SECURITY NOW!

Transcript of Episode #88

Listener Feedback Q&A #18

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

High quality (64 kbps) mp3 audio file URL: http://media.GRC.com/sn/SN-088.mp3
Quarter size (16 kbps) mp3 audio file URL: http://media.GRC.com/sn/sn-088-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 88 for April 19, 2007: Your questions, Steve's answers, #18.

It's time for another thrilling, gripping edition of Security Now!. Actually, because it's Episode 88, divisible by four more than once, it's more than just a thrilling, gripping episode. It's the thrilling and grippingest of all, your questions and Steve's answers.

Steve Gibson: Thrilling and grippingest?

Leo: Of all, yes.

Steve: I like it. Grippingest.

Leo: Well, you know, I feel that way. I feel like they're really the most interesting episodes because they're so wide-ranging. And it's one of the reasons I do talk radio is really people ask, you know, if you answer questions that people are asking, you know you're talking about stuff people care about. So I always enjoy these.

Steve: And in fact, I think also that they benefit from the fact that there are all kinds of different topics, not just, like, we're not focusing on one. Certainly when we're talking about a specific important topic, that's a really good way to do a show, which is what we do for three out of four. But I think also equally good is a potpourri that involves users' questions and also their comments. I'm finding people are now beginning not only to ask questions, but to say things that are really worth sharing with everyone else. So we're mixing a few of those in, too.

Leo: We have very smart listeners. And in fact it kind of surprises me that people, intelligent people, experts in this field, are listening to what we're doing. Isn't that nice? Doesn't that feel good?

Steve: It does. I'm almost a little bit embarrassed by it. It's like, oh...

Leo: It's just this little podcast we're doing.

Steve: But for what it's worth, people who have written, and many people have, who have said things like I've been in IT, you know, for the last 45 years, and I know all about this stuff, but sure enough, you know, I listen to you guys, and I enjoy listening to you. And every single episode has at least one or two things that even I haven't run across before. So that's just very cool.

Leo: And that's the nature of the beast. And I think anybody who's really a good security professional acknowledges that. It's the people who say "I know everything" who I would watch out for, frankly. And that's one of the things I love about you is your thirst for knowledge is insatiable. And you are the first to say, hmm, I don't know, but I can't wait to find out. And that's what...

Steve: Oh, and the web is just so amazing for that reason. I mean, just anything we want to know we're able to plow into. And frankly, Leo, we've covered some things in Security Now! that I've known in more generality than I have presented it to our listeners because, for the purpose of the show, I've had to get much more specific. So the show is driving me into a greater level of detail for areas, things I just haven't had any reason to explore before. So it's, you know, it's certainly good there, too.

Leo: That's excellent. Well, let's, unless you have something you want to talk about from past episodes, delve right into our questions. We've got quite a few here.

Steve: Yes, let's do it.

Leo: Number one, from Donald Park of Laurel, Maryland. And whenever I see a Maryland address I think probably works for the National Security Administration. Okay, that's fine. I won't hold it against you. He has been wondering about bot networks. We talk about those a lot. He says: Would you agree that the bot network epidemic exists because consumers leave their broadband-connected computer on 24/7? The best way to limit the number of bots a bot herder has at his disposal would be to limit the amount of time that broadband computers are connected to the Internet. I do know some people who have various ways of disconnecting from the Internet, whether it's hardware or just doing it by hand. Is that a good idea, do you think?

Steve: Well, yeah. It certainly, I mean, it's not going to solve the problem completely. I very well remember when I was doing my research into denial of service attacks. And I was – remember that I penetrated into an IRC chat forum where a major bot network was in operation, that is, as the bots came online, they phoned home into this IRC channel and basically said, hey, Bot #34926 ready for action.

Leo: Yeah, it’s terrifying to see that because it happens so fast, and so many thousands of machines join up.

Steve: Well, and so what I was noticing, and I did this because I was able to see IPs, is I would track those back out of curiosity to see, you know, what types of computers are these? Are these in universities? Are they on cable modems? Are they dialup? Are they AOL? At the time, and this is now many years ago, many of there were dialup. That is, so even though a user is using a machine which is not on the ‘Net all the time, when they do get on the ‘Net, the bot is aware of it, it connects up and spends as much time as it’s able to doing its bad stuff.

Leo: If you have a 100,000-computer network, even if half of them – and if they’re worldwide, it will only be half of them are offline at any one time, you still have 50,000 computers at your disposal.

Steve: Exactly. Now, one mitigating factor is, typically, well, almost exclusively, dialup people are using a modem, which means that their ability to flood the world with zombie bot traffic is really constrained by the bandwidth of their connection, you know, 56K or so, or typically. But certainly broadband users are a much bigger problem because they’ve got much more upstream bandwidth at their disposal. And, I mean, I don’t really disagree with our questioner here, Don Parker. He’s right that always-on computers on broadband connections, that’s the worst possible scenario from a being attacked standpoint. But it’s, you know, basically it scales back as people turn their machines off, and/or as they have DSL that has a lower rate than cable modem typically and so forth. So it’s really sort of a sliding scale.

Leo: Yeah. In other words...

Steve: It just makes things worse, but it wouldn’t eliminate the problem altogether.

Leo: Not going to get rid of it, yeah.

Steve: Right.

Leo: And I have to say I agree. Intrepid listener Matt in New Zealand made a disturbing discovery: I just wanted to let you know your topic on XSS, cross-site scripting, inspired me – uh-oh – to check whether my bank’s website was vulnerable. It turns out it is. Oh, dear. As if that weren’t bad enough, they do some silly things with their session ID cookie. These two flaws combined allow an attacker to craft a URL which, if visited by a customer, will load a persistent cookie into their browser, even if the customer doesn’t log in after clicking on the link. And what that means is, when they return to the site, the session ID is the same, so the attacker can easily hijack the session and perform transfers or any other bank transactions as that customer. I don’t want to disclose any more at this time as I am in the process of dealing with the bank on the issue. Is this possible that Matt’s found this?

Steve: Well, I also eliminated some additional content from his question because it wasn’t appropriate to share it with, you know, a huge anonymous audience. He was...

Leo: But it was sufficient to convince you of the veracity?

Steve: Yes. Matt really does seem to know what he's talking about. The one thing that, and I wrote back to him, that I would caution him about is nobody likes to be told that some high-value site like a bank's ecommerce banking transaction site is vulnerable. There are instances where well-meaning hackers have found themselves brought up on charges...

Leo: Yeah. Adrian Lamo might be the first name that would come to mind.

Steve: Yeah, now, there are some people who've done it in sort of the wrong way. So you could argue that there's a right way and a wrong way to do it. But technically, unfortunately, our laws now are so restrictive and subject to interpretation that an attorney for the bank could argue that in performing these experiments that Matt had to inherently perform, and notice we don't have Matt's last name here, in doing this he was "hacking the site," that is, he was performing conduct outside the terms and services that are posted on the site. Therefore, you know, even though he's trying to help them, there have been situations where people who have only wanted to be helpful have found themselves on the wrong side of a lawsuit and criminal charges. So it's something, unfortunately, that is very delicate. The only thing I could suggest is that, if other people were to do this – and maybe Matt is being anonymous, I hope he's being anonymous – I would certainly start out that way, you know, somehow bring it to people's attention in a way that's not putting a bull's-eye, you know, on you. But it is really interesting that he, you know, he did some poking around after listening to last week's, or two weeks ago, our cross-site scripting episode, and apparently discovered a vulnerability.

Leo: Unbelievable. Amazing. Wow. Nathan Clark of Allendale, Michigan, he says: I like Vista. But I'm having a little trouble with it. I recently purchased a new laptop. It came installed with Windows Vista Home Premium. I actually enjoy the OS more than I thought I would. I have noticed one key problem. Occasionally when a program is launched that requires administrative rights, the system won't switch to the protected mode correctly. The only way to get past this is to log out and log back in. I think when he says "protected mode" he's talking about UAC; is that right?

Steve: Exactly.

Leo: I was curious if you have any advice and wanted to raise awareness about this problem. I haven't seen this myself. Do you know what he's talking about?

Steve: Yes. There have been some reports of this. And having programmed UAC for my little SecurAble freeware, I understand what's going on. And it's a consequence of the way Windows is sort of trying, again, to give us the best of both worlds. As we know, when you log in, you are inherently protected because you've got non-administrative credentials. And then the User Account Control attempts to elevate your credentials. Well, it's very possible to write code for which that will not work. For example, code could be – and this gets into the entrails of Windows. But there are ways you ask for things from Windows which are marked with your rights at the time they're granted. So if a program were to ask for things before Windows realized that what it was going to do with those things are privileged, the rights granted to those initial tokens would be limited, and they're not dynamically regranted as a consequence of UAC.

So basically this is a perfect instance of, again, what Microsoft is trying to do is solve the problem of people normally running with too much rights, more rights than they really need for most workaday things, and then not making it necessary to log out and log back in, in order to do things that need administrative rights. Mostly they've solved the problem in an elegant fashion. But as is always the case, there are incompatibilities with some programs that assumed the rights you have when you log in are the rights you're ever going to have during that log-in session. That's what Microsoft deliberately changed. And what Nathan is seeing is a place where a particular program isn't flexible enough to accept the greater rights be granted it because Microsoft just isn't doing the right thing because Microsoft has changed the rules, essentially.

Leo: So it's a non-Vista-aware program, essentially.

Steve: Yes. And, you know, most programs today are non-Vista-aware. And Microsoft is trying to be compatible with them. This one they stumbled on a little bit. So unfortunately either the...

Leo: Look for an update.

Steve: Yes. The program could be updated for this to work, or this may well be something that just is not going to be happy if it's normally running with less privilege. And it's not going to understand that suddenly it's got more privilege now than it did when it was initially started up.

Leo: Right, right. Donald Parker in – I'm sorry. Back to Question 4. Don't want to skip Chris in Connecticut. He's been wondering for a while: I've used your Shoot The Messenger and Unplug n' Pray – these are free security tools Steve offers at GRC.com – to disable these unnecessary services. I guess Shoot The Messenger is that Windows Messenger service that actually Kevin Rose discovered was a flaw in Windows, and UnPlug n' Pray is that Plug and Play service. What's to keep malware from just turning them back on? In a similar vein, I've noticed that some software, Retrospect Backup, for instance, offers to open the necessary ports on the Windows firewall when it installs. Well, what's to prevent malware from doing the same thing and not letting me know? I know that once malware is running, all bets are off. But some of these measures are ideally supposed to help control the damage. Are they helpful? Or are they just window dressing? Oh, good question.

Steve: I thought it was a great question. Okay. First of all, most of the things that I was doing during pre-XP and also now pre-Vista, that is, you know, before XP but all through XP, I was doing some quickie little freeware, in some cases to fix a problem, often to turn off a service that Microsoft had running by default which was known to have security vulnerabilities. So, you know, Shoot The Messenger turned out to be a problem. Not only was it causing pop-ups on users' systems, but it turns out that because it was a service with an open listening port that was just collecting anything that came in, there were ways that it could be taken advantage of. The Universal Plug and Play was a service that was found to be vulnerable. And, again, I was just annoyed with Microsoft that they had all these services running by default just to make things sort of work automatically, even for people who didn't need them, which really represented a problem.

Now, what's changed is, with Service Pack 2, Windows' own firewall is on by default. So even if you had open ports on your system, they are behind a firewall. Now, the beautiful aspect of the second part of this question is he's talking about how some programs are able to programmatically and silently open ports through the firewall. That's absolutely true. And it's one of the reasons that having a router is still a good thing, even if you've got a PC with a personal firewall. Any software that wants to, whether good or bad, has a documented and officially supported means for basically bringing the firewall down for all intents and purposes.

It may not want to turn the whole thing off because then all kinds of bells and whistles start warning you that your firewall's been turned off. But it doesn't need the whole firewall down. It can just open one hole through the firewall that allows external hackers to get in through that route. So that cannot happen, however, if you're using a NAT router and you turn off the NAT router's autoconfiguration, which is Universal Plug and Play in the router.

So, you know, he's correct that, essentially, I would say that you could turn these services off if you want to using my freeware or just, you know, shutting them down using the normal service management panels in Windows, both XP and Vista. But it's really not that necessary now. And you can have side effects if things depend upon those services. As long as you're behind a NAT router with Universal Plug and Play disabled, it's not possible for software inside the network to go reconfigure the router to make it vulnerable.

Leo: Okay. But on the computer it can do anything it wants.

Steve: Yes.

Leo: And that's the real problem. Once you've been compromised on the computer, you're kind of out of luck.

Steve: Yes. You never then are really able to trust it again because you don't know what might have happened to it.

Leo: So they're not Window dressing, they're a barrier. But since a lot of hackers know enough to do this – this is why I always say that a software firewall isn't necessarily very useful because they can often be disabled.

Steve: Yes. And in fact we're also seeing the typical refractory delay from introduction to implementation of exploitation. You know, sure, XP's firewall's on. Before Service Pack 2 it wasn't, so it was never a problem, so hackers weren't worrying about it. Now that it is on all the time, and Microsoft has a published API, an Application Programming Interface that allows software to open holes, for example, you know, you would want your instant messaging program to work easily without requiring that people had to manually reconfigure their firewall. So good programs are doing this on behalf of their users. But so can malware.

Leo: Right, right. Donald Parker, Laurel, Maryland, has a solution to the bot network epidemic. He writes: I seem to be the only person on the planet – see, whenever I read that – all right. I'm just going to keep reading. I seem to be the only person on the planet that knows how to defeat bot network herders. I can't get anyone to listen to me. The botnet epidemic cannot be solved with a software solution. A distributed attack method must be met with a distributed defense strategy. No one seems to realize this problem exists because of broadband technology. With dialup computers, they were only connected and susceptible to hackers when they went through the long process of connecting to the Internet. With broadband – this is the same kind of thing as our first question. With broadband, people leave their computers powered on when they go to sleep at night and leave for work in the morning because they want the convenience of instant access to the Internet. Hackers use their computers more than their owners do. No broadband-connected computer should be allowed to connect to the Internet without some method of automatically disconnecting the computer from the Internet, whether the user shuts down their computer or not. If hackers don't have so many potential bots at their disposal, the threat would decrease significantly, and cyber law and order would be restored. Steve, do you want to just shoot this down now, or...

Steve: Well, we actually did. I guess I covered this in answering the first question much more thoroughly than I had intended to in the first question, thinking of this question. Basically he’s just saying that broadband-connected machines are the problem, and that they should not be left turned on or connected. The reason I wanted to bring this up is this does provide sort of some additional fleshing out of that idea because in fact there are many reasons why people do leave their machines on all the time and connected to the Internet. For example, you want to be able to download large podcasts in the middle of the night and not sit around having your bandwidth tied up by doing that.

Leo: And we see this all the time, a lot of botnets are in universities or businesses where computers are left on all the time for perfectly legitimate business reasons. And there’s always going to be enough of those on. And by the way, if you have a big enough pool, let’s say 600 million Windows users, there’s sure to be, let’s say 100 million on all the time. Plenty. Plenty. Right?

Steve: Yes.

Leo: I mean, even if they’re not on all the time, if they’re on now, on at any given moment I guess is what I should say.

Steve: Right. And also, again, it seems to me that, well, I’m not one who recommends turning machines on and off all the time. All of my main machines are on all the time. I know that’s becoming nonpolitically correct from an energy consumption standpoint. But it is really wear and tear on the system to heat it up and cool it down and heat it up and cool it down, as opposed to just letting it stay idling. And of course newer machines are much better about their energy consumption when they’re not in use. They’ll spin down the hard drives, they’ll blank the monitors and, you know, go into an energy-conserving mode. So, you know, I’m not someone who thinks you should be turning your machines off all the time. And certainly unplugging them from the ‘Net is a pain, too.

Leo: Yeah. Aaron Burns in the United Kingdom is seeking a bit of assurance: I was wondering if WPA-PSK is at all crackable? I’ve heard it can be cracked offline using the brute force dictionary attack. But if you use a 63□character random password you ’re virtually uncrackable; right?

Steve: Right.

Leo: Next question. No, people will ask me that almost every week, well, I heard WPA was cracked. And it was cracked in the same way that you would crack, say, PGP or any encryption tool if you use a bad password.

Steve: Yes. The proper thing from a crypto science standpoint to say is that the only known attack on WPA using a preshared key – which is what the PSK stands for, WPA with a preshared key – is that if the key can be discovered. The only way to discover the key, again, the only known way to discover the key is by taking a chunk of data that was encrypted using that key, taking it offline, means take it home with you and give a powerful computer the job of trying either all possible keys or all mixes of words in a dictionary, the so-called dictionary attack. These are brute force. And as long as you've got a long key that is just looks like gobbledygook, I mean, the kind of keys generated by my passwords page, GRC.com/passwords, I on-the-fly generate keys for people over a secure connection that cannot be sniffed which will never be generated again. And you can refresh the page a few times, grab pieces of keys if you want to, like, make up your own, so you're not even using one that I gave you, but it's just raw password material. If you use that, you are doing everything you possible can to be secure.

The reason I also enclosed this question was just last week huge progress was made in cracking WEP, the original insecure technology. And that's going to be the topic of next week's podcast is looking at what happened to WEP. It just got a lot worse. So anyone who's still using it thinking that, oh, it's good again, it's still good, it turns out you can crack now, using the update, any WEP-encrypted network in around a minute.

Leo: Wow. That's really bad. And just to reiterate, WPA-PSK is perfectly secure if you use a good, strong password, one that, you know, not in the dictionary, not your dog's name, not iamsosexy, something a little harder than that. And of course Steve's got that great password generator for you.

Steve: Yeah. Mine is one of those, I got it from my own page. And, I mean, I don't know what it looks like. It's nothing I can remember. But I have it in a text file. And when I'm configuring a machine, I've got it on a USB dongle, I'll stick it in, open the file, drop the password in, and then take it out. So I'm not leaving it on the machine. Because the password is then hashed into an actual key that the WPA system uses. So, you know, mine is not something I even know. I don't know my own password. It's just 63 characters of junk. But I'm as secure as I can be with WPA using a preshared key.

Leo: That's one of the nice things about WPA. You only enter it once, and that's it. You don't have to reenter it again and again.

Steve: And XP and Vista really like it. They work very, I mean, they're happier with WPA than they are being forced back into WEP encryption.

Leo: Oh, really. How interesting. Mark Jones of Midland, Minnesota says: I have a question concerning vulnerabilities such as the recent cursor zero day. I'm careful. I keep up to date. I also find myself diagnosing other people's machines, especially my less-than-computer-literate relatives. The nightmare scenario, next to the family fortune going to Nigeria, is that Pop's machine becomes a zombie. How do you know whether your machine is a zombie? This is good. I want to know this. Will processes be running continually? Do these machines just phone home and start up? Is there something to look for?

Steve: There are a couple things to look for that we've touched on in earlier episodes, but it's always worth, you know, making this current and reminding people. Two ways you could see a zombie. And this is what I do if I encounter a machine for the first time. I did a week ago, I went over to a friend's house, exactly like this, not a relative, but a friend who said, you know, something feels wrong. The system really seems slow. Could you take a look at it?

So I ran Task Manager, which is, you know, it's in every version of Windows, and you can also get it by doing Control-Alt-Delete, and it pops up the little window from which you can select to run Task Manager. And I look at the process list, which is everything that Windows says is running. Now, many of these things are cryptic looking. You know, there are things you may not be familiar with. But over time you can develop some familiarity with them. Of course, you know, I live in there. So I was able to use Task Manager to just sort of browse down through the stuff he had running and see whether there was something that really looked suspicious.

If you do find something suspicious, you can then – what I'll do is use the search, the file search built into Windows, I'll put the name of the program, because it'll say, you know, pqrdzx.exe. It's like, okay, well, that one doesn't ring a bell. So I'll just cut and paste that or type that into search, have it find the file for me in Explorer. Then I can right-click on it and look at its properties, and the version tab will tell me who made it. And almost always it's like, oh, yeah, that's my, you know, the Corel Photo something or other; or it's the driver that came with my HP scanner; or, you know, there'll be something that is immediately recognizable about that, that then disqualifies it typically for, like, further concern. If it's not something that has a version tab, and you can't recognize what it is, then even so, normally where it's located on your main system drive, that will often give you a clue. When you search for it, it'll show you where it found it on the disk. It's like, oh, okay, well, that's under an application that I installed a week ago, and so I know what that is. So looking at the process list is one good way.

Now, that fails if what you've got in your system has rootkit technology because, as we know, rootkit technology specifically is rootkit technology because it's able to hide itself. One of the things that really good kernel-level rootkit malware will do is, once it's running, it will go down and edit the data structures which govern running processes. There's a list, literally a linked list of processes where the first one has a link to the second one that has a link to the third one and then so forth. Well, what the malware will do is, when it finds itself running, it will read through the list until it finds the one that's pointing to it, and it will instead point that to the one it's pointing to, essentially unlinking itself from the list. When it does that, then simple task enumerators like Task Manager won't see it. It just won't show up at all. So while Task Manager is useful for sort of giving yourself sort of an appraisal of what's going on, it's not something you can absolutely count on; but it works more often than not, enough so that it's the first place I always go.

The second way that something that is running in your machine and playing with a network will show up is with the so-called "Netstat" command, which we did an entire episode on. So essentially there are commands you can give when you open a DOS box window, Netstat followed by some parameters, that will quickly show the activity, the network activity of your various network-using programs, processes again, running in the system. And you can see if ports are open, if connections are made; and, using the proper commands in XP and Vista, it will show you the name of the process doing the communication.

Now, what's cool about that is many fewer processes should be listening with open ports or talking. So Netstat can be used to narrow this down dramatically. And I would refer people back to the episode that we did entirely on Netstat. Anyone who wants a little refresher on this, we really gave it good coverage about a year ago.

Leo: So it really isn't like Norton Antivirus or something for detecting invasions. There are just too many possible ways you could be co-opted.

Steve: Yes. And there are always things that are happening before the AV companies are up to speed. And that's what bites people.

Leo: Yeah. Babak – oh, by the way, I think AVG, the folks who do the free antivirus, have just announced a free anti-rootkit, although there are several other choices, as well. But it's nice to have another anti-rootkit, or at least a rootkit detector.

Steve: The more the merrier.

Leo: Yeah. Babak in San Jose, California asks: I was wondering if you could comment on which hard disk defrag and maintenance tools you prefer. I'm referring to tools that run under Windows. What do you use to defrag and maintain your disks? Besides SpinRite.

Steve: Well, certainly SpinRite for sector-level maintenance. There is a defragger that you'll know well, Leo, probably. It's still around, and it's my favorite. It's from a company called Golden Bow Software.

Leo: Oh, that's still around.

Steve: That's Vopt.

Leo: That's the one that Jerry Pournelle would just sing the praises of.

Steve: Yes. And I'm still doing so. I've become less and less impressed by Microsoft's defragging. I find that when I use the defragger in XP or especially Vista, Vista they just took all the meat out of it. I mean, it's just amazing. I'd like to see...

Leo: You're not supposed to really need it, though, with NTFS, are you?

Steve: I know. But just it feels good. It just feels – I don't want all these little red teeth all over my hard drive. I like everything to be in little blue blocks. And so anyway, Vopt is my favorite defragger. It runs, I'm not sure about Vista, but certainly all the way through XP. And I'm sure they must have a version now because Vista's been around long enough. Anyway, they've kept it alive. It does a very good job. It's got a whole bunch of really nice features.

The thing that's built into Windows, there's a disk cleanup which is something I also recommend that people use because it empties your trash, it throws away a whole bunch of the cached stuff that your Internet Explorer browser has downloaded. I mean, and many times, in many cases, hundreds of megabytes of just stuff you really don't need. And so when you run this, it will get rid of that stuff. Vopt has an even more thorough version of that that does a better job, a deeper job of cleaning up really the debris that the systems accumulate over time. And it is the case also that defragging your system will mean less wear and tear on your hard drive. It's the head moving which will create wear and tear. And that's what causes heat. If a drive is just sitting there idling, or not moving the head a lot, it will run cooler than if the head is flying back and forth all over the place.

Leo: It is possible to over defrag and make it worse. In other words, you shouldn't run these things every day because then you'd be doing the opposite. You'd be overexercising your drive.

Steve: Yes. You'd be putting the drive through more work than it wants. And there is also something, a sort of a SpinRite-ish benefit to defragging, and that is that it does cause your system to revisit areas of the drive, and it allows the drive to see that it's got problems that it might otherwise not be aware of. SpinRite, of course, does it from zero to a hundred percent. I mean, it does the entire surface. Defragging doesn't guarantee you that, but it's another sort of a good thing to do, just sort of, you know, shake the dust loose from the drive.

Leo: And by the way, Vopt is available for Vista, does work with Vista, $40 at Vopt.com.

Steve: The other one that I love is a funky program called SpaceMonger. There was a program that I was using for years called DriveMapper. This is a really cool visual technology. Imagine that you – and it's visualization, which is what I love about this. How do you know, when you notice that your hard drive is, like, fuller, like, by a lot than it was a month ago, how do you know where the space went? Well, SpaceMonger, which is free, the reason I'm mentioning it, is v1.4 is free. Sean the developer is now charging for 2.0, which does a whole bunch of more stuff than version 1.4. But frankly, 1.4, the free version, is all anyone needs. If you just put SpaceMonger into Google, it'll find it for you.

What this does is, it shows your hierarchy of folders and files on your computer in a hierarchy, a nested hierarchy of rectangles on the screen where the size of the rectangle is the relative size of the file or directory. So when you let this thing run on your computer, the first thing you'll see is, like, big chunks of stuff which are like, you know, videos you downloaded three years ago and forgot about, or a whole tree of audio WAV file songs buried deep under My Documents, My Songs, My Audio Albums or something, you know, way down sort of off the chart. But there it is sitting there, occupying 650 megabytes per CD, and you just forgot about it.

So what's so cool about SpaceMonger, and in fact this is the thing I also use, I downloaded it on the spot when I was over at my friend's house a week ago when he was saying that his computer was running so slowly, and we found exactly that. This guy had downloaded DVD content from a couple years ago and forgot about it. He had, you know, each DVD is 4.7 gig. He had, like, six or seven DVDs sitting on his computer, just forgot about it. But boy, it stood out like a glaring problem under SpaceMonger because it makes it so easy to see where the space went. So I really recommend it without – it's free. I recommend v1.4 without hesitation.

Leo: And there's a similar OS X application. And I thought it was DiskView. I'm trying to remember, something X, anyway, that'll I'll find for you. We've mentioned it on the show before. It's very – it's really a useful thing. To see what you've got is so handy. I think it's actually, ultimately was based on a UNIX X Window application way back, from way back when, that would show this beautiful graphical display of what you've got.

Steve: Well, it's funny, too, because the original one, this DiskMapper, was all burdened with intellectual property claims and patents and all this stuff. Well, it turns out that Sean found that this had all been done in a university environment, and all this technology, this notion of nested size-relative rectangles, is in the public domain. It was put there by the people who developed it so nobody could claim ownership of it. It's very cool.

Leo: Let's see. So any other tools? Let's see, that's a defrag tool is Vopt, maintenance tool – oh, SpinRite. There would be a good choice. If you have a hard drive problem, just keep SpinRite in mind. Will you do me a favor, that's Steve Gibson's, not free, but excellent utility for protecting your hard drive. It really is, I mean, let's face it, a great tool.

Question 9, Jon in Katy, Texas: I'm on a small corporate LAN with a Cisco router and a separate stateful firewall. How can I allow outbound SSH connections, but block the building of outbound tunnels to defeat my proxy firewall settings? Outbound SSH tunneling seems like a great way to exfiltrate data. People do it all the time. It's a great way to build a tunnel; and then, you know, the IT guy can't see what you're doing.

Steve: Yes, and there is nothing he can do about it.

Leo: No. If you're going to allow outbound SSH, you're allowing tunneling.

Steve: Yes. The only thing that, well, I mean, the only thing that I could imagine doing would be to try to get control of your clients. But even the client firewall on the computer, it's going to see this connection originating from inside and allow it to go outbound. So it'll go through the client firewall, through the corporate firewall. And, you know, if he wants to allow outbound SSH connections, then, you know, they can be tunnels. And because they're inherently encrypted, I mean, there's no way to see inside those, tunneling is absolutely possible.

The only thing I could imagine would be, and this is, you know, hard to implement and sort of at corporate level, would be if he were to proxy SSH and force his users to use a certificate that the proxy server had. Then essentially that would allow him to perform a corporate authorized man-in-the-middle attack, essentially terminate the SSH connection at the corporate firewall proxy, decrypt it, filter it, and then establish a second leg SSH connection out to the remote destination. So basically you would be breaking the tunnel open, running it through permission filters, and then reassembling it, you know, connecting it back into a tunnel and sending it out. But again, you know, it's possible; but it's certainly not something easy to do.

Leo: He really could use a corporate VPN or something else like that that would be probably a better solution, right, than setting up SSH. Same problem?

Steve: Well, yeah, but a corporate VPN, I mean, a VPN is a tunnel, yeah, it's an encrypted tunnel. If you allow that to happen out through your firewall, you have no way of monitoring what your users are doing.

Leo: Jason Anderson in Wisconsin is a savvy webmaster who got caught by cross-site scripting. Oh, no.

Steve: This is the one I referred to last week, Leo.

Leo: Yeah. I heard your last podcast on cross-site scripting. My ears perked up when you said that Webmin had a vulnerability. Lots of people use Webmin. I have a server that runs Gentoo Linux, and it recently became compromised because of this Webmin vulnerability. Wow. I installed it and forgot about updating it. See, this is what happens when the newer versions came out.

Steve: Exactly.

Leo: And a hacker found my user credentials in Webmin, SSH'd – oh, boy – into my server. It then proceeded to set up a phishing site in one of the subdirectories of the Apache server. The phishing site was a PayPal site. After a week or so I got a call from RoadRunner saying that I was hosting a phishing site on my web server, and the domain name was being revoked. I logged onto my server, removed the phishing directory. I then checked the bash history to see if they did anything else that would compromise my server anymore. Remember, though, folks, they're going to change these bash logs and the other logs.

Steve: Exactly.

Leo: I also took a look at the SSH logs, saw that it wasn't a brute force attempt, they just logged right into my server with the right credentials the first time. I then began to think about what software I had installed on that server and narrowed it down to Webmin. When I checked the change logs on the Webmin site, I found that my version of Webmin was below "1" – P.S.: SpinRite has saved the day at our office many times. It's great to be able to tell high-ranking university professors we're able to save their hard drive and their millions of dollars of grant applications – and upgraded immediately. What's "below 1" mean? I don't understand. My version was below, oh, below 1.1. Oh, my gosh. And upgraded immediately.

This painful lesson has taught me that it doesn't matter what kind of software you have installed on your server. It has to be updated on a regular basis. After this was all over I set up a script that would check for software updates every day – many Linuxes do this now, by the way, automatically – and then email the list to me so I can keep track of what needs to be updated and as a reminder to update my server. Thank you for Security Now!. I love the show. Keep up the great work.

Now I'm going to read the footnote which I shouldn't have read when I read it. P.S.: SpinRite has saved the day at our office many times. It's great to be able to tell high-ranking university professors that we are able to save their hard drive and their millions of dollars of grant applications. Another happy customer.

Steve: So that was an interesting posting. I thought it was really interesting. I wanted to include it because, you know, we mentioned Webmin as a vulnerability. And here, from all the evidence that he has cited and his own forensic analysis, you know, he's told by his ISP that he's running a PayPal phishing site on his server that he had no knowledge of, and then goes in and tracks it down to them using SSH to log on to his server remotely. And then by further pursuing the parsing of his logs he sees that they logged in perfectly the first time, no brute force into his SSH tunnel, but first time they were able to get a command shell; and from there they were able to do everything that they wanted to. So really interesting. And it was caused by exactly what we were talking about two weeks ago, a cross-site scripting vulnerability. And he's a happy SpinRite user.

Leo: What more could you ask for?

Steve: Very cool.

Leo: College student Brian K. at Penn State has an interesting dilemma: I've been playing around on my Mac lately, and I've recently installed Vista into Boot Camp to see what's going on with this latest version of Windows. Of course, why not? I was also playing around with ShieldsUP. I noticed that the IP it showed is the same IP I have in my computer. I'm in a college dorm, and I had to give them my MAC address and such to set up while they gave me a static IP to use. Seeing this IP in ShieldsUP makes me wonder. Am I not running behind a router? Do I have a direct connection to the Internet? Should I be worried? I know when I'm on campus wireless they make you use a VPN, but that doesn't work from the dorm. I just want to make sure I'm secure, even though I'm running a Mac. Why is he seeing his real IP address?

Steve: That's a great question. This is evidence of the fact that he is not behind any sort of a NAT router. When he's using ShieldsUP, when any user is using ShieldsUP, the IP that

ShieldsUP shows, you know, my free security testing facility, is that public IP from which ShieldsUP sees a connection coming to the server.

Leo: That should be the router address; right?

Steve: That would be the router address, exactly. So any home user behind a router, whereas their computer will be 192.168.something.something, the ShieldsUP will show you, you know, the public IP, 38.172 or, you know, whatever your public IP is. But something that's not a 192.168, it's not 10., it's not 172, it's a public-facing IP. The fact that his campus IT people wanted his MAC address means that they probably are running a DHCP server where they have built a database of associated IPs and MACs. So his computer is set up to obtain an IP address automatically. When it does that, it says, here's my MAC address, what's my IP? Well, they're giving him a public IP which they've associated with his MAC address. That way...

Leo: That's wrong. Is that misconfigured?

Steve: No, it could just be the way their campus network is set up. I mean, they must have a big network where they can afford to give public IPs to the students in dorm rooms.

Leo: They have what we call "static IPs."

Steve: Yeah, it's like it's a cool, real IP. Now, the problem is, of course, you don't have the protection that we're pretty much now taking for granted is going to be between users and the Internet in the form of a NAT router. So it does mean that, you know, he asks, what does this mean, am I vulnerable, well, you know, it means you are entirely dependent upon your software firewall running in your computer. And that would make me nervous. Given that NAT routers are now in the $49 range, it's so simple to stick a NAT router on your IP connection from...

Leo: Oh, yeah. He can just put one in his room.

Steve: Exactly. Run your own local NAT router. And then you get the advantage of Wi-Fi, if you've got a wireless NAT router, and we know how you want to be secure that way, and then also being able to plug multiple computers into your own little personal local router. So that's definitely what I would do. It's funny, you know, maybe it's that I'm obsessed about security. But Leo, the idea of, you know, a direct, unfiltered, un-NAT-protected connection, I mean, even you sound like it makes you a little nervous.

Leo: Well, it does, yeah. If it's really the case. I mean, isn't it possible that he could still be protected but he's got a static IP address?

Steve: Well, yeah, that's a very good point. The campus could have a firewall which is firewalling all inbound traffic.

Leo: Yeah. They must be doing that. I mean, it's Penn State. It's not some...

Steve: It's not Podunk State.

Leo: No, it's Penn State. I'm sure they're doing that. I can't believe they wouldn't be. But of course we know we can never assume that anymore.

Steve: A lot of strange things happen in universities.

Leo: Yeah. Wow. Oh, I know why they do this. Well, they don't need to do it to do this. But I think maybe this is kind of to tell people, hey, if you are going to do filesharing or anything illegal on our network, you're kind of out there with your personal IP address. Maybe that's why they do that.

Steve: Oh, it's interesting, yeah.

Leo: Because you know they're filtering it. Anyway, just a thought. Question 12, Victor Rodriguez of Arvada, Colorado asks: I was listening to your episode about cross-site scripting, couldn't help but wonder why someone would want to use my browser when I visit a given website to seek out and find cross-site scripting vulnerabilities on other websites. Oh, he's talking about, what was it...

Steve: Jikto.

Leo: Jikto. Why wouldn't they just seek out those vulnerabilities from their own systems?

Steve: Well, of course the reason is that they would rather not expose themselves to anyone backtracking. Essentially they've turned your machine, not into a malicious traffic-generation zombie, like so many DDoS or spam-generating systems are. They've turned your machine, your browser, using browser scripting, into a vulnerability scanning system so that, if anyone sees, hey, wait a minute, somebody is poking at m web server looking for cross-site scripting and other sorts of vulnerabilities, well, as we know it's not possible to spoof a TCP connection. You can spoof UDP packets like are used in floods. But you have to actually have your real IP connection known by the server who you are trying to exploit. So if they've got logs, they'll lock down your IP and could pursue you directly, saying we know what IP you are. They get a court order to force your ISP to release your personal data, and the FBI will be knocking on your door.

Leo: Well, there's a second value, is that you can assign an army to this task.

Steve: Exactly. Exactly.

Leo: You can have much more success because you've got millions of people out there doing your work.

Steve: All working in parallel.

Leo: It’s the ultimate distributed computing, without your permission.

Steve: Yeah.

Leo: Steve, we’ve gone through 12.

Steve: And we’re on to Episode 89 next week.

Leo: Wow. You’re going to get to 100 before we know it. I’m going to have to work on some great prizes or something. I’m sure we can come up with something. Do we know what we want to do next week, or...

Steve: Yes, next week we’re going to talk about a dramatic breakthrough in Wi-Fi security cracking, the WEP protocol, which used to take a long time to hack, now takes less than a minute. And we’re going to talk about how that works.

Leo: Great. More security coming your way. We thank you so much, Steve Gibson. GRC.com is the place to go for the 16KB versions of this show, the transcripts, the great free security software Steve offers like ShieldsUP, SecurAble, and of course, lest we forget, the great SpinRite, the ultimate disk recovery and maintenance utility. If you’ve got a hard drive, you need SpinRite. Get your copy at GRC.com.