives go any faster?

Leo: Yeah, why don't they?

Steve: When IBM, who is the originator of the original big old eight-inch - was it eight or 8.5 inch? I think it was eight-inch diameter floppy...

Leo: And those were actually floppy. They really flopped.

Steve: Yeah, they were floppy - oh, I see what you mean.

Leo: They were floppy. That's where the name came from because they were flippy-floppy.

Steve: They had a soft outer shell in addition to the Mylar doughnut that was inside, correct. Get a load of this. IBM was running those at the maximum speed possible for the head to stay in contact with the medium. It turns out that you start to have aerodynamic effects if you go any faster than the original floppies did. And so because of the whole technology basis is a head in contact with the Mylar, instead of flying over - now, remember the Bernoulli Boxes had the heads flying. They were soft media, but they deliberately spun them fast enough so that the head would fly over the surface. And of course all hard drives operate on that technology. But floppies have always been a head in contact with Mylar technology, and you can't spin them any faster or the heads take off and fly. And then, again, even SpinRite won't be able to save you.

Leo: Although probably at some point some engineer, as you say, said hmm, flying heads, interesting idea. Maybe we can use that.

Steve: Well, I think what happened - oh, you mean, like, and then they went into...

Leo: Then they made hard drives, yeah. Although, now, when IBM made the eight-inch, I mean, they were doing Winchester drives probably at the same time.

Steve: I don't really remember the sequence of events, yeah.

Leo: Are you ready, Steve?

Steve: I'm ready.

Leo: It is time to ask you. This is from Dustin in Raleigh, North Carolina. Dustin offers a useful USB thumb drive reminder.

Steve: Wait, what are we on?

Leo: Oh, wait a minute, that's 11. We'll save that. Here's Aaron in Union - Steve's going, what page are you on now? I went out of order. I saw "1," and I said that must be it. Aaron in Union, New Jersey has a self-conflict resolving NAT router. Huh? Steve, I've been listening to you and Leo since Security Now! #1. I like to think I'm absorbing all of this wonderful and useful information, especially since I'm taking a collegiate summer course in computer security. But in all that time I never heard of anything quite like this. Ready?

He says: I have Verizon DSL, and in effect I have two routers, a Linksys Gizmo provided by the now-defunct SunRocket - that was a Voice Over Internet provider, a very good one, went out of business - which connects to the outside and DSL, and a Pre-N Netgear router. I wanted to take the Gizmo off my network since it no longer serves a purpose - he's not using SunRocket anymore - and it's not completely stealth, unlike my Netgear. Anyway, the Gizmo was set up to be the dialer into the DSL. He was using the Gizmo to do the PPPoE. So when I typed 192.168.1.1 to access the login for my Netgear, the router page told me that, due to some conflict with Verizon, my router is now 10.0.0.1. I thought I remembered you saying that 192.168.1.1 couldn't conflict with any ISP's IP range. I was also unaware that my router could so easily change its local IP settings, so when it does DHCP magic it now pulls from the 10-dot range. Are there any security implications? How is this possible? Is it common? Help me, Steve, help me. What's happening there?

Steve: Well, I was very impressed to hear, first of all, that the Pre-N Netgear router would automatically recognize that the upstream net range was conflicting with what it was going to be assigning to by default downstream. In other words, what happened was the first router that he's got on the Internet, that is, the Internet-facing router, is his DSL router, this Gizmo, as he calls it. It was taking whatever ISP assignment he had of IP and no doubt translating it through NAT, Network Address Translation, into the common 192.168.1.1 network. So apparently when the Netgear, which was connected downstream of the Gizmo router, when the Netgear saw that the WAN side was 192.168.1.1, it was smart enough to recognize that it could not use the same subnet, the same network range on its LAN side if it was going to properly perform Network Address Translation because you need non-overlapping networks in order for the routing in the router to understand whether the packet stays local or needs to cross into the outside world.

So anyway, I'm impressed with the Netgear. What it apparently did was when it saw that its WAN side had a private IP address - which is what 192.168.x.y are, those are private IPs that will not be ever used on the public Internet because there's no routing information for them, there's nowhere for the packets addressed to those destinations to go. So when the Netgear inside router saw that the outside was already a public IP - I'm sorry, was already a private IP, that 192.168.1.1, it chose 10.0.0.1, for example, as its own internal address. So I'm impressed with that. And really...

Leo: I think all routers do that. Here, I'll tell you what happened because I've had this happen to me many times. If you don't say to the second router bridge, if you say assign IP addresses, it's going to take it from the other pool. It's going to take it from the 10-dot pool. It's happened to me every single time. So what he's got is he's got both routers doing DHCP.

Steve: Okay.

Leo: So the Gizmo is doing DHCP, well, most of my routers, when they see that they're also being asked to do a DHCP, they say, oh, well, I guess I'd better use 10-dot. Because otherwise you will have IP address conflicts, won't you?

Steve: Well, I guess maybe I'm just older than I thought because I've run across...

Leo: You've never seen that?

Steve: No, the routers that I've encountered have to be manually reconfigured into another range if their ranges overlap. For example, you might have two routers that both want to use 192.168.0.0 to 50 or 0 to 100. And so you can set the second router to be .1.0 to 100. They don't have to be, like, move over to 10-dot. They just have to be non-overlapping ranges.

Leo: The best thing to do, though - well, actually I'm curious. I'm going to give you a follow-up question. Generally what I'll do, if I have one router doing DHCP, say this Gizmo, is like I put the second router into bridge mode. I say don't do DHCP, let guy number one do it. Then you'll never have any IP address conflicts.

Steve: And then I don't know why you would have a second router.

Leo: For WiFi.

Steve: Ah, okay, yes, absolutely, I mean, that would absolutely make sense. So, yes, you could use it, well, you could either have it bridging or convert it into a so-called "access point" so that it's not doing any NAT at all, it's just...

Leo: Or vice versa. Sometimes I have routers attached to WiFi routers as the front end. But, now, here's the question. Is double DHCP, as his system is doing, both systems are doing DHCP, is there anything wrong with that? Is that a bad idea?

Steve: No, there's nothing wrong.

Leo: Doesn't slow things down or anything like that.

Steve: No, essentially it means - when you say "double DHCP" it means that the interior router asks on its WAN interface to the exterior router, give me an IP. And so it received 192.168.1.1, for example. And then machines on the inside, on the actual LAN, they asked the inner router for an IP. And the inner router, knowing that it could not have IP ranges conflicting, it jumped over to 10-dot. Which, I mean, and it makes sense because, okay, for two reasons. First of all, we do have people who are now stacking routers, as this guy was. And most people plug them in and just set them and forget them. So having that function automatic makes sense.

There are also ISPs who are issuing public, well, they're issuing their subscribers private IPs. So you might, for example, not from your router would you receive a private IP, but from your ISP, that is, on your ISP's network you would have a private IP because your ISP is running their own NAT router. And so they're NAT'ing all of their subscribers, moving them to private IPs because they may not have enough public IPs in order to give all of their customers their own individual public IP. And in that case that would explain why routers now are smart enough. Because if they see that they've received a private IP that's conflicting with the range they were going to assign, they'll just jump over to a different range.

Leo: So the problem would be if you had this Gizmo both attached to another router and attached to another computer. And then it might assign the second computer something in the 192 range, and then the second router might assign a different computer, not knowing that there's a computer in that range of the same number. And then you'd get a conflict. So that's sensible behavior for the router to do this automatically. Of course, if you put a third router on it, then you'd really be out of luck. That's why I just say put it in bridge mode.

Steve: Actually a third router would work.

Leo: Well, you'd have to tell it, okay, don't use 192.

Steve: Well, no. It would, if you have a...

Leo: Because we're out of private ranges, though, aren't we?

Steve: Yeah, but you can ping-pong back and forth. So if you had a third router, that could be back to 192. It would translate it at 10, and back to 192. I mean, it'd be confusing, but it would all work.

Leo: And overhead is not a concern of something like that?

Steve: No, these are all very fast. So, but he's certainly right. I would get rid of the Gizmo and tell the interior router, if it's able to do PPPoE and connect you via DSL, then you'd get - apparently it's just a nicer router.

Leo: Here's another reason people often have dual routers on there. A lot of times the cable company or the DSL company gives you a modem that has a built-in router. And you, for whatever reason, may not want to use that router. That's, I think, our setup here. We're using an 802.11n router that's attached to Comcast's modem/router. And in that case, I think in that case - I should check. I think we're bridging. But bridging just means don't do the NAT, just pass through. But it would be okay for us to have it be doing routing as long as we don't have a setup conflict. And as you say, you can often in the router say, no, still use 192.168, just start at 200, 1.200, and avoid conflicts that way.

Steve: Or just 192.168.2.1. Or 0.1, 1.1, 2.1. Because normally these routers will only NAT within the lower byte. So, for example, it looks like 0 to 50 or 0 to 100. And so as long as you change even the second digit, then you still have something that's clearly recognizable as a private IP.

Leo: And to answer his question, he was right, 192.168 and 10.0 both are private, can't conflict with anything outside of your LAN. And there's no security implication to what it did. In fact, it's a good thing.

Steve: Yes, it's secure, good.

Leo: Brian in Raleigh, North Carolina wonders, how can all possible bit combinations, every possible bit combination, be reduced to a 32-digit MD5 hex hash? Steve, Steve, how can that hash uniquely identify any of an infinite number of bit combinations? 16^32 equals 10^38 possible combinations that a 32-hex-digit MD5 hash can represent. But a 700MB file, as your average CD image, has 2 - oh, do I have to read this? 2 to the - he didn't even put in commas. 2^587 - oh, let's see. 58 billion, 720 million, 256 - anyway, there's a lot. It's a high number possible combinations. Since the latter is much greater than the former, how are the collisions avoided? So what he's talking about is how these MD5 hashes, these 32-digit hashes are used to represent the contents of a file uniquely.

Steve: Right, yeah, exactly. He's seeing, for example, that somebody will, maybe a Linux or a UNIX distribution site, will post ISOs, that is, ISO images of CDs. And they'll say that the MD5 hash is blankety-blank. And so there's an MD5 hash, as he says, is 32 hex characters. The idea being that you download the file. Then you perform your own local MD5 hash. And you should always get the same result as they have posted on their website, which is the result they got when they used the original file, the advantage being that you're able to verify that nothing has changed in the file. So he says, okay, if you've got 700MB of data, which the hash somehow reduces to this just little 32 characters of hex, how is it possible that all of those possible bit combinations in a 700MB file don't have any collisions? How are collisions avoided? Well, he used the term "avoided" as opposed to "prevented." So we need to get into semantics a little bit. Brian is absolutely right that there is no way to prevent collisions, that is to say, many, many, many different combinations of bits in that 700MB file will all result in exactly the same hash. So it is not an absolute perfect fingerprint for the file. The only way you can get that is doing a byte-for-byte comparison of the original file.

Leo: However, it's a long number.

Steve: Yeah, the hash has been - the hashing algorithm, the thing that makes it cryptographically secure, is that it is computationally infeasible, as the crypto guys put it, to deliberately...

Leo: Ah, there's the key.

Steve: Yes, to deliberately create a fraudulent 700MB file which will hash down to exactly the same thing. So the risk would be that somebody could deliberately make - some hacker could create a malicious ISO image and stick it on the website, assuming they didn't also have the ability to change the hash, and I guess they probably would. Or, you know, or give you an ISO image and say, hey, this is a valid image, trust me, you can do the MD5 hash and check it against the publicly posted MD5 hash, and it'll be the same. The point is that it is not feasible to design a file which, when hashed, has any arbitrary given result. And so that's what's being protected. That's what's being prevented. And given that we've got 10^38 possible combinations, that is, that many different possible hashes, you know, that's 10 with 38 zeroes after it. The point is that it's statistically unlikely that any non-malicious change would go unseen. That is, there would only be one chance in 10^38, essentially, that a mistaken change during download would change the bits of the ISO such that it still gave you the same resultant hash. It's possible, but one in 10^38 makes it incredibly unlikely that that's going to happen. So it both prevents with a high degree of reliability a random change from going undetected, and specifically it protects from anyone malicious designing a file that's going to be the same length, but changed in a way that gives the same hash. You just can't do it.

Leo: It's just impracticable.

Steve: Yup. Computationally infeasible.

Leo: I like that. That's a good phrase. I'm going to work that into conversation sometime. I don't know how, exactly. Sande Nissen in Northfield, Minnesota worries about the PayPal browser plug-in. That's the one we've been recommending that will generate for you one-time-use credit card numbers, among other things. PayPal has a new web browser plug-in that offers some useful services. Over the years I've been pleased with PayPal's privacy and security and their dispute resolution. But a browser plug-in? What? Given how buggy browsers are, is there any way this new tool could really be secure? I'd love you to take a look, Steve. Sande.

Steve: Well, I took mine out.

Leo: Oh, you're kidding.

Steve: No. Not for any security reason. I was just - it was just dunning me constantly. Any time I went to a...

Leo: I don't like toolbar add-ons. I never install those.

Steve: Well, it's worse than that. This thing, I mean, it's just a little "P" for PayPal, a little icon. Except that any time you go to a page that has forms, even when you're not buying stuff, if it happens to see, like, a field with "Name," this thing descends down from above in this little, you know, would you like me to fill this in? No, go away. I mean, I tried to disable it and turn it off and reconfigure it. It doesn't seem to obey its configuration settings, if I even understood the way they work because - and then you have to go back over to PayPal's site and say no, I don't want this dumb thing descending from above every time I visit a page that has a form where it says, oh, we can fill in your shipping address. We can fill in your billing address. Just leave me alone.

So I finally said, okay, I've had the plug-in experience, I don't need it anymore. If I need to get a one-time credit card from PayPal, I'll just go log into PayPal and get it there. So I sort of agree. On the other hand we've got password plug-ins, and who knows if someone will come in with an exploit. It is the case that you still have to authenticate yourself every single time you use this. And if you've got the football, which for $5 I don't know why anybody wouldn't have one of the VeriSign/PayPal/eBay authentication footballs or credit cards, you can require that that be used every time to really prevent any sort of shenanigans. So I sort of agree. For me it wasn't a matter of security, it was this thing was just too incessant. And I finally said I'm always trying to get rid of this thing rather than wishing that it were present.

Leo: It would be nice to have the auto-generate of the credit card numbers. But I just do that on the PayPal site. The one thing I've found, Amazon keeps saying your credit card - I made a multiuse one. And Amazon doesn't like it for some reason.

Steve: Oh, interesting.

Leo: Because you have two choices, as you pointed out. You have two choices. You can make a one-time only, which is really the most secure. It can only be used for one purchase, and then it's no good. But if there's sites like Amazon where you buy from a lot, you can make it specific to the site. But it doesn't expire, it just continues working, but except it doesn't seem to.

Steve: I did have a nice experience. I remember mentioning on the podcast that I used the PayPal one-time card on a site where they were really pushing for, like, a subscription sort of deal, and there was no opportunity to turn it off. I didn't like the idea that they were going to have my credit card information because I just didn't like the way they were behaving. And I noticed that the card, whenever you get it, it shows an expiration date of the following month. And so even your one-time-use card, here we are in June, so if you were to get a card now, it would say that it expires in July of 2008. So sure enough, I got email like a week later saying...

Leo: Your card's about to expire.

Steve: Yes, we wanted to notify you, so you don't miss any of our valuable services, that the card we have on file is about to expire. Please come back and update your credit card information. I was thinking, I don't think so.

Leo: Yeah, I mean, I'll just keep using the one-time thing. The one-time thing is great, and it is the more secure because it could only be used for one purchase. Let's see. [Curtis] Wyatt, Phoenix, AZ, got the job. He got the job: Steve and Leo, I recently graduated from NMSU with a Bachelor of Science in Computer Science. I went on some interviews with a government contracting company. In the course of the interview it was explained to me the particular job that I was applying for focused on security, and specifically encryption. I was asked a few questions, and thanks to your podcast I was able to explain the difference between asymmetric and symmetric encryption. He didn't learn it in school, but he learned it on our show. The interviewer was apparently impressed enough to give me a job. Well, today was my first day. And after recognizing what TPM is in laptops, my supervisor was impressed, as well. Thanks, Leo and Steve, from a longtime listener. That's nice.

Steve: I just thought that was a neat little note, yeah.

Leo: That's really nice, yeah. I'm sure that they must have covered that. You probably took the day off in computer science class.

Steve: A little refresher always helps.

Leo: I think one of the things that you do so well is, because we're focusing on just one topic often, you're able to really explain it thoroughly in a way that everybody can understand, and it sticks with you, as opposed to when you're studying this stuff in school it just goes and goes and goes and there's a lot to remember and stuff. So I'm sure they covered it. But he remembered your discussion.

Al in Lowell, Mass. was asked an interesting question: Hi, Steve, I love the show, he says. This isn't a question about a problem I'm having or anything like that. I consider myself very well versed in networking and the Internet in general. But a friend asked me a question I had never really thought about, and I couldn't come up with an answer. So I present it to you. Do you think the Internet and all of its protocols, like IP, TCP and so forth that we know and love, will be phased out at some point in favor of more secure protocols? One of your earlier episodes was on the social implications of Internet anonymity and the bad guys having access to the same encrypted communications the good guys have. Do you think this will ever change? Is there a way to protect digital rights and fight piracy without infringing on people's privacy? In our current model the answer is no. But may this ever change? As long as the government doesn't seize all personal computers, we'll always have the ability to maintain our current Internet model. So a new one would have a hard time replacing the old one. It's an interesting question. If you have any opinions on the matter, I'd love to hear it. So would I. What do you think?

Steve: Yeah. Well, we're seeing an evolution, certainly. The whole IPv6 standard which exists now and is sort of trying to move to the forefront, although it's not being adopted with any great speed, it for example does incorporate security, like encrypted communication security, in the base protocol, which the original TCP/IP, the original Internet protocols did not have. Things like IPSec and L2TP and SSL running over TCP, the things we've talked about in this podcast, those are all sort of optional afterthought add-ons that were sort of grafted on afterwards. And the main protocols we use, like HTTPS, like POP and SMTP and IMAP, for example, these are nonencrypted protocols by default. There are people pushing for encrypted versions of those existing protocols to be used. And also, as I said, IPv6 incorporates really good end-to-end encrypted security in the base protocol. So if you have IPv6, although you're able to have nonencrypted communications, part of the support for the spec means that you will also have good, strong encryption available. So it certainly is the case that over time we're moving in that direction, although because as Al mentioned, you know, the protocols we have today are functioning, and they do work, there isn't...

Leo: They don't work well, but they....

Steve: Yeah, they work well enough.

Leo: Actually they do work well. I mean, in a way it's a remarkable system. You know, there is an initiative going on at Stanford, they call it the "Clean Slate Internet Initiative." And the idea is, if you were to redesign the Internet, if knowing what you knew now you were to set out to design the Internet, what would you do differently? And it does address all of these issues. You know, in fact, if you go to the website, which is cleanslate.stanford.edu, they've got articles on all this, including "The Future of TCP: Train Wreck or Evolution?" That's one of the articles. And so, I mean, it's a really interesting question. Our listener raises the most important question. Given, you know, let's say you did come up with a better - it wouldn't be too hard to come up with a better solution. How do you implement?

Steve: Well, and for example we've talked about all of the various extensions which have been made to TCP, for example, over time as the world changed. I mean, the original protocols have turned out to be surprisingly robust and able to have extensions added to them in a forward-looking, backward-compatible way. So, I mean, I'm impressed that this system is running as well as it is. But certainly it's the case that we've learned an incredible amount. And if there was only a way to turn back the clock, then...

Leo: Well, I guess what you would do, I mean, if you - you see it sometimes when they build a new bridge. You keep the old bridge. You build the new bridge. And you decommission the old bridge after the new bridge is completed. You'd have to do something like that. You'd have to design, and it would take decades, design and build a new Internet structure that then you slowly replace the old Internet structure with. Or maybe not so slowly. The problem is, it can't be too big of a change. Look at how hard it's been to do, as you say, IPv6. You put it in the routers. The ISP can support it. You can support it maybe at home. But who's going to flip that switch?

Steve: Yeah, I think it'll happen in a sort of an incremental...

Leo: It's got to be evolutionary.

Steve: Yeah. For example, many people are now using secure versions of POP and SMTP and IMAP. They're deciding they want that end-to-end security for, for example, email, that is notoriously insecure. I mean, exactly the example you were giving earlier about being on the cruise and having your own email conversation intercepted, by default that sort of stuff could now easily be encrypted. And it's just it's not because of inertia.

Leo: Well, and it's also, you know, it's the same problem Microsoft has to face. Do you start from scratch and go through all the pain that involves? Or do you try to overlay on top of existing protocols more security? And that's what we've been doing up to now. But at some point it just becomes a bag hung on a bag hung on a wire on a - it's a kludge. And at some point you really would like to start over. Did you ever - programmers do this sometimes. Did you ever start over on SpinRite and say let's just do it all over again? Or do you always build on the existing?

Steve: Oh, no. SpinRite's probably not a good example. Although when I went from v2 to v3 I completely rebuilt the program. So I guess that's a good example. What I wanted to do with v3 just would not fit at all within the structure that I had. And so I just started from scratch. I built a whole new UI engine underneath it and essentially rewrote it from scratch and added a whole bunch of really cool new technology. So, yeah, from time to time. You know, and I think the Perfect Paper Passwords is another example. I think we're on v3 now. And it kept getting more sophisticated, and I scrapped it each time and just started from scratch again.

Leo: Programmers often do that. Or anybody does that. You'll start over. We've done that with a show, where we lost a show, we did it the second time, and it was better the second time because we started from scratch, knowing what we - learning what we learned.

Steve: Exactly.

Leo: Udo Penther in St. Thomas, U.S. Virgin Islands - what a beautiful place. I would love to be in St. Thomas right now. He worries about insecure bank logins: Hi, Steve. I've been a long-time listener of your show, thoroughly enjoy it. Did I skip? I did. I skipped. I'm sorry. You knew this. You didn't stop me. I'll go back. Udo, we'll come to you - actually let's finish Udo, then I'll go back to six, how about that?

Steve: Or you could just have Dane edit this.

Leo: No, Dane's busy as it is. I don't want to add any more work. We'll do Udo because there's no compelling reason for me to go six, seven as far as you're concerned?

Steve: No, no.

Leo: He says: I've been a long-time listener of your show and thoroughly enjoy it. Today I do have a question. At least two of our local banks' websites provide a regular HTTP login web page rather than an HTTPS page to enter one's password and userID. After one provides this rather critical info, the next page it changes to HTTPS, secure HTTP. So is that sufficient? I've tried to access the login pages through HTTPS, but the system does not accept them. I'm rather hesitant to use this setup. I'd hate to join our local power company and phone company in bankruptcy. And if it needs to be HTTPS, is there some professional reference I can place on the banks' managements' desks in order to bring their IT departments into the 21st century? This is a question we get a lot.

Steve: Yes. And it's been - we've talked about this before.

Leo: I've had this question.

Steve: And because it's important, and it does come up a lot, I wanted to just sort of revisit it again. The way the technology works is funky. It's an example of the fact that - it's a consequence of the fact that web pages and the web system was originally a read-only medium, that is, you would surf around, and you would read stuff, but there was really no notion of information going in the other direction. So that was grafted onto the specification in a strange way. When you're submitting a form to a web server, you're submitting form data, you actually do it in the form of a query because queries is the only thing HTTP, that original protocol, understood. And so you ask a question that includes the data you're submitting.

The confusion here is that because this site was not well-designed by the web designers, the form is not on a secure page. But the fact that he mentions that once you enter the critical information the next page changes to HTTPS, what that means is that the query itself that is being sent containing your data is secure because the page you get in response to that query is a secure page. So what your browser does, when you actually click the button to send this back, it establishes a secure connection. Then in that secure connection goes the confidential critical information to the server, which then responds to the query as such with the result page saying, oh, yay, you successfully logged in, welcome to the bank. So anyway...

Leo: So normally it's secure.

Steve: I don't know how to summarize it that way. I would say that - oh, yes, I see what you mean. What you mean is that it would be very unlikely for the bank not to secure your login information.

Leo: But it's possible.

Steve: It is absolutely possible. So IE7, for example, changed the way they handle links and buttons, such that if you float your cursor over the button, down at the bottom in the little tray it will show you the URL that button is going to take you to. And Firefox v2 and v3 both do the same. And I know that Internet Explorer has an option to warn you if any form data is being submitted over a nonsecure connection. So that's something else you can do.

Leo: You can turn that off, unfortunately.

Steve: Yes, you are able to turn that off. Again, this requires some awareness on the part of the user. It would certainly be possible for someone to design a web page that accepted form data that was not secure. So, I mean, again, this is a - it's a fault of the designers of these banks' websites that they didn't put the form request data on a secure page. Although it's worth mentioning that even if they did, the button to submit it could be insecure. So your data, even though you were filling the form in on a secure page, it's not the page where you're filling the form data in whose security matters. It's the URL of the query which is sent to the server when you click the link. I mean, it's very confusing. And as I said, it's a kludge.

Leo: So most banks do it this way, although I notice Bank of America now has changed that. Amazon does it right. So when you go to the login page, it's HTTPS.

Steve: Yes, because it gives you a warm, fuzzy feeling. It's how I designed my eCommerce system for purchasing SpinRite, as well, is I put you on a secure page. You can check, you know, our security certificate. You know then that we've got good, full, 128-bit, industrial-grade security. And then you fill in the form, and you move through the eCommerce process.

Leo: But so you said it only gives you a good feeling because it could still be insecure.

Steve: Yes.

Leo: Because the form button could be insecure.

Steve: That's a very good point. The page you're filling in could be secure, but the form button submission could be insecure, although that would be nutso for anyone to design it that way.

Leo: It would be hard to do that by accident?

Steve: Well, it could be done by accident, yeah.

Leo: So I guess that's the real point. So you should really hover your mouse over the button.

Steve: Yes, and see where you're going when you click that, you know, see where you're going to go, where the information you submitted is going to go. God help you if it goes to DoubleClick and bounces over to - if it goes to DoubleClick on the way to PayPal, it's like, oh, goodness.

Leo: Well, now I really want to see - I'm going to go to Amazon. See, IE does that, but I don't know if any other browser does that.

Steve: Firefox does.

Leo: It does?

Steve: Yeah, I'm looking at my cursor hovering over Firefox 2, and I'm seeing the URL in the bottom of the Firefox window. And somebody mentioned when we talked about this before that there's a plug-in that you can use that will pop up the - where if you hover over a button, it'll show you like right there, in a tool tip for the button, what the URL is.

Leo: Amazon puts you on an HTTPS page, and their button says "Sign in using our secure server," which is probably a good idea. But again, you know, unless you hover your mouse over there and you see that HTTPS, you don't know. There's no, you know, they could be saying that.

Steve: Yup.

Leo: This is exactly what we were talking about with the last question is that the web isn't designed kind of inherently to be secure, and there's lots of little loopholes like this. And I don't think - I wouldn't expect any end user to know about this stuff. Just, you know...

Steve: No. And it is the uninformed user, the people who are not listening to this podcast, that fall into these traps all the time.

Leo: But that's everybody. It's a small percentage that know this stuff. Now we'll go back to Aaron Feickert in Fargo, North Dakota. He wants longer passwords, darn it: I'm an avid listener of Security Now!, and I have a pet peeve to share with you. It's online services that try to limit the security of my account passwords. Oh, I'm with him on this one. Gone are the days when an eight-character password is sufficient to protect my personal information. While most sites are getting better, I'm a paranoid person when it comes to password length. I started signing up for online services, intending to use my preferred password length of 25 characters, including spaces - that is pretty long - and was rejected. Heck, even my bank limits me to 16 characters, no spaces. Mine also. With modern one-way hashing what it is, can you think of any plausible reason why I should be limited in such a significant way? If services widened their limit and at least allowed spaces, I could come up with a multiword passphrase that would be easy to remember but nearly impossible to break. Why do they do this?

Steve: Well, yes. This is a great point. And we were talking about hashes in responding to an earlier question about, remember, about the MD5 hash on a CD. The nice thing about a hash is it can take in data of any length, and it always returns a fixed-size result. That is, it's always 16, for example, in the case of MD5, it's those 32 hex characters. Even if you give it in, you know, like a three-character password, each character goes into the cryptographic algorithm, and every character sort of changes it that you add, changes the hash's output under the influence of that character and a whole bunch of internal state that the hash algorithm is maintaining. So what this means is that servers could be designed so that they, I mean, MD5 is fine, although it's regarded now as maybe not the best hash choice to use in the future. But what must be happening...

Leo: Why not?

Steve: Well, just because it's, you know, longer hashes are better, and there have been some problems found with it. But for this sort of application it's fine. The only reason I can imagine that a server would say we're limiting you to this size is they're actually storing, they've set aside in your record, in your logon record at the server, they've set aside X number of bytes. Well, the reason that's troubling is that says they're storing the password and not the hash. And the reason that's troublesome is that means that if someone got a hold of their database they would end up with all of their customers' names and passwords. The beautiful side of storing instead, storing the hash, is okay, it's fixed length, so it's got the advantage of them being able to have a fixed amount of data set aside just like a fixed length of password does. But by not storing the user's password, instead of only storing the hash, it's not something that is reversible. That is to say, if somebody, I mean, you don't want anyone to get a hold of your login database. That's a bad thing. But if they did, and if they weren't able to modify it, they would still never be able to figure out what the password was, even though they had the hash. And that's one of the cool things about hashing and storing. So Aaron is absolutely right. There's no good reason that anyone is limiting the length of a password.

Leo: Is there anything in Windows Server, whatever, that makes spaces problematic?

Steve: No, no, there isn't. It's just some algorithm just decided they don't want to allow certain wild characters, spaces, you know, maybe underscores or hyphens or who knows what. And a perfect example of a modern-day hash-based password is the WPA key. The WPA can be pretty much anything you want, the longer the better. And it always hashes it down into a fixed-size result. So there's an example. And hopefully this is the kind of best security practice that other designers will adopt in the future.

Leo: Yeah, it's a good idea. In fact that's, you know, UNIX is always - well, not always, I shouldn't say that. But UNIX, any secure version of UNIX uses MD5 hashes for its password tables for that very reason. And that eliminates any issue with weird characters or whatever, just hash it. It's always the same length. To be honest, it puzzles me. All right. So you're right. They should be doing that. Aaron. Go out and spread the word.

Steve: Complain.

Leo: My bank does it, too. I hate it. In fact, you know what really gripes me is that banks really encourage you to use four-character PINs for ATMs and stuff like that, which is really bad.

Steve: Yeah, I mean, certainly there it's the problem of they want you to memorize it. They don't want you to write it down. And they don't want people to say I can't remember what my 12-character PIN is. So they make them short, which makes them insecure, but it's a classic example of security versus convenience tradeoff.

Leo: Yeah. Paul Thomas in York, the U.K., York, England, wonders about deep packet inspection, traffic shaping, and security. Steve and Leo, great show. My son Gareth thinks I'm a geek as your podcast - that's all right, my son Hank thinks I'm a geek - as your podcast is generally playing in the car when I pick him up from school. Hi, Gareth. He always asks me, do you understand what they're talking about? No, I reply, but they do, and I will shortly. I like it. To my question, which I apologize for if it's been asked and discussed before. I've just changed Internet service providers, and I've come up against traffic shaping. My previous ISP didn't do it. Would it be possible for you to explain how this technology works and whether there are any security issues regarding its use? It does appear to be the way ISPs are going. I read today that Comcast in the states has been hit with three new class-action lawsuits due to the company's traffic-shaping practices. Thanks, Paul and Gareth.

Steve: Well, that's a really good question.

Leo: It's a hot topic right now.

Steve: It is. And essentially the idea is to traffic shape or not. If an ISP did not do any kind of traffic shaping, as has traditionally and historically been the case, they're selling you basically, or renting to you, one of their IPs which you have full access to. They're just saying you pay us so much a month, we're going to give you so much bandwidth on this connection, and we're going to give you an IP that allows you to transact traffic in any way that you want. So things began to change a few years ago. Actually, unfortunately, Microsoft was the original driving force in this change because Windows was having so many constant problems with security that ISPs began protecting us from Windows by blocking specific ports. So, for example, I know that I'm using Comcast here - I'm sorry, Community Cablevision, Cablevision. And if I have a non-firewalled machine, and I use ShieldsUP!, for example, to do a port scan of that, I'll see many ports closed but some stealthed. For example, 135 through 139 is a range that'll be stealthed. That's not me doing that. That's my cable provider that has just blocked off those ports because that's the old Windows filesharing port range that was such a problem.

Leo: Good. That's an appropriate thing to do.

Steve: Well, you might say yes, you might say no. I mean, again, that's an ISP filtering...

Leo: It's Big Brother.

Steve: ...my traffic. What if I - okay. What that means is that I could not deliberately expose my Windows filesharing, even if I had adequate security, if I used a secure username and password, if I trusted Microsoft not to have any vulnerabilities that weren't known. Say that I wanted to make some drives available that I could have access to through Windows filesharing. Well, I cannot do it if my ISP is going to deliberately filter some port ranges that they've decided unilaterally are not good for me. So it sort of began there. And of course it's escalated ever since. Traffic shaping means that the ISP actually looks at, okay, that the ISP has a policy, hopefully it's a public policy. The problem is this started happening in secret. And in fact it's not the only thing that's been happening in secret. Next week we're going to finally talk about the form system, the so-called "Phorm Webwise" technology which has really got people upset because ISPs that have adopted this are changing the pages people download from foreign servers. That's next week's topic.

Leo: Oh, interesting.

Steve: But what's happening here is that the ISP is looking at the traffic and making a proactive, unilateral decision about whether they want you to be doing what you are doing or not. So essentially, for example, in the case of using BitTorrent and peer-to-peer clients, what the ISP is doing is they're seeing that your behavior is outside of hopefully their publicly stated policy, and they're injecting packets of their own into your connection in order to alter the behavior of your computer and/or remote computers. And they're just doing it because they've decided that they want to.

Leo: This is what Comcast is getting in trouble for because they're deliberately disconnecting BitTorrent peers saying, oh, we didn't want that after all, on your behalf.

Steve: Yes.

Leo: Oh, we weren't really looking for - and then of course all of a sudden your BitTorrent download stops working. Regardless of whether it's legitimate or not they just decided - they're using something called Sandvine.

Steve: Yes. And really the problem is that people felt that they were buying an unfiltered...

Leo: Ah, there's a mistake.

Steve: ...unshaped connection. Now, yes, so either the ISP is going to have to be very clear that they reserve the right to play cop, essentially, on the connection, and filter whatever they choose to, or maybe ISPs will offer a premium connection which is unshaped, and then you'll be able to get a cheaper one that is shaped. It's not clear how this is going to evolve. But it is certainly very controversial.

Leo: I'm of mixed mind. I mean, on the one hand I want to get what I pay for in terms of bandwidth. And what the ISPs say is, well, if we don't packet shape, if don't do deep packet inspection, if we don't run Sandvine, you're not because there are going to be some hogs out there who are going to use up all the bandwidth and make it not available to you. So on the one hand I think that maybe that is a legitimate business decision. But I think they do need to be clear that they're doing it. And as you say, maybe they would offer a different tier of service. But then you'd have that problem big-time because these bandwidth hogs would be up on the higher tier where you're paying a lot of money.

Steve: Well, and remember that we did a whole episode on bandwidth congestion, on this notion of how does the Internet and how do our protocols and how do routers handle congestion. What we really need is a more mature implementation of existing technology so that, for example, people who just want to do email or browse the web would have priority over their ISP's routers and bandwidth; whereas somebody who wanted to be doing BitTorrent, huge mega gigabytes of downloads, they'd still be able to do that, but their so-called "class of service" would be such that they get to use any available bandwidth at a lower priority than the people who are just doing email and surfing the web. And of course then you have the advantage that web surfing is all snappy, and the ISP actually gets more value because they're going to have, you know, all their customers are going to be happy. They're not going to be doing traffic shaping. They're going to be doing traffic prioritizing.

Leo: But that's what gets back to the question that we had earlier, which is the Internet infrastructure. Doesn't it need to be rewritten to really do this efficiently and effectively?

Steve: Yes. These kinds of problems are not things that originally had good answers. And they weren't recognized as things that were going to be problems.

Leo: Question 9, Travis in Indianapolis, he's out driving around, looking for his iPod Touch. Where did it go? Well, we'll find out.

Steve: Oh, this is a good one.

Leo: Steve, I had contemplated this concept a few weeks ago while listening to the podcast, also reflecting upon full-drive encryption and so forth. Then just a couple of days ago I've been able to test this theory myself. I hope it gets me results. Someone broke into my car in my driveway and many others, too, I guess, and took my GPS device - they love those, by the way, they're like candy to car thieves - as well as my iPod Touch. I was devastated about the loss of the Touch, not so much about the GPS, because that's how I listen to your podcast and others two hours a day in my car to and from work. Man, I know how that feels. If you don't have that, and you've got that commute, boy, it's like, I've got to listen to the radio?

When I filed the police report right away that morning I told the officer that I'd hacked my iPod Touch, and if the thief tried to check my mail it would show up in the logs of the mail server I run myself. When I got to work, indeed, I found in the logs that while I was driving to work they were accessing my email. Bingo. I got the IP address of where those curious thieves were reading my email. This guy's smart. I quickly changed the passwords and for about another hour it continued to try to check email but was now getting failures. Oh, they continued to try to check the email, but they were getting failures. Of course my hacker instinct was in full force. Unfortunately they don't have any ports or remote administration open where they are.

[Talking simultaneously]

Steve: And this is where I told you that I removed his last name from his posting because...

Leo: He attacked their address. I don't blame him, but...

Steve: No, I mean, and sad to say, it's illegal for him to try to penetrate their system, even though they're creepy thieves.

Leo: It's called vigilante justice. I did verify via ARIN [American Registry for Internet Numbers] that it was a Comcast address who services my area - he did a WhoIs - and was based in the area, too. I've passed this information along to the officer. He's submitted it for a subpoena to find out who the present owners of that Comcast IP address are. I hope he can use this to help get back my stuff and everyone else ripped off. You know, if it's a big enough case, that's pretty good evidence they've got there.

Steve: Yeah.

Leo: I'm also doing some war driving looking for the MAC address - oh, because he has the MAC address of his iPod Touch. He's looking for the MAC address as it's certainly broadcasting unless they've turned it off. Seeing as how they were...

Steve: Can you see this poor guy driving around looking, trying to find...

Leo: iPod.

Steve: ...where's, you know, checking all the MAC addresses that he picks up in the air. Ohmygod, there's my MAC address. I mean, he still doesn't - it doesn't tell him where it is exactly.

Leo: It's just in the neighborhood now. Couple hundred feet. Seeing as how they were dumb enough to check my email, he figures it's probably still on. But here's the question. Had I made it so that the thief couldn't access my computer to communicate back to the world at all, I wouldn't have been able to get the IP address where it was located. Now, this is the only way I can get my stolen electronics back. So in this case if it were a laptop or something, then the whole drive encryption, BIOS passwords, any other of those effective security measures that would limit their access, I'd have nothing to go on. So I guess in some cases letting them have access to your device can actually increase your security, or at least your chances of getting that device back. I love the podcast. Hope this gives you a different view on security that perhaps you hadn't thought about. Wow, that's - well, there are LoJack devices for laptops that do this exact thing.

Steve: Yes, there are a number of companies that sell a tracking technology. They install themselves down essentially sort of a preboot technology that in the old days they used the phone, the modem in the machine in order to dial out and identify you. And now of course they use the Internet and IP addresses in order to do that. So there are - there is that kind of technology. But it was sort of an - I loved his whole story about seeing whether the iPod Touch was going to be having email checked, and if so, then grabbing the IP and giving it back to law enforcement so they could track these guys down. Sounds like it's probably going to be successful. Because I do know from my own experience that if law enforcement gives an ISP a subpoena saying we need to know who has this IP address, I mean, the ISP is absolutely happy to do so. They only want the protection of being compelled to do that by receiving a subpoena from the court so that they can't be brought I trouble, gotten in trouble from their customer who says, hey, you know, I'm not happy that you told the police where to find me.

Leo: There's a product for the Macintosh that's very clever. It looks for - it knows what the IP addresses are for Apple stores. So hoping that the thief goes into an Apple store, it will immediately log into the open Internet there and take a picture of the thief using the camera built into the laptop and send it back to you and say he's in the Apple store right now. Which is I think a great idea. They're having some cases of laptops being recovered with these kinds of technologies. There actually is a LoJack for laptops, kind of same idea. I like it. But it doesn't enhance your security, only enhances your chances of getting the computer back.

Don in Burbank, California has a book recommendation: Hi, Steve. Just finished "The Code Book." Oh, yeah, that great Simon Singh book. Oh, I loved that. You guys mentioned it and talked about it before. This is a must-read for Security Now! listeners. It starts off a bit slow, but it builds up very carefully, and it really makes - just like our show. And it really makes you sure you know cryptography when you're done. What I got out of it was, oh, so that's what Steve's talking about. Thanks for the great podcast. I always learn something new. "The Code Book," Simon Singh. You have nothing to say, I take it.

Steve: Nope. I just wanted to pass on his recommendation because I...

[Talking simultaneously]

Steve: ...it's a great book.

Leo: Yeah. And then I have "Codebreakers," too, which we mentioned before, by David Kahn, which is the thick book that actually was a little out of date because I think it ended in the '60s. But it covers Enigma and everything. And then he updated that just a few years ago. Maybe up to the '90s, anyway. You know where I got the bibliography for those two books was from "Cryptonomicon."

Steve: Ah, right.

Leo: Neal Stephenson's wonderful novel about encryption, which is great to read.

Steve: Long.

Leo: Long. But it's my favorite book of his.

Steve: I really enjoyed it, yeah.

Leo: And in the back, you know, he did a lot of research on crypto. In fact, he comes up with his own crypto scheme using a deck of cards in there. Which I think Bruce Schneier said, yeah, that's pretty good. I think Bruce helped him do it. Anyway, he has got a long bibliography in the back of that, that includes those two books, among others.

Dustin in Raleigh, North Carolina. Finally we get to Question 11. He says - he offers a useful USB thumb drive reminder: Steve, I just had a Security Now! moment. I'm attending a work-provided course this week, it's a PowerPoint-driven course, so the instructor wanted to share his slides with the students. On the second day of the course, the instructor passed around his USB key that contained the education slides. When exactly does the red light go off for you, Steve? Just curious. Listening to Security Now! prepared me for situations like this. I know it's not a good idea to plug an unknown USB key into your computer. I went ahead and held down the shift key when I plugged it in to prevent anything from auto running. Good man. Luckily I did because it was a U3 device, which I can't stand. Either way, it pays to be safe when plugging in unknown USB keys. I know the instructor wouldn't try to do something malicious. But if he wanted to, he could infect all the work laptops in the room. By the way, so could anybody else in the room.

Steve: I was just going to say, if you're passing it around, baby, who knows about the machine that it was plugged into before yours.

Leo: It's not the instructor I'm worried about. Just wanted to share this real world application of Security Now! know-how. Love the podcast, keep up the good work. Was it enough to hold down the shift key, Steve?

Steve: It's really better to disable the feature entirely. So that is optionally done on Windows. Unfortunately, because Microsoft's bias is still to just make it all work by default, it's something that users have to go in and manually do to turn off the autorun of the USB. And there are some technologies that make it less easy to do that. There are USB devices that look like a CD-ROM, not just like a regular static drive. And so those can be a little more tricky.

Leo: Oh, that's kind of what U3 does. It mounts a CD, and it autoruns that. Yeah, so it wouldn't just have to be the professor. The next guy he hands it to could - and it wouldn't be hard - could quickly drag something onto the USB key while he's copying the file off.

Steve: Or Leo, I tell you, I mean, remember in the old days floppies were the viral medium of choice because it was so-called Sneakernet because you'd stick a floppy in, and then you'd copy some data, and then you'd go somewhere else. I mean, floppies are the way viruses existed before the Internet because viruses predated the Internet. And the only way a virus could live would be if some way it could move from machine to machine. And so it was floppies that were virus carriers back then. Well, similarly, a USB dongle or thumb drive is, I mean, a virus looks at that and thinks, hey, there's a way I can escape from this machine. So you can well imagine viruses that are sitting there, waiting for a USB drive to appear and jump over to it as soon as it gets logged onto Windows. So it wouldn't even be somebody else malicious in the class. Their machine could be infected with a USB-propagating virus that will put itself on to any drive that comes along.

Leo: That's a very good point. So, yeah, it could be - that's easy to have happen.

Steve: Yeah.

Leo: Wouldn't have to be maliciousness in your class. Just could be incompetence. Are you ready for Question 12?

Steve: The looming threat.

Leo: The Looming Threat Observation of the Week. John in Ottawa says: Steve and Leo, my item of interest is perhaps a bit more vague than your usual items. My concern arises from my belief that we are at the cusp of seeing a real proliferation of home-use servers emerging onto the Internet. What does he mean? What is he talking about? Steve doesn't like it. He feels it looming over him. In particular I see NAS storage boxes - which I use, I have two of them - migrating into server territory and being targeted for sale increasingly to retail consumers. But are consumers ready? Here's my point, twofold.

One, these next-generation units are becoming increasingly full-featured and economical. To illustrate my point I quote a manufacturer's website: "Build Up Your Dynamic Website." This is an ad. "Web Station runs Apache Web Server that allows you to publish websites with only a few steps. With preinstalled PHP and MySQL you are free to install popular blog or bulletin board programs on your DS107. No advanced IT knowledge is required to build up your community." So I guess I didn't realize they're selling these as web server boxes. And this unit's $260 including hard drive. So that's point number one. That is a little scary, isn't it.

Point number two, because these units seem like an external drive, they're all too easy not to take seriously. Since most people will look to this primarily as a critical data backup, it'll be tempting for them to have the unit central to their LAN. Bad news if the unit's also accessible to the Internet. In other words, if it's a web server and your NAS and your backup...

Steve: And it's got PHP and MySQL running...

Leo: Oh, that is terrible. Oh, the hackers are just drooling over this. Which, let's see, to me this formula adds up to a real potential for trouble. I hope the producers of these products will begin taking steps to inform their customers of security concerns. Perhaps each product should contain a link to the website of a well-known security researcher. Maybe they should just put all of our podcasts on them. I'd be interested in hearing your take on this matter. Thanks for the great show, and keep up the good work. John. Wow. I didn't even know they were doing this.

Steve: Yeah. Doesn't that sound like a bad idea.

Leo: Well, tell us why, Steve.

Steve: Well, I mean, okay. We've talked extensively about what a big problem commercial websites have with security vulnerabilities which are more or less constantly being found on the Internet. And so here we have PHP and MySQL, and they're talking about "free to install popular blog or bulletin board programs." So many of those are exploitable with cross-site scripting problems. I mean, basically we're talking about a huge expansion of some of the most problematic Web 2.0 facilities which are going to be set up and installed by people who have no appreciation or understanding of the security implications. Like, oh, look, I can run my own web server and my own bulletin board and stick it up and have it be public. Oh my goodness, I mean, we're looking at a serious problem in the future.

Leo: So, yeah, I mean, not only do you open up holes into that server by running services on it, but because it's attached to your network and because you're using it as a backup device, let's say, you may have valuable data on that hard drive. It may provide a portal into your LAN.

Steve: Yes, exactly.

Leo: I didn't realize they were selling these kinds of things. Wow.

Steve: Yup.

Leo: We've seen people do this all the time, but it takes some more sophistication. For instance, you know, people will call me and say, well, I want to run a web server. I've DMZ'd my computer so that it can serve to the Internet. It's like, great.

Steve: Or Leo, I mean, it's exactly like what happened when Windows was first stuck on the Internet. It was like, first you had Windows. And then people, like, oh, I want to put my computer on the Internet. And no one happens to notice that I've got my C and my D and my E drives all shared. Because I was never plugged onto the Internet before, I didn't bother with username and password. And so we've got open drives available through Windows filesharing. I mean, that was where all this began and the reason I created ShieldsUP!. So it's like, I mean, we're in the same sort of situation with that sort of a next-generation of nave user who installs this, who then makes this available publicly on the 'Net with a whole bunch of not necessarily secure server-side applications running.

Leo: Wow.

Steve: [Audibly shivering]

Leo: Now, that's not - we're not talking Windows Home Server or anything like that. This is a whole 'nother class of devices now.

Steve: Well, this is a UNIX-based machine. It's got Apache running with PHP and MySQL. And they're saying, hey, feel free to install blogs and BBS.

Leo: I'm sure that's very tempting, somebody who wants to run a website, they say, oh, well, I've got enough bandwidth on my Internet access now, I could just run it out of my office.

Steve: It's going to happen.

Leo: Well, Steve, we've gone through 12 questions, 12 great answers, had a great conversation. But I'm afraid our time is up.

Steve: But we'll be back next week.

Leo: We will. And you said you're going to talk next week...

Steve: About the whole Phorm Webwise technology. There's a very disturbing new trend which is ISPs are actually modifying the pages their customers download. So when I go to a website and look at the page, an ISP has tacked on their own JavaScript, which is being used to monitor me and track me and profile me. Not a good idea.

Leo: Wow. All right. That'll be interesting. We'll do that next Thursday, and every Thursday Steve's here for Security Now!. If you want to get a 16KB version of the show for the bandwidth-impaired or to share with your friends, or you want transcripts, show notes, it's all available on Steve's site, GRC.com. Of course that's where you'll also find all of Steve's great security programs - ShieldsUP!, Shoot The Messenger, DCOMbobulator, the fun Wizmo, all those are free. There's only one program he charges for in his whole life, but it's the one you should buy, and that's called SpinRite. Because it is the best, the one, the only, the one they all copy, disk maintenance utility, drive recovery utility. It's just, you know, if you use hard drives, you need SpinRite. GRC.com. Steve, we'll see you next week on Security Now!.

Steve: Talk to you then, Leo.