SECURITY NOW!

Transcript of Episode #146

Listener Feedback Q&A #42

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-146.mp3
Quarter size (16 kbps) mp3 audio file URL: http://media.GRC.com/sn/sn-146-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 146 for May 29, 2008: Listener Feedback #42. This show and the entire TWiT broadcast network is brought to you by donations from listeners like you. Thanks.

This is Security Now!, time to talk about protecting yourself on your computer, online with your credit card and every other way possible, with Mr. Steve Gibson. Hey, Steve.

Steve Gibson: Hi, Leo. Great to be with you again.

Leo: Good to talk to you. Now, let's see. This is an even-numbered episode; am I right?

Steve: Yup, it is, 146. In fact, we are ten weeks away from finishing our third year.

Leo: Really? It's been three years?

Steve: Can you believe it? It's, like, where has it gone?

Leo: That is amazing. Well, you were the second show I did, right, I mean, after TWiT?

Steve: I think that's the case, although they were coming on pretty fast.

Leo: Amber and I did one. I think you were first because I remember it hit me, I could do more than one of these. What a fool that I am.

Steve: And, exactly, and if you're doing one, than adding a second one is way less than twice as much work, so...

Leo: Right. Well, and that's true. It's all incremental, isn't it, yeah. And then turning on a video camera, incremental. The problem is I've been incrementaling myself to death here. It's like, can't take it anymore.

Steve: Yeah, I've been watching you on camera. I think you really like this whole...

Leo: I have fun. I have fun.

Steve: ...Leo studio routine.

Leo: As long as people understand that sometimes I have to stand up and go somewhere and, you know. But they seem to entertain themselves quite well, actually.

Steve: There was, oh, it was last Friday that you guys, I think you cancelled your recording with Paul. And I thought, oh, I'll just see what's going on. And, I mean, the chatroom was just going crazy all by themselves. So they really don't need you.

Leo: Yeah, Paul was on deadline for his book. He couldn't record on our normal Thursday. So he said we'll do this tomorrow, I've got to get my chapter out. Yeah, they're perfectly happy in there. In fact sometimes - a couple of times, just as an experiment, I've turned off the video but left the Stickam chatroom going. And there's usually 900 people in there all night long.

Steve: Go figure.

Leo: I don't understand it.

Steve: It's the power of your celebrity.

Leo: No, it's not me. No, and I think that's really an important point. It's the community. And I think that's what's really exciting about what's going on with the web in general is it's all about community now. It's not about, you know, the old media it was about celebrity. I think now it's about community. I like that.

Steve: That's a very good point.

Leo: Yeah, I really like that. So we're going to talk today about a variety of things, as we always do on an even-numbered show. You've put together it looks like 12 great questions from all over the world.

Steve: Submitted by our - these are issues and mostly questions, some little info tidbits submitted by our listeners.

Leo: Yeah, I shouldn't just call them "questions." It's a variety of stuff, yeah.

Steve: Yeah, yeah. And I did want to mention, many people mentioned that apparently I broke the URL for the feedback form. We've been saying GRC.com/feedback. And my system appends the .htm to the end. I added the technology so that, if a page were given without the .htm, the server would add that. And something I did in the last couple months broke that. And so it didn't limit our listeners. They still managed to send 375 questions since I last pulled them two weeks ago. But several of them did mention that they had to manually add the .htm. So by the time anyone hears this - I haven't had a chance to get to it because I just read that this morning - it'll be fixed. So whatever it was I did, I will fix it.

Leo: It's a little scary that you don't know.

Steve: Well, I've been playing around, working on some very cool technology that we'll be talking about in the show before long. And it involves classifying assets on the site for the purpose of analyzing browsers' behavior in handling cookies depending upon what type of asset it is. And so something in there I broke. And it's like, okay, fine, I'll go figure it out and fix it.

Leo: Apache, which is the web server that I use and most UNIX types use, has a very nice module called "mod_rewrite" that handles this URL kind of massaging. You're using IIS, Microsoft's server. What does it do? Do you have to do something special?

Steve: No, it does nothing. I'm actually - all of this stuff I do, ShieldsUP!, the eCommerce system, the Perfect Paper Passwords, the Perfect Password Generator, all of that is my own code.

Leo: Oh, you're not running - but you're running IIS as your server, though.

Steve: Basically yes. I've got IIS, and they have an API called IISAPI - or actually there's no two I's, it's just ISAPI. And it allows me to insert my own code in front of the web server and behind the web server. So essentially I've completely encapsulated Microsoft's original web server in my own code, all in Assembly language, of course. And so all the stuff that the site does, it does because I've added code for it.

Leo: Right, right. Well, that's interesting because it's so nice, I use the Apache mod_rewrite all the time. You can do redirects with it. But you could just take a URL that's incoming - for instance, a lot of URLs on things like WordPress and so forth are really ugly, you know, they have .php, and they have queries and so forth. And you can just massage those automatically, on the fly, to something looking much, much nicer. And same thing with incoming that stuff. Anyway, enough of that. Enough of that. So are there any updates from last week, anything you want to...

Steve: Always.

Leo: Always.

Steve: Got a whole grab bag of goodies. Somebody in the Security Now! newsgroup that we have mentioned my comment about how when you disable the phishing filter on IE7, it disables the display of the EVCerts, the Extended Validation Certificates. And it turns out that you can - there is a way to disable the phishing filter and still have EVCerts displayed, which many people may want because they might want the benefit of knowing what's going on with EVCerts while, for whatever reason, not wanting the phishing filter to be enabled. In the advanced settings of IE there is an option down toward the end of the very long list, in the last section, called Check for Server Certificate Revocation. And if you turn off the phishing filter, that's not checked. You need to check it and then restart that instance of IE7. And you get EVCert display back. So anyone can get IE to show EVCert presence, even if they've got the phishing filter disabled, by just...

Leo: Wait a minute. Say that again because that was the most arcane thing I've ever heard of. So you have to...

Steve: Who knows why, but that's what Microsoft has got it set up for. So there's an option in their advanced security configuration called Check for Certificate Revocation.

Leo: Okay, so you want to check for certificate revocation.

Steve: Yes. And if you do, then you get EVCerts.

Leo: So if you uncheck that, turn off phishing filter, then check it again, it'll come back on.

Steve: Yeah, or just turn off the phishing filter and then go there and...

Leo: And recheck it.

Steve: Yes.

Leo: Got it.

Steve: And enable the checking for server certificate revocation.

Leo: So the phishing filter just disables a bunch of things, including that. But you can reenable that one feature.

Steve: Yeah. Actually I didn't verify whether the phishing filter changes the setting of that. It may be that it's not normally turned on. The phishing filter also enables EVCerts. What I do know is that, with the phishing filter off, and you enable the checking for server certificate revocation, you've got EVCert display on IE7.

Leo: Got it. Got it.

Steve: Also I was talking with some bit of excitement about Service Pack 3 of XP last week. And I have been bitten by it, and my tech support guy Greg has been bitten by it. I actually had to remove it from this main, brand new, recently built system of mine because twice - and that's all it took - in the day after I installed Service Pack 3, my Start Menu died. Which I have never had happen before.

Leo: That's a weird thing, yeah.

Steve: I could press the Start button, and up comes the menu. But it was completely dead. That is, mousing over and clicking and things, nothing happened. And in several cases, I mean, I put up with it briefly. I could log out and log back in again, and it would bring it back to life. But it's like, okay, this is ridiculous. So I just backed out of Service Pack 3. And that was last week, and it hasn't happened again. So it does...

Leo: From a security standpoint, is there any negative to doing that? I mean, are you now going to be more vulnerable?

Steve: Well, it's not clear what's going on with Service Pack 3. When you use Windows Update, you still do all the pre-Service Pack 3 updates. And then it gives you a new version of the Windows Genuine Validation tool, and then that says oh, we recommend you install Service Pack 3. So it's like, oh, good, you know. And I have had people say that even that non-network install, where it says it's only going to be 66MB, actually downloads the whole thing.

Leo: Oh, that's interesting. So you still get a huge file download.

Steve: Yeah. Anyway, so what I'm going to do is - and I wanted to bring this up to our listeners because they may want to consider this, too. When this happened I did a little bit of Googling, and lots of people are having problems with Service Pack 3. Just weird sorts of things like I ran across one where somebody's Control Panel applet doesn't run after installing Service Pack 3. Took it out, and it's back again. So it's looking like Microsoft is having some problems with this service pack. And my feeling is let's give it a month, and they may be patching the service pack.

Leo: What's discouraging is this is a new machine that you're having trouble with. I mean, it's not like you have a lot of crap on it; right?

Steve: Right. It's brand new. No age on it at all. And again, I took out Service Pack 3, and now it's been behaving itself just fine.

Leo: Well, now we know what it is.

Steve: Yup. And speaking of the PayPal EV certificate, PayPal made a mistake which put them actually in a spotlight they didn't want, which was they had a cross-site scripting error on their EV SSL-enhanced...

Leo: Oh, no.

Steve: ...page. And so I thought it was sort of an interesting little gotcha because they put out a press release a couple weeks ago saying that non-EV-capable browsers were soon not going to be welcome at PayPal. And that generated a bunch of furor.

Leo: Oh, yeah.

Steve: So they sort of backpedaled from that a little bit, said well, no no no, we would just prefer that you used EV-enhanced browsers because they're working to promote this whole notion. Well, here they are with this green bar, this green URL, and this is on pages where there's a demonstrated cross-site scripting vulnerability. And essentially what this means is, by leveraging that, you can take somebody essentially to a different server that still shows the green PayPal URL, thanks to this cross-site scripting vulnerability.

Leo: Wow. Wow.

Steve: So the point is that it's important to understand that the fact that you've got the green bar, the green URL, the EV validation, says nothing about the way the site works. It just says, oh, we paid more for our certificate, and we were checked out more, and the certificate issuer...

Leo: Checked out more in terms of who you are, not in terms of your security.

Steve: Exactly. Exactly. So it's a bit of a black eye.

Leo: Yeah, no kidding. Especially since they were going to require...

Steve: Exactly. And then the other little interesting bit of news is there is now an unpatched vulnerability in Apple's current version of iCal, their calendaring program. Apple was informed of this by a security research group back in January. And they've been going back and forth and back and forth. And the security group has been getting frustrated with Apple not fixing this thing. One of them is a remote execution vulnerability such that if you were to open an iCal file by clicking on a link that you received in email, or if somehow this iCal file is displayed, there's a buffer overrun that allows a code exploit to be executed.

And so finally last week the security company said, okay, we're tired of waiting for you. Apple did say that it would be patched by the end of April and then didn't patch it. So this group went public with the exploits. Which has not made Apple happy. But, I mean, I can understand their position. It's like, okay, the longer this known hole stays open, the more opportunity there is for exploitation. I mean, this is what we've been putting up with over on the Windows side, of course, for quite a while.

Leo: I remember talking to w00w00, one of the security firms that does this. And they knew about a vulnerability in Windows that had been going on for a year. I think it's not unusual for these to go on for quite some time.

Steve: That's true. And in this case Apple was saying, look, we're going to get this patched, it'll be patched by the end of April. And that was after many, many prior excuses and delays and broken promises. So finally these guys just got fed up and said, okay, good luck. The only way to put pressure on you to really get this fixed is by going public. We tried to do it the right way, the responsible disclosure way. And we're tired of waiting.

Leo: Right. I'm sure there's a reason why Apple didn't put it out. Probably something wrong with the patch, right, or the patch caused more problems. This does put a lot of pressure on you to do something now because the hackers know about it.

Steve: Yup, exactly. Exactly. So I had a fun SpinRite story, another failure actually, since sort of we're on a roll doing SpinRite failures. A listener by the name of Alex Walters wrote, and he said, "SpinRite failed me, but it's NVIDIA's fault." And I thought, huh, what? And so he says, "Well, the subject of the email is somewhat deceptive. It's likely not the video chipset's fault. But I digress. A little while back, S.M.A.R.T." - the Self-Monitoring and Analysis Reporting Technology that are built into the drives - "started to give me the dreaded "Backup your data now" error." Now, he says, "I'm not a dumb person, but I hadn't backed up my data on that drive in some seven and a half years. I was quite interested in backing up that data."

Leo: That's a long time.

Steve: It's a long time. So he says, "I was quite interested in backing up that data. So I booted up and tried to copy the data that I wanted off onto a fresh 500GB external drive that I have had nearly as long as it's been since I last backed up. It all went fine and good till I got to my brother's basic combat training graduation photos, where Windows could not read the data. The horror," he says. "So I shut down and booted to my SpinRite floppy." He says, parens, "(I have a floppy drive, how novel)." And then he says, "I let it run and went to take a shower. No, not with the computer." He says, "When I came back, the computer was dead. The video card ate my machine." So then he says, "Cut to one month later. I have rebuilt my machine totally. I hooked up the old drive and ran SpinRite, and it clicked along until it had recovered everything. Well, everything that I wanted, at least. It's good to know that when everything in my machine wants to die, SpinRite can breathe enough life into my hard drive to safely recover. Thanks, Steve." Signed, Alex.

Leo: Happy news.

Steve: So not quite a "SpinRite failed me," but that's what his subject line said.

Leo: All right. Well, let's get into the meat of the matter. I have some very good questions for you from our listeners. Are you ready to answer a few?

Steve: You bet.

Leo: All right. We'll start with number one. Listener "Steve," in quotes, in Florida, notes we're not out of luck. He says: Steve, during last week's Security Now! episode, talking about XP's new Service Pack 3, you said, "But I don't know what happens if you push past that and effectively refuse to download Service Pack 3," which in fact we now know that Steve won't do. Last Patch Tuesday I hooked up to Windows Update. I was of course presented with the Service Pack 3 download option. However, there was some small print down below which said something to the effect that even if you choose not to install Service Pack 3 now, you may still need some updates. And there was a little button over on the right to click if you don't want to install SP3. You click that, you get the usual menu of patches and updates for Service Pack 2 XP as usual. Which I did. And everyone else can, and probably should. So in other words, you can get these critical updates even if you don't install Service Pack 3.

Steve: Yeah. I don't know what Microsoft's going to do about this.

Leo: Long-term it may not work; right?

Steve: Service Pack 2 gave you the option of not doing a backup. And I am really glad that Service Pack 3 forces you basically to keep all the uninstall files around. Of course it's a huge blob on your hard drive. But I needed that in order to back out of Service Pack 3. So now I'm feeling in retrospect like I haven't taken my own advice by waiting a while for the service packs, any new service pack to settle down before I just jump right into it because Service Pack 3 is causing people problems, and not just people with AMD processors. I know I've got regular genuine Intel, a quad core chip in there. And my Start Menu dies. I need my Start Menu.

Leo: You're referring to that well-known problem with AMD processors, which is even more of a showstopper.

Steve: Right.

Leo: Now, what's interesting is, as far as I can tell, Vista Service Pack 1 has not been a problem for most users. Don't know why not. But...

Steve: Yeah, Vista did something wacky on a machine that I have, too. I was using it to do some experiments last week and updated two - there were two updates that Microsoft Update installed. Now it won't shut down. It gives me the little spinning wheel and then goes to blue screen. It's like, oh, that's nice. It's like, fine. Okay.

Leo: Thanks a lot. Thanks a lot, Microsoft. Jawny G. in BC, Canada, mentions the Debian/Ubuntu Linux OpenSSL mistake. Which I'm not familiar with, but I'm sure I'm going to learn about right now. Steve, I realize this is a simple mistake that has more or less been corrected with a patch now. But since you were quoted in the article, I was curious as to what you think about all this. I look forward to your thoughts on the whole problem and possible problems with open source in the future. What happened?

Steve: Oh, Leo. First of all, although I chose this one note from Jawny G., many of our listeners said, hey, why aren't you talking about this horrible mistake in OpenSSL the Debian guys made? And many people wrote. So I thought, well, yeah, I mean, okay...

Leo: Let's talk about it, yeah.

Steve: Now is the time. We're in our Q&A episode. Okay, so get this. Back in September of 2006, so, what, year and a half ago, the Debian guys ran a program over their source code called Valgrind and another one called Purify. These are automated tools designed to check the security, sort of like an advanced form of Lint, which Lint is a tool that's been around forever that sort of, like, helps people with their C programming find things like uninitialized variables and sort of questionable syntax in code. So this thing purports to be, this Valgrind and Purify, sort of a higher level version of that.

Well, it finds a block of memory which has never been initialized. It's just this uninitialized block of memory. It looks around through the source, can't find anywhere that any code is initializing it to zeroes, which is typical. And so it flags this as something that needs to be fixed. So the programmers didn't really understand the code, so they made some changes which caused some errors. And then they ended up commenting out a critical aspect of the random number generator in the OpenSSL package in Debian in September of 2006. And this affects the offshoots from Debian like Ubuntu.

Leo: Oh, dear.

Steve: So, okay, get this. The OpenSSL generator, the random number generator is, as you would expect, a lot of thought's been given to it. That block of memory was uninitialized on purpose to be just some more random stuff that probably - now, see, you can't really count on it being random every time because, as we will remember from when we were talking about the freezing your RAM stuff, in general memory that you don't initialize tends to come up in the same state from one time to another. But that's probably going to be unique per machine. So it's just sort of - and believe me, that's not the only thing they were doing. They do things like microscopic timing analysis of the hard drive to get speed variations, the mouse movements, you know, all kinds of stuff is all fed into the pseudorandom number generator, which is the heart of OpenSSL, used to generate, for example, all of the security certificates that it produces. So unfortunately what these guys did is they commented, essentially, all of that out, so that the only thing left was taking the process ID. So since September of '06 Debian and Ubuntu and any other derivatives, there are a couple others, have had a seriously broken pseudorandom number generator in the OpenSSL package such that there were only - now, okay. Process IDs are 15 bits in Linux. So that's 32768. There were only 32K possible keys. And they've been brute-forced; they've been reverse engineered. And tools are beginning to surface. So...

Leo: It's using the process ID as a seed, or is it using it as a key?

Steve: Well, it uses it as the seed for the random number generator.

Leo: Oh, but if you know the seed, you can tell what the number that it's going to generate will be.

Steve: Well, and the developer's intention, the original coder's intention, was to use all of this stuff as the seed.

Leo: A variety of things, yeah.

Steve: All that other stuff, including this little patch of deliberately uninitialized RAM, just to throw some more entropy into the seed generator. So what happened was, as a consequence of this so-called let's find security bugs, this automated software that flagged the problem, the guys didn't really understand what the code was...

Leo: That was the consequence, by the way, somebody modifying code he didn't understand.

Steve: Yes. And commented out everything except using the process ID, which means that there's only 32K possible seeds for the pseudorandom number generator.

Leo: Whoops.

Steve: So it turns out that the bottom line is, the takeaway is, any certificates which you are depending upon, for example SSH certs, anyone who made a certificate pair using Debian and Ubuntu since September '06 when this happened, has a bogus, essentially a bogus certificate. And there are already hacks that are out there. There is a test that's available to see whether you've got one of the bad certificates. So I just wanted to, for those listeners who this affects, you probably know who you are, I mean, it's easy to find information about this. This is only about two weeks old. So of course it has been immediately fixed. So you're going to want to update your OpenSSL, recompile it or get the new package from Debian. And if you've generated any certificates in the last couple years for any purpose, you need to consider them as really exploitable.

Leo: Who discovered this, and how did they discover it?

Steve: I don't remember who - it was a security researcher. Luciano Bello is the guy who found it. And our good old friend H.D. Moore at Metasploit has got an extensive page on it and already writing toys to exploit this problem.

Leo: Wow. Wow. Now, Jawny G. says what are the possible problems with open source in the future. I think what he's saying is with open source you've got people potentially modifying code that they don't understand. I mean, I think there's a breakdown in the process here because you shouldn't be able to modify the pseudo number random generators, you know, it's a library, I would guess. You shouldn't be able to modify that.

Steve: Well, yeah. They clearly thought they were doing the right thing by running this security tool over their code. And what they did, of course, is they forked the software so that now they're off on their own, and the OpenSSL groups, I mean, there have been a number of other things that have been fixed in OpenSSL since then.

Leo: So the proper process would be to say, hey, we ran this on OpenSSL, and go to the OpenSSL group and say, hey, what's the deal?

Steve: Right.

Leo: Instead of forking the code - there was the mistake - and saying we're going to fix it.

Steve: Yeah, I did read somewhere there were some comments by the OpenSSL people. And they said, well, after we got through literally rolling around on the floor laughing at what had been done to our deliberately carefully created super high-quality pseudorandom number generator code, we would have explained that commenting that line out was a bad thing to do.

Leo: Yeah. So, okay, we've all learned our lesson here. I think in a way this could be, I guess, ascribed to a structural problem with open source. But I think in another way it's very clear that, if the proper procedures are followed, it isn't a problem. And the fact that the security researcher found it has something to do with the fact that it's open source. Had Ubuntu been a closed source project or Debian been a closed source project, this might happen and you'd never know because you wouldn't be able to see what's going on.

Steve: Yeah.

Leo: Interesting. Wow, really interesting. Dave in Grand Rapids, Michigan, he's worried about losing his YubiKey. You know, it's funny, I was just looking for my YubiKey, and I've misplaced it. He says: Steve, maybe I've misunderstood your description of the YubiKey, but it sounds like the only item needed for authentication is the YubiKey, and not even a username or password. So if a YubiKey is lost and found by someone else at your YubiKey-using corporation, the only thing they'd have to do to logon as you would be to plug it in and press the button. This seems like a huge flaw. Or does the user still need to type a username and then in the password field press the button, thus entering the one-time password?

Steve: There's been a lot of confusion that I guess I'm responsible for because I got so carried...

Leo: I don't think so. I remember you saying it right.

Steve: Well, I got so carried away with how cool this was, I - the confusion is that many people have said, oh, okay, I want one, now what do I do? That is, it's not...

Leo: That is your fault, by the way.

Steve: Yeah. It's not an end-user tool in the same way, for example, that the PayPal football is. Or I should say that the reason the PayPal football is an end-user tool is it's offered by PayPal who supports it. And so immediately upon getting it you've got something to do with it. The YubiKey is a cool technology. So it's sort of like it's more like a wholesale rather than a retail technology. So...

Leo: Now, I think you implied, though, that I could use it with Yubico as an OpenID key.

Steve: You absolutely can. So they've got an OpenID server. There's also an interesting site called MashedLife.com that is YubiKey enabled, and they're sort of a third-party password and website log-on provider. They've got sort of a clever technology that works with a YubiKey. So the point I wanted to make to Dave when he says, okay, you know, if I only need the YubiKey, what if somebody else gets it, was like, yes. It's not meant to be anything other than one additional very cool factor in a multifactor system.

Leo: Yeah, and that you were clear about, I think.

Steve: Yeah. So absolutely, someone could misdesign a system. And just because you're using a YubiKey doesn't mean that the system is going to be secure because everything else has to be secure. I mean, again, you absolutely want a something you know to be part of something you have so that, you know, there is a passphrase that says, okay, I am me, and you plug the key in, and it's like, and look what I have. So again it's - I believe this thing is going to take off, and there will be people who are using the YubiKey, I mean, I know from Stina, for example, that I think she told us during the podcast, that she's received queries from universities that want to give one of these to every single student and faculty member. And so it would be something that they use to log-on to terminals and things around campus. And so there these kids are going to get these and go, wow, that's interesting, never seen one of these before, not knowing how much more it is because essentially the IT department will provide all the backend infrastructure for it.

Leo: So I'm holding my YubiKey right now, for those of you watching.

Steve: I'm holding mine, too.

Leo: So this is something that Stina sent to us. But so can I use this with Yubico as an OpenID provider, and it would do everything, I mean, in other words, can they become a provider for me and so I can use it standalone? Or do I need to kind of do...

Steve: No, absolutely, they have an OpenID server. And you could use them as your OpenID authenticator.

Leo: And what would that look like? I would still have to enter my login and my password; right?

Steve: Yes. At any OpenID site you would be giving them your username and password. Okay. You'd give them whatever it is that site wants because...

Leo: Well, I'll give you an example. When I log into a site that uses OpenID, I click the OpenID link. They say, okay, what's your OpenID provider? You give them a web page that is an OpenID provider. So you'd give them Yubico.com. Then you're pulled to Yubico, where you give whatever Yubico requires. In this case I presume it would be a username, password, and the YubiKey.

Steve: Yes. I don't know. I have not looked at that.

Leo: Right, right.

Steve: But I know that they have an OpenID facility that is there and free and will always be free. So somebody with a YubiKey who was getting into OpenID could use Yubico's OpenID service.

Leo: Great, great. And presumably Yubico's implementing it as you would recommend, which is multifactor authentication - name, password, and then the YubiKey.

Steve: Maybe. But they wouldn't have to. I mean, all they're really saying is, I mean, are they saying this is you, or are they saying you have a YubiKey? Because the site that you're logging into could also require username and password...

Leo: Well, but that's the idea of OpenID is it bypasses that.

Steve: Okay.

Leo: So the whole point of OpenID, we use it - we can use it on TWiT.tv, for instance. You go to TWiT.tv and either provide the credentials we have given you, or say no no no, I want to use my OpenID credentials, at which point you're sent to the OpenID provider, who verifies your identity.

Steve: You're right, your entire identity, and then makes the assertion that...

Leo: I trust them.

Steve: ...this really is you.

Leo: Right. And then you get sent back to TWiT.tv, which says, okay, I know you're you. Do you want to set up an account now using this identity? In other words, should this identity be trusted. And so that's the process for OpenID. So, yeah, they would need to do some identity validation.

Steve: Right.

Leo: Kyle Hasegawa in Tokyo, Japan has a tip and a link for me. Steve, he says, this one's actually for Leo. Leo mentioned he was looking for a tool to test his server against attacks. I have more information on our hack attack, by the way. Fascinating story. I'll tell you about that in a second. I use these two Firefox add-ons for security testing, found them to be very good. They're from http://www.securitycompass.com/exploitme.shtml. The first is XSS-me, which brutally yet safely slams your site with a torrent of XSS attacks. I don't even know what those are.

Steve: Cross-site scripting.

Leo: Oh, cross-site scripting, okay. The second is SQL Inject-Me, which does the same thing except with SQL injections. After the test the tools provide in-depth reports about your site's security, at least for those two vulnerabilities. Cool, huh? I've run these against my Drupal 6.2 site as well as my own home brew websites, and I'm glad to say they all passed. Probably a good idea to limit testing to your own sites to avoid a visit from your local FBI agents.

Steve: Actually these are - we've talked about cross-site scripting and about SQL injection attacks. These are two tests that do a whole ton of, for example, cross-site scripting script exploits. So, I mean, it really hammers your page, the pages that you give it, with all kinds of attempts to perform SQL injection and cross-site scripting attacks.

Leo: Well, in my case it was neither.

Steve: Okay.

Leo: So I found out what happened. Are you curious? You want to know the story?

Steve: Yeah.

Leo: It took me a little while to figure out where the vulnerability was. And it was me. I was the vulnerability. I was the stupid person. So what happened, I guess the hacker sent me another note. By the way, he said, I'm not Brazilian. I don't speak English, but I'm not Brazilian. So we don't know where he's from. But he posted on my blog, he posted a comment saying you should harden your site, and if you want some help send me an email. And later sent me a note saying, well, this is how I got in. I had posted - you know you have a site, kind of a private site on Leoville.com, Steve, where we put Security Now! so you can download it.

Steve: Sure.

Leo: It's not hidden by anything by obscurity. It's almost like an open directory except there's some PHP code managing it. So you can go there, and you can see if the new files are there and download them. I have several of those. They're not websites, and they're not FTP exactly, they're just kind of file storage places for various people. I put my commercials up there for the radio shows and stuff. On one, and only one of them, and I enabled a feature that allows people to upload files. Oh, dumb.

Steve: Oooooh, yeah.

Leo: And it turns out that the folder it gets uploaded to, the incoming folder, I either didn't set permissions on or set them incorrectly to allow a scripting error to be executed. So all the hacker had to do was he uploaded a script, which I found. In fact, I have, and at some point I'd like to go through it, I want to parse through it on camera, and maybe you can be on when we do that. And it's PHP code, but go through it and show exactly what the script does. It's very interesting. It's not a very well written script, by the way. But it's an interesting script.

Steve: It did the job, I guess.

Leo: It did the job. Because what it does is it looks for other vulnerabilities on the server, looks for open directories, files it can download, particularly looking for configuration files. So apparently what it found was a directory with a configuration file it could download that had an SQL, my SQL password in it. And that's how he got in. And there were only two of the sites on that server, there are about five sites on that server, but only two of them were unprotected enough that he could zap them. The rest of them were hardened. So I've since found...

Steve: I think that's a pretty good hack, Leo.

Leo: Well, it would - yes, it's a great hack. And I think probably - not great. But I think what the guy did was he's going around, and probably has an automated tool, looking for open directories that he can upload to and execute. And he found one. Foolish of me to leave that there. I don't know what I was thinking. I certainly know that's a vulnerability. And but it did give me - and it's going to be fun. We'll go through this little C9 script that he wrote - or he got, probably, from elsewhere - and show how it works. So it'll be a good kind of extra security episode we'll do on TWiT Live at some point.

Steve: Yeah, cool.

Leo: Anyway, these tests would not have found that because it was too stupid of a vulnerability for these tests. John in Moncton, New Brunswick wants to know about the effectiveness of MAC address filtering. Not Macintosh, but MAC address filtering. My wireless network has the ability to restrict address by MAC address. How secure is this if I'm behind a NAT router? It would seem to me that as long as a MAC can't be hacked or faked, this is pretty secure. But could it really be that easy?

Steve: We've talked about this before. I know that some of our listeners are kind of rolling their eyes thinking, oh, Steve, why are we going to go over this again? The reason is it keeps coming up. And major tech support people keep telling...

Leo: Keep recommending it.

Steve: Exactly, keep recommending it as if, oh, just use MAC address filtering. Oh, it'll make you completely secure. It absolutely won't. So let me - I won't take too much time, but I want to explain, to John and any other listeners who might be wondering if that solves the problem, why it doesn't. The reason is that the MAC address is in the clear, even if you've got encryption working in your network. The MAC address has to be in the clear because it's sort of like the outer address for the packets for any endpoints on an Ethernet. Ethernet by definition uses 48-bit MAC addresses. And that's how the packet gets to where it's going on the network.

Leo: You should mention that every network device has a unique 48-bit address called a MAC address.

Steve: Right. And it's interesting, the way they're guaranteed to be unique is that that 48 bits is divided into two pieces, 24 bits each. One is a manufacturer serial number, and then the other is a manufacturer's serial number, meaning that each manufacturer gets its own unique 24-bit ID, and then they increment the other 24 bits so that together those are guaranteed to be unique 48 bits.

Leo: Unless they make more than 14 million cards. Or whatever that number is, I don't know.

Steve: And you can always get another manufacturer ID, so that's not a problem. And in the worst case, if you had a LAN that had identical MAC addresses, you would immediately get various sorts of alerts from your equipment saying that there's a MAC adapter collision. An adapter would say, wait a minute, there's somebody else on this network with the same MAC address as I, in the same way that, I mean, those of us who configure IPs manually have probably had IP collisions before, where you get a dialogue saying wait a minute, some other machine on the LAN has the same IP as I do. So it's just like, whoops, well, we can't use that card, that LAN card on this network because by some bizarre coincidence there's been a collision. And that can happen because users in more recent adapters are able to change the MAC address. They're able to set it to be whatever they want to. So it's not always hardwired into...

Leo: That's the key. He realizes, he says, it would seem to me that as long as the MAC can't be hacked or faked, this is pretty secure. True, but MACs can be spoofed.

Steve: Well, yeah, they can be spoofed that way. But the point is, in a wireless network, if you just turn on a sniffer, you're going to see all the MAC addresses of all the machines on the network. And so...

Leo: And then you can set your card to be one of those.

Steve: Exactly. It's trivial to spoof. So where MAC filtering is good is if you want to prevent inadvertent use of your network. So, for example, say that you, for whatever reason, you cannot secure your network. You can't use WEP or you can't use WPA. But no one really wants to use WEP because it can now be cracked in about a minute. So if for some reason you can't use WPA because, for example, you've got friends coming over all the time, or you've got some equipment that doesn't yet support WPA, the good WiFi encryption. It's like, okay, well, for some reason you have to have your network not encrypted. Well, the problem is that anybody within range will see your network listed and just connect to it. So the one thing - in fact, many people connect inadvertently. They just turn their computer on, it finds the strongest signal. Well, if you've got...

Leo: That looks good, yeah, I'll take that one.

Steve: If you've got a good, strong signal, people will be using your network without your knowledge or permission. So MAC address filtering is useful if you wanted to prevent that kind of sort of like soft - you wanted to put up a soft barrier to inadvertent use of your network. But anybody who knew what they were doing could easily use your network despite the fact that you're using MAC address filtering. So it's just not secure. WPA is the only solution, using a strong password.

Leo: And to save us another email, same thing with SSID hiding. Doesn't do a thing.

Steve: Right.

Leo: Because like the MAC address, the SSID is sent all the time in the clear.

Steve: Right.

Leo: You've got to encrypt. You've got to. Tom Terrific worries that if he goes to someone's PC, something might come to his. He says: Hi, Steve. I'm having more and more friends wanting me to work on their computer, so I thought it would be nice to be able to look at their computer remotely via GoToMyPC or something similar, so I wouldn't have to be driving all over the place or trying to diagnose over the phone. My question is, if their computer is filled with malware, viruses, et cetera, is there any way I could be infected by connecting to them remotely? Thanks, keep up the good work.

Steve: I thought that was a really interesting and good question.

Leo: I had never thought of that.

Steve: Now, okay. In a perfect world, that would be completely safe because...

Leo: You're not really running anything on your system. It's a window into their system; right?

Steve: Exactly. Essentially you're seeing their video, and you are taking over their mouse and keyboard. So it's purely a remote IO sort of deal. But we know it's not a perfect world. In fact, it's substantially less than perfect.

Leo: Oh, no. Oh, no. Tell me. Tell me.

Steve: Seeming less perfect every day.

Leo: Oh, no.

Steve: So if, for example, there were a vulnerability in whatever remote communications software you were using, and malware knew about that, it would be very possible for the malware to detect that you had connected using VNC, GoToMyPC, Remote Desktop, whatever application, and exploit a known problem in order to cause a buffer overrun at your end of the connection.

Leo: So anytime you're having a conversation with another computer, there's always that potential no matter what protocols you're using.

Steve: Yes. So what I would do if I were a person who was going to be sort of habitually connecting to probably infected remote machines, this is definitely somewhere you'd want to do that in a VM at your end.

Leo: Oh, good idea.

Steve: So you'd fire up a virtual machine session. You'd use that virtual session to connect with their machine, and that would probably, I mean, again, I'm maybe being overly cautious. In general it's not a problem. But again, it's not a problem until it becomes a problem.

Leo: Well, there have been problems. I remember with RPC, the Windows remote protocol security, man-in-the-middle attacks, things like that. So it's certainly something to be aware of. It's not like these things are invulnerable.

Steve: Correct. And so, again, it is something to be aware of. And I would say putting yourself in a virtual machine so that only the virtual machine might have a problem if something crawled back up the connection into your machine, that's really probably safe enough.

Leo: When I brought Auntie Dawn's computer in here last week - and she had complained about spyware. I said bring it up, we'll see if we can fix it. Before I connected it to my network, I made sure it was clean. Even though we run firewalls locally, you know, on all the machines, Windows firewall and the Mac firewall, I wasn't going to connect it to my LAN.

Steve: I'll tell you, Leo, I don't let any foreign machine ever touch my network. Just don't.

Leo: No. Well, I did eventually because I cleaned it up, basically reinstalled Windows, completely scanned it. Once I was fairly - again, there's no perfect answer. But once I was pretty sure I was clean I did put it on there just so I could update it.

Steve: I have a cable modem here that I have nothing else connected to.

Leo: That's a good idea.

Steve: And when anyone is over who wants to, like, check in, I mean, even my buddy Mark Thompson, AnalogX, he brought a laptop with him. And it's like, sorry. And he completely understood.

Leo: Oh, I'm sure he did.

Steve: I said I'm happy to let you use this cable modem that nothing else is connected to. But my own internal network is just sacrosanct.

Leo: Let me get this straight. You have a cable modem just for your guests?

Steve: Yeah.

Leo: That's taking it seriously. But we have more connections in here, and I could certainly dedicate one. I mean, there's one dedicated to Skype, it's nothing else. Of course, everything's on the LAN, though, because even the Skype machine has two Ethernet connections, one for the...

Steve: There you go, there's a bridge.

Leo: It has to be because that's how we can see the screen on the TriCaster. So, yeah, we're always at risk. I'll have to make a - and people are going to come in here and use our WiFi. So that's where - would it work, then, to have that triangular WiFi, you know, the three WiFi connection thing?

Steve: It's a three-router connection, yes. That's absolute security.

Leo: Okay. So maybe I'll have to end up doing that. Oscar Aguirre in Gardena, California wonders about URL file extensions: Hi, Steve. Thanks for the constant updates you provide for the security conscious. I eagerly listen every week - hi, Oscar, glad to have you - knowing I'll learn something new Steve and Leo will help me understand. Question: I'm currently house-hunting in Glendale, California. My realtor provided me with a printout with home listings. The printout had the following page at the bottom of the path - the following path at the bottom of the page. This is pretty common. When you print a web page, you'll get the URL of the page, and it ends...

Steve: Sort of down in the footer.

Leo: Yeah, in the footer. And it ends with mgrqispi.dll. I'd never seen a DLL file exposed to the user on a printout. I Googled that DLL file, realized that numerous hits were listing the same file and a scripts path. Should I be concerned about the home listing web server displaying paths to DLL files? Maybe I should inform my realtor or work with another realty group whose web server doesn't show their DLLs. Thanks again for the inspiration you provide for us security software folks.

Steve: I thought this was a great question. And it bears on - really for all of our listeners because we're seeing a proliferation of different file extensions on pages. Interestingly enough, if you use ShieldsUP!, you see "ne.dll." "NE" stands for Net Engine, and that's the thing I was talking about earlier in this show. It is the container for all of my Assembly language code. The front-end filter and the back-end extensions to the server is a DLL. We're seeing PHP. Many people have seen, for example, PL for Perl. The original was CGI. So you'd see something, blah blah blah, .cgi. So those have all been file extensions of either executable or interpreted code. So when you see .htm, html, shtml you saw before...

Leo: You may see .exe even, sometimes.

Steve: And sometimes you'll see an EXE. So if you see HTM or HTML, then those are actual text files. And for example, ASP now, Active Server Pages is one of Microsoft's technologies. So you can either have scripts which are interpreted by some executable code, or you can actually have the executable program itself, which is typically what you had in the original .cgi type of original web extensions. Anyway, the point is that many of these different file extensions are going to be seen in the future, more even than we see now. But there's nothing about a .dll or an EXE should put anyone off. It's just the way these people have implemented their Web 2.0 functionality.

Leo: Although, and this gets back to our conversation a little earlier, it's probably a good practice from the point of view of the web guy to use mod_rewrite or some sort of URL modification technology to hide that information. I mean, I don't think it's a great - it's not, you know, it's again security through obscurity. But it's not a bad idea to say, you know, not let them know what's running in the background there. If there is a flaw, for instance, in that DLL, they'd know how to attack it.

Steve: That's a very good point. For example, Oscar says he Googled that filename and found lots of instances where it was being used. So he was able to use Google to find all of the sites using that file. If a problem were found in that file - and unfortunately this is how search engines are being used now, in order to find other sites that can be exploited in the same fashion. So you're right.

Leo: Yeah. And in most servers that's a fairly easy thing to do, to hide those URLs.

Steve: Well, and for example in mine, when you go GRC.com/passwords...

Leo: You don't see anything.

Steve: Exactly, because it is aliased to an ne.dll URL that invokes the code to display the Perfect Passwords page, but users don't see it.

Leo: It's also just cleaner. The guy who invented - Tim Berners-Lee, the guy who invented the World Wide Web, said he'd never intended for URLs to be visible by the end users. You know, those are always machine readable. And he just never thought that people would actually have to type in http://. He just didn't expect that. It wasn't designed for humans.

Welles B. Goodrich - love that name, Welles B. Goodrich - in Santa Cruz, California, he wants Stealth with his NAT: Hi, Steve. Recently I purchased an AirPort Extreme 802.11 WiFi base station from Apple. After installation and configuration where I chose to enable the NAT firewall, I tested the security using ShieldsUP! To my dismay, the All Service Ports test indicated that there were ports closed but not stealth. No possible configuration could change that result. I've owned several routers with NAT firewalls built in. My computers, one PC and three Macs, a tiny LAN, have always tested stealth in the past. Do you want to explain what "stealth" and "closed" is, real quickly, or should I go on?

Steve: Let's finish first, then I'll go back.

Leo: I'm not particularly technically astute, but having my LAN hidden from the larger Internet universe always provided me with a sense of security. As I couldn't get a Stealth status using our grid scan, I stopped using the AirPort Extreme in favor of a D-Link Broadband Gigabit Gaming Router and an AirPort Express for my WiFi needs, including printing. So he basically has two routers. This arrangement has once more restored my stealth mode.

The question is, if the Apple AirPort Extreme truly includes a NAT firewall, why did the ports only read Closed? Was I overly paranoid to mistrust the AirPort Extreme's insecurity based on the results of the ShieldsUP! test?

Thank you for your work with Leo creating Security Now!. Since my discovery of the program about eight months ago I haven't missed an episode, even though I'm primarily a Mac user, and some of your subjects are sufficiently arcane that they may as well be spoken in Klingon. In spite of those limitations, I gain a great deal by listening. Oh, one more thing: SpinRite for the Mac, please.

Steve: Okay. So this is actually a little controversial. That is to say, there isn't universal agreement, especially among the old bearded UNIX guru folk, that stealth has any value. And you'll see this in postings, I mean, when I'm roaming around the 'Net looking at DSL reports or something, somebody will say, hey, I wasn't all stealth at GRC. And some curmudgeon, right, will say, ah, that's a crock, you don't, you know...

Leo: You created this notion, though, of open, closed, and stealth; right?

Steve: I coined the term "stealth" as far as I know. I don't know that there was any real concept of it before. So the fact is, anything that is stealth is technically breaking the rules. Any TCP/IP stack by definition should, for example, respond to a ping because that's what ping is for. It's for Internet engineers who need to ping things to verify that packets are able to get there and get back. So it's really useful to be able to ping.

Leo: Again, let's explain. A port is a connection between the outside world and your computer. You could think of it as a socket. You could think of it as a handshake. It's a highway, it's a path between your computer and the outside world. There are many ports, 65,000 and some ports that you can use. If a port has a service at the other end of it, let's say I have a web server running, and you say hello to that port, it'll say hello back, what would you like today? It'll say yeah. If there's nothing running on it, you have kind of two choices. That's open, by the way. If there's nothing running on it, it could be closed. It could just say, hey, I'm here, but I don't do anything at that port. Or it could say nothing. That's stealth, right, it just doesn't respond at all.

Steve: Exactly. The idea for TCP port, which is really the only one that will necessarily respond when you attempt to make a TCP connection, which is what, like, web and email and many of the most common services used, if you attempt to connect to a port where there is no service listening for connections, again by definition, by RFC formal regulations, that you should get a reset back saying this port is not open. I mean, you should be actively told that you have found a machine at this IP address, but it does not have that TCP port open. And similarly, if you ping using a protocol called ICMP, the ping you should get back should say, oh, there is a machine, or there isn't. So what stealthing is, is a breaking of the rules. It is a deliberate breaking, saying if somebody tries to connect to a closed port, we're going to just drop the packet. We're not going to affirmatively respond that, hi, we're here, but that door is closed. We're just going to - we're going to say nothing, as if the packet just went off into the, you know, yonder.

Leo: Why would I want to do that?

Steve: Well, because I mentioned before that it's very useful for Internet engineers who are wearing white hats. Unfortunately, it's also very useful for hackers who are wearing black hats.

Leo: They're looking for machines.

Steve: Exactly. It confirms that there is a machine available at that location.

Leo: Now, so that's important. Just because there's an IP address doesn't mean there's something at that IP address.

Steve: Well, because yes, there are four billion IP addresses. And you have no idea when you send stuff to a given IP address if it's going anywhere. Is it actually hitting a machine or not? If the machine obeys all the rules, any machine will respond and say, hey, you found me. But, you know, ping, and it'll respond with a ping, or it'll say, oh, that port's closed, try again. Guess again. Try, try...

Leo: Guess again.

Steve: I have 65,535 ports. Maybe one of them is open. You know, exactly, guess again. So I argue that, yes, I'm not saying that it's increasing your security. I'm saying - and you can argue that it's security by obscurity. And I'm saying sure. It's more secure to be hidden than not.

Leo: It's pretty good obscurity. If there's no response, there's no way a hacker could tell if there's anything there, right, unless he knocks on your door and says can I see what your IP address is.

Steve: There is no way. The packets hit, and they die. And that's why typically NAT routers are stealth. And in fact I'm - ShieldsUP! and GRC are significantly responsible for NAT routers being stealth because there were initially lots of holes where, like, NAT routers would respond with a ping, or they would say their ports are closed. And, you know, we popularized this idea that, hey, why not just be stealth because stealth is better.

Leo: Now, I just did ShieldsUP! on my system, and it's all green, all stealth, but I still failed because I respond to a ping. Why would I not want to respond to a ping? Same reason; right?

Steve: Well, yeah, because you're responding to a ping, therefore somebody knows there's something there. And so you could argue that it heightens a malicious person's interest. Or say they wanted to know - say that, I mean, it's giving away information. It's, you know, like for no reason. Like maybe they want to know if you're there, if your machine is on, when you're on vacation? Because then they can go over and break a window. I mean, the point is, better just to say nothing. Better just to be absolutely an enigma than to say, oh, good guess, you've got a live IP.

Leo: This is, by the way, the Comcast router default configuration. And I will go in there, and I will turn that off. How do you turn off ping?

Steve: It varies. There's no universal way. But normally it'll be like there'll be some WAN-side - WAN as opposed to LAN, WAN meaning Wide Area Network - there'll be some WAN configuration where you can typically say - it'll say, like, disable ICMP, WAN-side ICMP. Or it might say respond to pings or something. And the good news is, most routers now default off. Otherwise people complain because they go to Security Now!, and then they say, hey, I'm not stealth, I want to be stealth. And I say why not?

Leo: Why not? There's no reason.

Steve: Yeah. And so to answer Welles's question, there's nothing arguable insecure about the fact that Apple's AirPort Extreme, and this is a well-known characteristic of the Apple AirPort...

Leo: Which port is it? Is it the ident port that it turns off?

Steve: No. It's all of them. They all come up as closed as opposed - it just - it responds by the book. It responds to pings. It responds to attempts to open a port by saying this port is closed. So it could easily just say nothing. But for whatever reason, that's not what they chose.

Leo: And worse, they don't give you a way to change it.

Steve: Right. Which, you know, seems dumb.

Leo: That - at least you should offer that as an option for somebody...

Steve: Because people want it. And again, I'm not saying it makes you more secure. I'm just saying - and again, the old UNIX curmudgeons who say it's against the rules not to respond, it's like, okay, yeah, fine. So...

Leo: Well, that makes sense in a gentle, benign world where there were reasons why you might want to look at the topologies of the network or whatever. And I'm glad that Yahoo! and Google still allow pings because I use it to test to see if I'm up.

Steve: And within my own LAN network I'm pinging myself crazy all over the place.

Leo: Well, you ping me. You ping my servers to see if there's a new file for you. But actually it's a different kind of ping, but that's fine. I mean, if you're running a server you have open ports.

Steve: And in fact people are welcome to ping GRC if they want. I mean, I have deliberately let GRC be pingable so that people who are having a connection problem, couldn't bring up our pages, could just ping GRC.com to see if we're there. Because I'm not hiding.

Leo: Well, that's different. You're a server.

Steve: Exactly. We are, exactly, we're a big public server. Everyone knows where we are. I'm not an end-user. For example, you can't ping this famous cable modem of mine because it's just stealth. There's no reason to expose that.

Leo: Shouldn't be able to. Gary Warner, who is the director of research at UAB Computer Forensics - ooh, wow, we've got some biggies listening - cautions about certificate dangers: Steve, I thought you might enjoy seeing an example of one of the dangers that I'm concerned with regarding extended certificates. If we train users - we've talked a lot about extended certificates. We just talked earlier on the show about it. That's the green bar you get on Firefox and Internet Explorer 7 when you go to a site that has a special certificate that goes to extra lengths to identify them. He says there's a problem. He says: If we train users that, quote, "green bars are safe," we still haven't fixed a primary problem, that webmasters ignore security. You actually mentioned something that proved this with PayPal. My team at the University of Alabama at Birmingham looked at more than 15,000 phishing sites in the first quarter of 2008.

Steve: Okay, the fact that there are even 15,000 phishing sites, I mean, there's a problem right there.

Leo: Amazing. And they're pulled down the minute people discover them, so that just shows you how they proliferate. What started as a very occasional thing is now happening at least on a weekly basis: Phishing sites are running on sites with valid certificates. Most are on hacked servers where, for example, a web store is broken into and a phishing site is placed on their SSL-certified site. We're actually also seeing situations now where the criminals are buying certificates. Will the high cost of a certificate provide a disincentive to criminals registering an unrelated company, then using it for phishing? Absolutely not. They're using stolen funds to pay for it anyway. The certificate says, quote, "We have documentation that a company owning this site exists." It doesn't say, "The bank login content on this webpage corresponds to the company that registered this website." So they could have a certificate that says, yes, we're BadGuys.com, and have the site look like BankofAmerica.com. I just wanted to point out that making certificates harder to get and training users to trust them won't cure all of tomorrow's problems. Thanks for all you do. I've been a fan since your TechTalk columns way back in InfoWorld. I even bought your "Passion for Technology" books when they were published.

Steve: So I thought this was a perfect, a perfect illustration of what we were talking about before. It is important that we be clear about what it is that extended validation means. It means that the company that acquired the certificate was much more thoroughly vetted than just somebody getting a random SSL certificate. So that's what the certificate authority is vouching for. VeriSign issuing a certificate to Gibson Research Corp. has verified my identity, our corporate address, that our corporation is a company in good standing, blah blah blah, I mean, they're doing all this work to say that for me to get the green bar they've really checked us out. But it says nothing about my company's security practices. Am I encrypting all of my eCommerce data? Well, yes, I am. Every single node of my B-tree is encrypted. So I'm going over the top. But that's not what that certificate says. So it's important that people understand that all it is asserting is the identity of the company that obtained the certificate, nothing more.

Leo: And you know, I think most users are going to look at that green bar and say, oh, green means safe, and they mean safe, not green means I've identified this is a person. So I think he raises a very, very good point.

Steve: Actually, do you know that - remember, I think I made some sort of a mildly disparaging comment a few weeks ago about that Hacker Safe certificate that's appeared on all these sites all over the place? Get this. It was on the PayPal page that had the cross-site scripting vulnerability.

Leo: I don't even know what they're - who knows what they're testing for.

Steve: Oh, exactly. It's just bogus.

Leo: Yeah . Kyle Hasegawa is back. He has another question from Tokyo.

Steve: Good memory, Leo.

Leo: I don't forget names like Kyle Hasegawa. He has a quick fix for really dead hard drives: Hi, Steve and Leo. In Episode 145 you mentioned there are some hard drive problems even the mighty SpinRite cannot fix. Those, of course, being hardware failures. You also said that once a hardware failure occurs, you'll have to spend thousands of dollars to have data recovery professionals dismantle the drive and extract your data. Well, here's a cool trick that has worked for me, and it only costs you the price of another drive. I must warn you, this is a last resort and should only be done if software repair attempts are futile and you're not willing to spend thousands of dollars. What he's saying is by doing this you're making it impossible to fix if it doesn't work. The trick is to locate a drive that's exactly the same as yours. Matching the model number is imperative, but matching the production code is even better. Carefully remove the PCB - that's the, you know, the circuit board...

Steve: Printed Circuit Board, yup.

Leo: ...from your dead drive and replace it with the new drive's board. If your old hard drive's problem was the PCB board, you're good to go. If not, well, you're no worse than before, and you're out a few bucks. Thanks for the great show.

Steve: I thought that was worth mentioning...

Leo: That is worth it, yeah.

Steve: ...because it is a - it's a midpoint in between, you know, running SpinRite and hoping that SpinRite's going to be able to do the job, and forking over $2,500 to a DriveSavers group. And you mentioned something that I also know, and that is that these drive recovery companies, they have inventories of old drives just for this purpose.

Leo: They've got parts.

Steve: Yes, exactly, because they need, if it's a PCB that is fried, and just swapping out the printed circuit board with a good one will allow them to pull the data off, they're not giving you their PCB, they're just giving you the data. So they'll move an equivalent printed circuit board, the motherboard of the drive, onto the bottom carriage of the drive, suck all the data off, then put that PCB back into their inventory for the next time it happens.

Now, the fact is, when you consider all the complexity of the hardware in the drive, it is statistically far more likely to be a problem in the enclosure, that is, the heads and platters and so forth, especially because drives are often dropped, and the heads are bouncing around on the platters. So I would say certainly, if it's feasible for you to swap the PCB, do so. Especially many people will buy a couple copies of the same drive at the same time. So you may have a machine with the same printed circuit board because you got two drives at the same time, and one of them died. It's certainly worth giving that a try if you're comfortable with swapping printed circuit boards on the bottom of hard drives. My sense is it's probably not going to give you the leverage that you could count on that happening. But it's a really nice compromise between just using software and spending literally thousands of dollars.

Leo: It's a good point. I hadn't thought of that, Kyle, thank you. Yeah, very good idea. Do you have to desolder the board, the PCB, to replace it? Or is it just socketed in there?

Steve: I've never seen one that you had to desolder. Normally there's a flat cable coming around, and you can pull the flat cable off. Or very often now you'll see pins sticking up through a connector. So if you pull the board, sort of wiggle it back and forth but pull it directly away from the drive, it'll just unplug...

Leo: Oh, that's cool.

Steve: ...very cleanly because of course the reverse of that happens during manufacture.

Leo: Yeah, they want to be able to put it in easily.

Steve: Exactly.

Leo: Bob J. in Claremont, California has a really bad router: Hi, guys. Verizon recently changed my system to their Actiontec modem/router. It fails the ShieldsUP! test by responding to a ping - oh, that's funny, my Comcast does that, too - something my Linksys router never did. It also exposes its homepage, which shows my network setup to the world. Oh, boy. I typed in the public IP, and it pulls up the router's homepage with all the information. Granted, I'm using a strong password, but I hate this thing. I called Verizon, and their assistance was, well, useless. Any suggestions? Can I disable the NAT on the Verizon router and go back to using the Linksys? I've been listening since the beginning. Thanks for the great information. A lot of ISPs are doing this now. Comcast did it to us, too. The cable modem, or the DSL modem, is built into a router. So you have to use their router.

Steve: Yup. Okay, first thing. The fact that he typed in the public IP, and he got the router's homepage, that in itself does not guarantee that that homepage is available from the WAN side.

Leo: Oh, he needs to do it from home or somewhere else.

Steve: Yes, he's got to get a friend of his, somebody he trusts, while he's on the phone with him or while that IP is the same, to type it in from outside and see whether that page comes up. There is a characteristic of some routers, and not all routers, that if you - you are able to get to them, not only using the so-called "Gateway IP," which is your private IP inside the network, typically 192.168.0 or .1 something. But you can also sometimes get to the same, essentially the same thing using the router's public IP, but only from inside. So it may very well be, and I hope that it is the case, that from the outside it is not exposing its configuration page. And if it is, many routers give you the option, again, in a WAN configuration page, to disable what they will call "WAN Administration." You absolutely want to have WAN Administration disabled. You don't even want to expose a login page because someone could just sit there and pound on your login over and over, doing a brute force attack until they get in. And that's not something you want to allow to have happen.

Leo: Yeah, yeah. So disable WAN management. You certainly - it shouldn't pop up a page. If it does, you should have a password on the darn thing. So make sure you put a password, change the name of the router if it's a wireless router, all of those things are very important to do.

Steve: And also, while you're at it, tell it not to respond to pings, if that's a configuration option. Now, if none of that is possible, well, if at least it's not possible to tell not to respond to pings, you can still put your Linksys router behind it, that is, have it there out in front and then your Linksys router. And so you want to bolt that public face down as much as you can. If there's an admin page, tell it not to respond to ping. Use a username and password and so forth. But still you could plug it into your Linksys router and use your Linksys router. So then you've got all of the Linksys's security. And we know that that could be bolted down tight.

Leo: Excellent. Are you ready to have your bone picked?

Steve: Oh, yes. Pick away.

Leo: KakeMan in Malaysia says there is a potential problem in using one of those Linux boot disks that we've talked about. Because I like this idea. He says: On Security Now! Episode 142 you recommended the use of a Live Linux CD to access questionable websites or other risky activities. I'm not sure this is a good idea. It's true that I can use, for example, the fresh Hardy Heron Live CD, the Ubuntu that just came out for online banking. However, as time goes by - and this is a very good point - there will surely be security holes and vulnerabilities found in this Live CD. But by that time I might be vulnerable. Unless I can build an up-to-date Live CD, I wouldn't use it for that purpose. I might be wrong, but can you clear my doubt on that? Thank you.

Steve: Well, can you say "OpenSSL vulnerability"?

Leo: Perfect example. Perfect example.

Steve: Yup. Yup. As you said, that's the new Ubuntu. And unless it's as new as, you know, a week ago, it may very well have the vulnerable OpenSSL pseudorandom number generator that's been in place since September of '06.

Leo: Now, the point is, if you install it on a hard drive, it has automatic updates just like Windows or Mac, so it would be fixed. But a CD can't be updated.

Steve: Yup. So I think that when we were answering the question by our listener who was going to be using remote access to help other people with their machines, one of the things I was going to say, you know, I talked about running that in a VM. An alternative would be if you were using VNC that is multiplatform, you could use VNC on a Linux boot CD and again be very safe, as we said when we were talking about this before, as long as your drives were not available. Then, I mean, you're really working with belt-and-suspenders security there.

Leo: Right, right.

Steve: But so KakeMan's point is good. If your Live CD is old, that would be bad. So it probably makes sense to keep it fresh and not use it for a long period of time because at some point there will be known vulnerabilities there.

Leo: I guess the point is bad how. Certainly not bad in the sense that your disk could be modified, because it can't, and so you're safe in the sense that you're not going to have a virus or spyware get on there. But your network is vulnerable. And if you're vulnerable and you type a bank login at that moment, then they could be capturing what you're doing. So it's bad in that sense. Steve, we've run out of time and run out of questions. I think it's time to say goodbye.

Steve: Well, and isn't it nice that those two things happen at the same time.

Leo: Well, the nice thing about a podcast is we just go until we're done. So next week what is on tap? What are you planning for us?

Steve: I've got a whole bunch of topics. I'm going to have to choose one. So I don't know yet.

Leo: Oh, that'll be fun. People can participate, of course, by going to your website, GRC.com/securitynow. When you're there you'll see there are transcripts of every show. There is a 16KB version of every show, so if you need a smaller file, if you're bandwidth-impaired or whatever, you can get there. Steve puts great show notes up, so you can read along and click links and so forth. It's all there at GRC.com/securitynow. While you're there, take a look at ShieldsUP!, we were just talking about that, and many other free programs Steve offers to help you lock your system down, fix bugs, fix problems with Windows, even just simple utilities like Wizmo that are just fun to have. They're all free at GRC.com.

But don't forget his bread-and-butter, and everybody should own a copy, just if only to keep Steve doing this show, it's SpinRite. It is the ultimate hard drive maintenance utility, a recovery tool everybody'll want. It's just a really great program. I'm glad I have a copy in my toolkit. When Auntie Dawn came, of course the first thing I do is I SpinRite the drive. You'll find it all at GRC.com. Steve, thanks for joining us, and we'll see you next time on Security Now!.

Steve: Right-o.