SECURITY NOW!

Transcript of Episode #255

Listener Feedback #95

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-255.mp3
Quarter size (16 kbps) mp3 audio file URL: http://media.GRC.com/sn/sn-255-lq.mp3

Leo Laporte: This is Security Now! with Steve Gibson, Episode 255, recorded June 30, 2010: Your questions, Steve's answers #95.

It's time for Security Now!, the show that covers your security, your privacy online. Our guru, Mr. Steve Gibson, who comes down from his secure mountaintop lair every week to deliver to us the tablets. Steve is the expert at GRC.com, the author of SpinRite, the world's first hard drive - the world's best, not first, but best hard drive and maintenance utility. Might be close to the first.

Steve Gibson: Actually I was probably more like the last.

Leo: I like that. I like that.

Steve: I'm the survivor, though. Everybody else dried up and went away, and I'm still there.

Leo: The last man standing, yeah. He also coined the term "spyware," wrote the first antispyware program. His site, GRC.com, has lots of great free security stuff, free stuff of all kinds. Steve, good to see you again. You have your venti latte in there, or...

Steve: I do. I have my - this thing is a double-lined, vacuum-filled - I guess you really wouldn't have vacuum filling, but lack of air filling.

Leo: Airless. It's an airless...

Steve: It keeps the coffee hot like all day long. It's just fantastic.

Leo: Wow. And how many shots are in that bucket?

Steve: That's just two. I've scaled back, actually.

Leo: Wow. Any reason for that?

Steve: No, you know, if you're going to fill it with milk, then you really need some shots to sort of bring up the coffeeness. But this is just hot water with a couple shots of...

Leo: Oh, interesting.

Steve: Sort of like a fresh cup of coffee.

Leo: Yeah. I've been doing that a lot lately, just the filtered coffee. It's delicious.

Steve: Yeah, it is.

Leo: So today we have a Q&A session, which we do regularly. This is number - I should correct the lower third. I think this is our 95th Q&A.

Steve: Well, it is; but the significant fact is the episode number.

Leo: 255.

Steve: This being the most binary podcast that you produce, Leo, 255, which of course as all of our techie propeller-head listeners know, that's a full byte.

Leo: Hex FF. So are we going to roll over to zero now, or are going to - are we a 16-bit register?

Steve: No, we've got 16 bits, and so the carry comes out of the low byte into the next byte. And so we'll be at 256 next week. But today, 255, we've got a ton of - we have a little bit of updates, but important updates; a ton of really interesting security news; and of course some interesting questions and hopefully interesting answers.

Leo: Well, I always look forward to this. Sometimes this is the gloom-and-doom report. I'm looking at your show notes here. A few updates, but not too bad.

Steve: Well, and we went from - no, not too bad. We went from famine to feast, news-wise, because we've had a couple podcasts with very little going on. And this time there's just, like, we've got news coming out our ears. So lots of stuff to talk about.

Leo: Well, let's get the updates first because we always like - this is kind of, of late, what we've done is we've started with patches.

Steve: Yes. So two pieces of news. First is that Adobe, because of the severity of the Flash exploit, which they fixed a few weeks ago for the Flash player itself, the Flash plug-in that browsers normally invoke. As we know, that got moved. Essentially they early-released their seventh release candidate of version 10.1, which offers hardware, video acceleration, and some other enhancements, which actually some people have had problems with. They now offer 10.1 for - if you go to get Flash Player, that's the one you're given.

There is, and I meant to tell people this, if they're really having a problem with 10.1, there is a fixed version of the 10.0 development chain which you can update to if, for whatever reason, 10.1 has a problem on your system. And there have been reports of people who just can't get 10.1 going. So it looks like Adobe has a little more work to do on that front. This was an aggressive change they made from 10.0 to 10.1, offering basically lower CPU utilization by taking advantage of the graphics accelerators that are available. And so...

Leo: Boy, they needed that, too, because Flash is a pig.

Steve: Yes. Yes, yes, yes. So that was for the plug-in, but not for its presence where it was still vulnerable and where we had talked about addressing problems. In that blog-posting that I made a few weeks ago I instructed people how they could rename essentially the Flash that is built into Reader and/or Acrobat, which is different from the plug-in. Well, anyway, so the point was that that vulnerability still existed.

Now, Adobe famously stated that they were going to do a quarterly patch cycle. And so July 13 would have been the second Tuesday of July, which was their regularly scheduled next opportunity to update. But because of the severity of this, they pushed it out early. And in fact they pushed it out two days ago, on June 29. So some people have reported that their use of PDF Reader noted that there was an update and updated them. Others have said that they did not get an automatic update. So I wanted to let all of our listeners know.

What you can do either way, well, if you have been updated in the last day or two, you probably know it. And so you're fine. You will be taken to probably either 9.3.3, or I got actually taken to 8.2.3 because I'm still back on version 8, haven't moved up to version 9. So all of those various version threads have been updated. However, in my case it was necessary to open a PDF, and under the Help menu choose Check for Updates. And it said, oh, there is an update, what do you know. And it's like, well, it would have been nice if you'd tell me that. But it didn't do so proactively. So do make sure that you've got...

Leo: Apparently Acrobat does.

Steve: Okay.

Leo: You're talking about Firefox, I know. I'm just, I'm reading in the chatroom. I'm sorry, didn't mean to throw you off there.

Steve: Well, because I'm Acrobat.

Leo: But Reader doesn't.

Steve: But maybe v8 doesn't because I'm on v8. Anyway, some people have said they didn't get an update.

Leo: That's interesting.

Steve: Other people said they did. So just make sure that you're running or that you just, when you open a PDF, just do check for updates and make sure that it agrees that you're current. And that will...

Leo: That's weird.

Steve: That will give you the latest and greatest. And finally we can put this annoying, latest annoying Adobe problem to bed...

Leo: For a while.

Steve: ...two weeks earlier, yeah, until your - until our next podcast.

Leo: Dr. Mom says it was Acrobat and Reader 9. Reader didn't, but Acrobat did. So maybe 8 doesn't.

Steve: The other change was Firefox moved, has been moving actually very quickly forward; so quickly, in fact, that I missed a couple intermediate versions. I was at 3.6.3, and that was just not long ago. And now we're already at 3.6.6. What happened was at 3.6.4 they introduced some new technology, some enhanced crash protection such that, if you have Flash Player or - of course Adobe's Flash Player, Apple's QuickTime, or Microsoft's Silverlight, so any of the big three mega plug-ins - running on a page, and if they hang, there had been a problem that it could lock up the whole browser.

And so they, in going to 3.6.4, they fixed that. They've improved what they call "crash protection," allowing you just to reload that one page. And so it keeps the stability of the browser, even if one of these heavyweight plug-ins decides it doesn't want to behave. And then they quickly went from .4 to 5 and 6, which is where they are now, just making some additional tweaks and some security fixes. So Firefox users, we are now all at 3.6.6. And the v4 beta candidate has begun to float around a little bit. It's actually a candidate release for the beta. And it's like, okay, well, I guess they're tiptoeing into beta.

Leo: Beta for beta.

Steve: Exactly. They're tiptoeing into this one carefully. This is a significant - this move to 4, Firefox 4, is going to be significant. They've - a major change to the, as we call it, the browser chrome, that is, the window dressing. Tabs get moved up at the top, the way they are in Chrome, in Google's Chrome browser, and in Opera. So they're making that change. Which I think will be nice. It'll be nice to see.

And then the big deal is they're beginning to address this evolution toward web-based applications, like, for example, Google Calendar. The idea would be that, with Firefox 4, you'll be able to break windows, or like we might call them "tabs," but you'll be able to break them off so they look like standalone apps. So like Google's Calendar would be running in a window, looking not like a browser, no browser navigation, no forward-backward button, no menu, no shortcuts or all that other stuff that we're used to, it would just look like an app. But it would actually be a web-based, web-hosted app running in sort of this freestanding window with, like, better desktop integration, so you could have an icon on your desktop that launches Google Calendar without ever going through what looks like a browser.

So, and then there's a whole bunch of cloud stuff for, like, shortcut and tab sharing in the cloud between multiple instances of browsers running on different machines, all that kind of stuff. So lots of new features coming in 4. And hopefully not lots of new security problems. Though we know that's always a problem with anything new. So that's our updates.

Leo: There you go. Security news, Steven?

Steve: Well, there was a lot of controversy this week about whether Congress had given the President an Internet kill switch.

Leo: Yeah, I saw that. I'd love to know what you think of that.

Steve: Well, first of all, it's not possible. The Internet isn't somewhere. It's not like in Virginia or somewhere. It's inherently distributed. So, I mean, and it's, like, distributed fabulously to make it so robust and resilient against accidental problems. So, first of all, that is not what the so-called "Protecting Cyberspace as a National Asset Act" does. The only thing, and I looked at it carefully because I was curious what it was that legislatively had just happened, the only thing I could really see was that there were some formalization of presidential authority to ask - or, you know, compel - private Internet providers to do the right thing, sort of the stuff they would probably do anyway.

Like, I mean, if there were some sort of cyber attack, you'd expect the major backbone people, from a technical standpoint, if it was the right thing to do - and I don't even know what that means because it would be a function of what the nature of the cyber attack was. But, for example, if there was some major denial of service flood pouring into the country, and we weren't able to block it effectively, well, the individual inlets could simply pull their transatlantic plugs and just say, okay, we're just going to deny all incoming traffic into the U.S.

Now, that's not something that the White House can flip a switch and have happen because these are all privately owned and privately run. But the President did get sort of the formal power to formally ask private Internet backbone providers to do the right thing. So that's really all it was. I mean, I suppose, technically, if you really needed to, you could wire up some sort of master switch. But what it would have to do would be somehow literally insert itself between every incoming feed into the nation.

And then of course the problem is how you keep that from being flipped inadvertently. Then, of course, that becomes a huge hacker target. It's like, oh, let's go get control of the Internet kill switch and take the U.S. off the Internet. So anyway, it just - it's very difficult and impractical technically, and that isn't what this Protecting Cyberspace as a National Asset Act did anyway. So it was sort of some early misreporting of it, and some overactive journalism, I think.

Leo: Well, "kill switch" is such a good word in a headline.

Steve: The hot-button word, yes, yes. Meanwhile, many of our listeners asked if I was going to talk about something else that happened last week. Actually it was on June 25th, the White House DHS, the Department of Homeland Security blog introduced the concept of a national strategy for trusted identities in cyberspace. Now, just the idea of the government getting involved in something this important, the idea being that the government is beginning to say, you know, impersonation and identity theft and spoofing and all these things are problems, and they need solutions. So what they produced is a very substantial document, I think it's 40 pages long, which is the so-called "National Strategy for Trusted Identities in Cyberspace." And we're going to give it a podcast because I can't do it justice here.

Many people, what I see more than anything in the industry's reaction is, oh, no, don't let Big Brother, don't let the government get involved in this. And my feeling is, yes, I mean, I understand the dangers. But it is a direction we need to think about. My feeling is that it's a good thing it didn't happen 10 years ago because we weren't ready. I don't mean the government wasn't ready. The government will never be ready. The government can't do this. This has to be, it'd be like the government - it'd be like the first WiFi encryption, WEP as opposed to WPA. The professionals got involved finally with second-generation wireless encryption and got it right.

Well, my feeling is that, at this point in time, the security and the identity and just in general the security community understands the problem, both the good and the bad. We've talked about on this show through these last five years many of the problems with identification. I mean, for example, you might say, oh, biometrics is the answer. But then the problem, of course, is, well, what if you get your fingerprints stolen. You can't change your fingerprint, so that's a problem. And do you want really your fingertip to become of very high value, in which case maybe someone wants to go cut it off in order to borrow it and do something wrong with it.

So, I mean, there's many sides to this issue, and it'll make for a great podcast to discuss what this "National Strategy for Trusted Identities in Cyberspace" document and proposal is, coming from the government. So, yes, I'm aware of it. And my sense is, well, it's a useful dialogue. I think I don't see the White House looking to, or even DHS, to impose anything. I think they recognize it's really complicated, and ultimately the solution will come from the industry. We want to make sure that - all of us in the industry want to make sure that it's done right. And it'll be an open process. So I'm cautiously optimistic and interested. And lord knows it's a fantastic topic for the podcast. So I have a feeling the next five years of Security Now! will be touching on this as it lumbers forward at no doubt glacial speed.

Leo: Slowly.

Steve: Yes. I also wanted to note, again in recent news, that this accursed launch option in PDFs is ramping up still. What annoys me is Adobe has done nothing, and we're now at T plus three months and counting. Remember we talked about this at the end of March. A guy by the name of Didier Stevens revealed a means for causing PDF files to run executable content, which they provided. We've talked about it a number of times because this problem is not going away. The good news is you can disable, I mean, our listeners are probably safe because I have pounded it into everybody that you can open up preferences in whatever you use to read PDFs.

And this is not only an Adobe problem. It's an Adobe format problem because for some reason they thought it would be great to have a launch option in a document which allows it to run, like, by design to launch code that the document contains. It can be turned off by using the so-called Trust Manager menu items under Preferences, and then turn off the "Allow opening of non-PDF file attachments with external applications." You don't want that on. No one wants that on. No one probably ever wanted that on. But it's on by default. And Adobe is still thinking about what they're going to do about this.

Meanwhile, what's happened is targeted attacks are occurring using this at an ever-increasing rate. There are variants of the worms we've talked about before, the Auraax and the Emold worms, which drop a rootkit onto infected machines. And then they attempt to copy themselves to all attached drives, which will then use the Autorun tactic that we've talked about in order to reinfect those machines when you move an attached drive to some other machine. What's arriving for our listeners to sort of be aware of is email, targeted email, that appears to come from the company's system administrators, telling them that they need to update their email settings.

And again, some of this is not new. But unfortunately it's still being very effective. And it's getting a lot of these rootkits installed. The subject of the email is "Setting for your mailbox are changed." So not quite English properly formatted, which is the first tipoff, of course, as is often the case. And then inside it says "SMTP and POP3 servers for mailbox are changed. Please carefully read the attached instructions for updated settings."

And of course that short email then contains the PDF that is the malicious payload, which only can get a grip on your machine if you still have this launching option enabled, which hopefully none of our listeners by this point do because it's been three months, and we've touched on this several times. I bring it up again because it is ramping up. And, for example, there was one of the news reports was a major publication, it was IDG that published, I think it was Computerworld, and IDG staffers have reported receiving a lot of this. So, I mean, it's really being targeted. And unfortunately it's being effective. For a while a couple of weeks ago the Zeus botnet had taken up trying to use this, also. So, I mean, this is just - it's a current serious issue in the security world. So don't get caught by it.

We've talked a little bit about how, just a few weeks ago, Google still shows it in beta, how Google has allowed secure SSL connections for searches, which a lot of people like because we know that, if you use https://google.com, or www.google.com, to get to Google search, you've established an encrypted, authenticated tunnel, a connection between you and Google's engines, and that nothing along the way is able to see what you search for, nor see what Google returns.

Well, everybody's happy with that except many school systems, which depend upon their access filtering to protect students or limit students from getting objectionable content of various sorts. So the problem was that Google had many other services that were fine with using encryption. But when Google.com's search was using encryption, now there was a problem, and educators were forced to block all of Google's security rather than just search security. And that turned out to be a problem because, even within school administration, many people were taking advantage of the Google cloud services that were secured and securable, and so they wanted to be able to use SSL. The problem was they just didn't want search.

So what Google did was split off some different IPs just for secure search, to make them, to make search individually blockable over SSL. So what now exists is something called - it's actually an alias for a different domain. It's called encrypted.google.com. And that's an alias for a different domain, www3.l.google.com. Well, that's got different IPs than regular Google.com. Regular Google.com is 66.102.7.99 and 7.104. This funky domain www3.l.google.com, it's got 7.100 and 7.101.

So what happens is, if you attempt to go to Google.com over an SSL connection, you're actually, your browser is - you do get an SSL connection, but you are immediately redirected from those normal Google.com IPs to the second set of IPs. And that is currently the only way to access search. So what this enables is people who want to filter search, or block search because they can't filter it, you're still going to get a secure connection if you attempt to go over SSL. But that allows them to block the connection to those IPs at port 443 over at the Google side, which we know is SSL, which prevents, for example, students within this protected environment from being able to get to Google securely. They have to do Google searches over non-SSL, which will leave them on the original IPs, which then allows the filtering and web monitoring software to do what it wants to do.

So that was an interesting, I think it's - certainly it's not something that Google anticipated. It's probably why SSL still has a little beta flag on it. And they're working out the bugs of allowing secure search. This is a way, again, of essentially allowing filters to block some secure access, which is the only way to get to search, but not block all of Google's security because it turns out people do want to be able to, like, look at their calendar securely and do other administrative things using the Google cloud stuff.

Leo: I wonder how much of this has to do with China.

Steve: Good question. That's been active, of course. Google is going to stop...

Leo: They backed off on this, so...

Steve: Yes, yes.

Leo: And they're going to - I don't know exactly what they're going to do. They're going to offer - they're not going to forward to Hong Kong anymore.

Steve: Yeah, but they're - and they're saying they're going to, like, have a link so that...

Leo: You can get unfiltered results, but it won't be - you'll have to click a button.

Steve: Yeah. And so no one is really sure because their contract is expiring with China, and so they're hoping to get renewed here in the next couple days. Be interesting to see how that pans out.

Leo: It's obviously like, well, we'll give you this much, and then they're waiting to see what China says. But I wonder if some of this filtering issue and this SSL issue doesn't have to do with schools and have more to do with governments that would like to keep an eye on what people are searching for.

Steve: It could very well be. Good point. They are billing it as, of course, schools.

Leo: Schools, yeah.

Steve: But who knows?

Leo: It's less sensitive.

Steve: And then I got a kick out of this. Our listeners will, too. The FBI - and this is a mixed blessing. I want to make sure I'm not giddy about this. I don't like, I mean, we've talked about how encryption in general is a powerful tool that is a double-edged sword. We just got through talking about how valuable encryption can be. People want to be able not to be spied on. They want to know that they've got some privacy. Certainly we know we use encryption with SSL connection for banking and for being able to, like, really enable security on the 'Net. The problem is, it's really good. Encryption today is really good. And so not only can it be used for good purposes, but it can be used for questionable, shady...

Leo: This is always - I remember talking to Phil Zimmerman about PGP. And this is always the thing people say, well, you can't have encryption because the terrorists will use it. And it's tricky.

Steve: Yes, it is. So what happened was, in Brazil, someone was under investigation, Daniel Dantas, under investigation for money laundering in Brazil. He had five hard drives which he had encrypted with TrueCrypt, which we've talked about often, a fantastic whole-drive encryption tool, or even a partial drive encryption tool...

Leo: Love it, yeah.

Steve: ...if you just want to encrypt a folder. The Brazilian authorities tried for five months to - whatever it was they did, we don't really know what they were doing. But they tried for five months to get access to his drives. What's interesting is that under Brazilian law the police do not have the right to force this guy to reveal his passwords. So after five months they asked the U.S. FBI to please see if the FBI could gain access to the contents of these drives. And after a year the FBI gave them back, never having succeeded.

Leo: Now, we're sure of that; right? They didn't get into it and didn't decide, well, we don't want anybody to know that TrueCrypt...

Steve: What we believe is, and it's entirely believable because the various stories have gone into some detail, the FBI has something for TrueCrypt called Dictionary. And so we know what that means. We know that any good password-based encryption, if there are no other known vulnerabilities, the single glitch, the single vulnerability is password guessing. You just brute force try to guess the password. And so the FBI has in their arsenal of tools some sort of a something called Dictionary. And for a year they had these five drives spinning, pounding on them and on TrueCrypt, just trying to guess the password using their Dictionary.

Leo: Wow.

Steve: Now, Daniel Dantas, this guy who is under investigation, clearly didn't use any of those passwords. He came up with something unique. And that's all it took. So I'm not, I mean, I don't know anything about him, whether he's guilty or not of laundering, I mean, this is the double-edged sword. But it is a lesson to our listeners because we've talked about this often. If you use a password that is not in the dictionary, that is sufficiently long and has a lot of entropy, a lot of randomness in it, then all other things being equal, if there aren't other backdoors or other trapdoors or failings in the cryptography, and we know that TrueCrypt has been very carefully and beautifully designed over time and has been evolving, then there's no way in.

And it's funny because the actual, one of the news reports I read, I got a kick out of it, said, literally, "Under Brazilian law the police do not have the right to force either Dantas or TrueCrypt's makers to reveal the passwords used to protect the hard drives." Well, TrueCrypt's makers are absolutely powerless to help. I mean, they designed it as a robust, high-quality crypto system such that there's nothing they can do. I mean, anyone can examine TrueCrypt, which is open source, and see what the technology is. And absolutely no one on the planet, given a year of pounding, we don't know how much, how long it would take brute-forcing all possible passwords. But if the password is long, TrueCrypt's technology is such that Dantas won't be worrying about having his drives looked at anytime soon.

Leo: So, now, I wonder if this does say, though, that TrueCrypt is impermeable. It merely means that a brute force attack can't be used against it; right? I mean, that's all they did.

Steve: Right. So it's - I guess the way to say it is to be very careful with terminology and to say, everything else notwithstanding, that is, we don't know there isn't a bug in TrueCrypt. But we know that very good people have deliberately designed it using all of the state-of-the-art knowledge of how to do this correctly, very high-quality random number generation, the best ciphers, advising people who use it about the nature of the password, that that's the vulnerability, so choose a really good password, and we know what that "really good password" phrase now means.

So, yeah, we can't ever state that a crypto system is invulnerable or perfect. I mean, there are people who are kind of chiseling away at AES right now. And they've not made great progress. But reduced-strength versions of AES are, eh, they're beginning to sort of understand what AES does. It's many years old now. And so they're kind of, they're sniffing around the edges. But still the formal high-strength AES, the one that's part of the standard that anyone would use, is absolutely bulletproof, so far as we know. I mean, yes, a breakthrough could happen tomorrow in deciphering technology. Seems really unlikely, but it could happen. But this was just sort of an interesting case in point where someone who chose, clearly chose a good password was protected. And unfortunately, I mean, we don't know what's on the hard drives. Nobody ever will unless Daniel Dantas reveals his password.

Leo: Okay.

Steve: And then my last little bit of news is that Google's Chrome browser has moved ahead of Safari to take the #3 spot in total access, browser-based access to the 'Net. Which I thought was interesting. I'm surprised by that. But just a little, I mean, we know that IE is in strong first place. Firefox is in strong second place. So the two of those, #1 and #2 browsers, pretty much are soaking up all the oxygen on the 'Net. There's not much left over. It's a single-digit percentage. But Chrome's a little bit ahead of Safari, which happened last week.

Leo: Yeah. I use Chrome religiously now. I know you're a Firefox fan. But, boy, I love Chrome. It's fast. It's got extensions. It's really just great.

Steve: It's been well designed. And in my one little bit of errata, just totally, not completely out from left field, but ICANN has finally decided to approve a .xxx top-level domain...

Leo: That's not errata, that's erratica.

Steve: ...[laughing] for adult content. And it's been controversial because people take all kinds of different views of this. And I'm not saying it's a good thing or not. I'm just reporting that this is a little bit of news. Because people are saying, well, that doesn't mean that the porn sites are going to give up their dotcom domains. They'll just grab .xxx as well. It's like, okay, well, that's probably true. But it's been essentially the arguments for them not allowing a .xxx top-level domain ended up falling short. And it was shown that it was just sort of irrational bias against having it, and that irrational bias wasn't a good enough reason not to allow it. So why can't we have it? And they said, okay, yeah, fine.

Leo: Yes, ICANN.

Steve: And actually the - yes. The registrar who will be managing this and who's been pushing for it is going to charge, I think it's $60 per year for domains in the .xxx top-level domain.

Leo: And do you have to prove that you're xxx? I mean, could somebody register Google.xxx?

Steve: I don't know. But he will be donating...

Leo: I should get TWiT.xxx.

Steve: He will be donating a non-inconsequential, that is, a consequential amount of money to child protection charities.

Leo: Good, good.

Steve: Which I thought was neat. So a chunk of those domain registrations will go to sort of work against some of the ickiness of that side of the Internet. So that's cool.

Leo: Yeah. Yeah.

Steve: And I had, in keeping with today's Q&A theme, a note from Troy Haskin in Madison, Wisconsin. He was wondering about a SpinRite tip jar.

Leo: Yeah.

Steve: He said, "Steve, I've recently listened to your Q&A discussing the ins and outs of SpinRite's personal license concerning use by friends, family, et cetera. I must admit that about a month ago I used SpinRite to save my roommate's computer from a bad sector. She was about 20 pages into her master's thesis and wasn't one backup. I made her a huge fan of Live Mesh after this. So I threw SpinRite in, and all was well in 36 hours. Nothing special, I know from years of listening."

He said, "Anyway, I did feel at the time that I was slightly overextending the wonderful freedom of the product, but really wanted to help a friend out. And after listening to you and Leo discuss this topic, it occurred to me, why not try a 'Tip Steve' jar like the 'Tip Leo' jar recently instituted for TWiT. Then, the next time I have to save another friend, which will happen, I can tell them to go and give as they feel."

Leo: That's a good idea.

Steve: "I think this would be a nice and nonintrusive way of allowing people to thank you for your wonderful software. Thanks for reading, and thanks for the years of excellent content." Well, I don't know. I think that sort of formalizes breaching our license agreement, and I don't feel comfortable doing that. I think I would, I mean, I look at the value people are receiving from SpinRite. And while SpinRite's not cheap, it does deliver. And people's time is worth something, as is mine.

So, I mean, I would - I think GRC is probably better off encouraging people to purchase SpinRite if they want to use it on their own machines. I'm going to turn the blind eye toward people who help friends. But I would rather encourage people to get their own copy of SpinRite than sort of turn this into a pay-it-as-you-go sort of basis. I'm uncomfortable with the tip jar notion. So I just wanted to share that in case it had been something other people had thought about.

Leo: And we haven't heard a yabba-dabba-doo in a while.

Steve: Well, I have it muted during the podcast.

Leo: Yes. For those who haven't heard earlier shows, that's the sound that - Steve has sounds for every network event. And the sound when a credit card goes through, a purchase goes through of SpinRite, is "Yabba Dabba Doo." Which happens probably fairly frequently.

Steve: Always makes me smile.

Leo: I bet it does. Are you ready, Steve?

Steve: Let's go.

Leo: We've got questions. We've got about half an hour to answer them. So we're going to do as many as we can in 30 minutes.

Steve: Sounds good.

Leo: Starting with Timothy Hahn in Maryland and many other listeners who are concerned by recent stories about invalid SSL certificates. Tim starts the ball rolling by writing: Steve, eSecurityPlanet and Slashdot and many others have front page articles today saying, well, we have about 22 million SSL servers with certificates that are completely invalid because they don't match the domain name on which they reside, meaning only about 3 percent of SSL certs in use are actually valid. What? Naturally, I and many others are concerned by this, and I knew of only one place where I could get it explained by someone who understood what was going on. And we both know where that is. You, Mr. G. What's the deal?

Steve: Well, this really was - I don't know how to explain what happened here because a bunch of news outlets picked up this story, which is completely bogus. And it's from, I mean, a reputable security firm, Qualys. And, for example, reading, just so you understand what the background is here, reading one of the stories from June 29, which is just a day ago, new research conducted by security firm Qualys has revealed that only 3.17 - it's good to have accurate numbers on these bogus stories, by the way.

Leo: Yeah.

Steve: 3.17...

Leo: It makes people believe it.

Steve: Exactly - percent of secure websites have valid secure socket layer, SSL, certificates. Okay, well, right off the bat you know it's like, wait a minute. What? The company said that it had scanned 119 million domain names, of which only 92 million were active. More than 12.4 million domains had resolving issues, and 14.6 million failed to respond. Of the remaining 92 million active domains, 34 million domains used both port 80, typically used for HTTP, and port 443, which is used for websites with the prefix https://, those secured using SSL.

Ivan Ristic, director of engineering at Qualys, said that by taking a closer look at those sites that used port 443, the firm discovered that only 23 million were actually using SSL. However, Ristic said that less than a million, only 3.17 percent, of the domain names matched. That means that 22 million SSL servers have certificates that are completely invalid because they do not match the domain name on which they reside. Ristic said, "For us, the question is, how exactly is SSL used on the Internet as a whole? Interestingly enough, as popular as SSL is, no one had made public the information about how it was used."

So I read several versions of this report. Because, I mean, everyone was in a panic, and I was getting tweeted, and people were sending emails, oh my god, only 3.17 percent of SSL certificates are valid. It's like, no. No. What - and I don't know why they did this, or who authorized...

Leo: It's just scurrilous.

Steve: Well, who authorized the release of this news release from a reputable security firm, because Qualys is. Apparently what this guy did was get all the domain names there are. And I think he used, I saw somewhere, .com, .net, .org, .edu, .gov - so, like, the main top-level domains, TLDs - and recursed through all the domain names, did a lookup of their IP, then connected to that IP and checked to see whether port 80 and port 443 were accepting connections; and, if 443 was accepting a connection, initiated an SSL connection to obtain the SSL certificate for port 443 at that IP. And then was upset if the certificate name was different than the domain name they had used to look up the IP to get to the port to get to the certificate. Okay. Which is - which tells us nothing.

Leo: Right.

Steve: For example, I have www.grc.com. It ends in .202 - 4.79.142.202. It's www.grc.com's IP address. And I have a valid certificate. And everyone who uses HTTPS can connect to me with no problems at all. Well, I also have SpinRite.com. I own that domain. But I don't have a big separate website there. It points to the same IP. So that if someone goes www.spinrite.com, they get to GRC.

So using Ivan Ristic's logic, my certificate is invalid because he would have taken www.spinrite.com, looked up the IP, which is shared with GRC.com, same IP. And he would have received GRC.com's SSL certificate and said [gasping], they don't match. Well, yeah, they don't. Who cares? You cannot use https://spinrite.com because I don't have a separate SpinRite.com SSL certificate living on its own IP address. Multiple domains share a single IP address. Well, where have we ever heard that before? That's called "shared hosting." Which is hugely popular.

Leo: Right. Everybody does that.

Steve: Yes, which is the hole this guy fell down...

Leo: Oh, please.

Steve: ...and doesn't seem to have realized it.

Leo: Moron.

Steve: So anyway, everybody can breathe a sigh of relief. This doesn't mean anything. This is ridiculous. I don't, again, it grabbed some headlines, and everyone was panting over at Slashdot. It's like, well, but they do that a lot there, Commander Taco and his crew.

Leo: Well, I mean, normally I love Slashdot. But I think a little bit of an uncritical eye here on this one.

Steve: Yeah. So people need to recognize, I mean, I do, too, Slashdot, for what it's worth, I mean, I think they do - there's lots of interesting stuff pops up there and gets discussed. So certain...

Leo: Oh, yeah. But it's - this is actually a good time to say that we more and more have to use our thinking caps when we read something. Just because you read something online doesn't make it true, even if it's in a trusted source. And you just have to think about it, does it pass the sniff test? Or write to Steve.

Steve: And you have to, again, to understand the details you need to understand exactly how the Internet works.

Leo: Right. So that may be the problem.

Steve: And so that's what we just went through was I explained to our listeners where this number came from and what it means and why it's nothing to worry about because the fact is, if anyone attempted to access a website whose certificate did not match, they get all kinds of warnings. I mean, if you, well, for example, anyone could try it. Go, https://spinrite.com. You will be warned that there's a security certificate problem because you will have tried to get to SpinRite.com. You will have received a certificate from www.grc.com. Those names don't match, and your browser will say, whoa, hold on, stop. Well, because, you know, no one's meant to use SpinRite.com securely to get to the same IP address. It's just sort of a - I had to aim it somewhere.

And the other thing is they were talking about, like, domain names that don't go anywhere. Well, yeah. I've got domain names I used for a while, and I sort of still have them, but they don't point to anything. They're unresolved. It's like there is a lot of sort of debris around the Internet. So based on what the story says, I'm sure this must be what happened. It fits the facts perfectly. And it's not anything for anyone to worry about.

Leo: Whew.

Steve: Yeah.

Leo: Breathe easy. A sigh of relief. Stand down. Question 3, Corby in Reno, Nevada. He's wondering about "Almost Perfect Passwords." Steve, I've been using your PPP for a long time. Recently, my wireless NAT router went belly up and I had to replace it. Unfortunately, it also meant re-entering a Perfect Password into my two TiVos. Oh, this sounds painful because they're 64 characters; right?

Steve: Yeah, they are. They're long, and they're gnarly-looking.

Leo: And they're random.

Steve: Yeah, very.

Leo: As all good passwords should be. TiVo uses a virtual keyboard. Onscreen entering letters and symbols is a huge pain. Even worse, once they're entered, editing mistakes, impossible. You have to start over. Oh, geez. However, numbers can be entered simply by using the number buttons on the remote control. I've decided to simplify my life and use only numbers for my WPA2 AES password. Oh, that's a good idea. That makes it easier to enter, anyway.

Since your Perfect Password service doesn't have an option for generating a numbers-only password, I had to create my own "finger random" numerical password. Probably not ideal, but at least I can enter the digits into TiVos without too much fuss. Would you consider making PPP have a numeric password generator? I realize the security is lower than when using alphanumeric and symbols, but I think 63 numbers are good enough for WPA. What do you think?

Steve: Well, let's talk about that. That's a good point. First of all, the passphrase which is put into any WPA2 system, whether it's a TiVo which supports WPA, if it's your router, if it's your laptop, whatever, there's a well-defined, uniform algorithm which mashes up the passphrase you put in and results in a 256-bit key. So if you put in the passphrase "A," you get a 256-bit key, even though "A" only - it was a byte. It had eight bits. This algorithm converts it into 256 bits. It converts anything you put in into 256 bits. It uses a SHA-1 hash algorithm.

So the problem, of course, with putting "A" in is that it's the first thing someone's going to guess. And so they'll be able to crack your network. So you don't want to use that. You want to use something much longer. So what you'd like to do is use something ideally where the key itself, the passphrase itself has 256 bits of entropy. Whereas, for example, "A" can't have more than eight bits. And you could easily run through all of those. There's only 256 possible combinations of eight bits. And again, you've cracked the network in a short time if you have only eight bits of entropy. It's too easy to check that.

So the point is, first, that anything you put in is converted into 256 bits. If you put something with more than 256 bits of entropy in, it's converted to 256 bits. So at some point you sort of - your super long passphrase of upper and lower case with a wild alphabet that's 63 characters long, let's see how many bits that is, that's - say that we had, what, maybe 127 or maybe seven bits per character of actual entropy times 63 characters. Well, that's 441 bits. So that's a lot of entropy. But it's reduced to 256 bits. That's the actual key that's used for WiFi.

So in Corby's case, he wants to just use digits. And so that means he's got zero through nine. Well, we can - we know how to figure out the equivalent amount of entropy, the amount of bits, essentially, in an alphabet of any size. If we have 10 characters, zero through nine, the log10 over log2, that is, because we're wanting to do binary bits, so it's log2, that tells us that we have 3.322 bits per digit. So an alphabet of 10 characters, zero through nine, gives us 3.322 equivalent bits per digit. So 63 of those, assuming he's going to fill up the whole input box for his WPA, is just 63 times 3.322 bits per digit, which is 209.281, 209.281 equivalent binary bits. That's a lot.

So the fact is, okay, you didn't get all the way up to 256. You didn't go over 256. There's really no point of going over 256. But you got to 209. That's an incredibly strong key. So, yes. If you do a good job of choosing randomly just the numeric characters zero through nine, remembering that each one of them gives you 3.322 bits of strength, if you use 63 of them, you get a little over 209 equivalent bits of key strength.

Leo: That's plenty.

Steve: And that's plenty.

Leo: Especially for WiFi. I mean, I don't...

Steve: Yup, exactly.

Leo: I don't even go that far.

Steve: Nope.

Leo: Question 4, let's...

Steve: Actually, we skipped #2, which was a good one.

Leo: Oh. Whoops. Whoops. Whoops. David Newton in Leamington Spa, U.K. suggests - oh, yeah, sorry about that - the EFF's HTTPS Everywhere. For the Q&A this week I recommend you feature a new Firefox extension called HTTPS Everywhere. It's created by the EFF - the Electronic Frontier Foundation - and aims to automate the process of using TLS for web pages where possible. For example, after installing the application, all pages on Wikipedia automatically go to their TLS version. Awesome. It uses regular expressions to redirect URLs so that the TLS version is automatically used. I guess it rewrites the URL when possible.

Steve: Yes. I wanted not to skip this because many, many listeners have written in to say, hey, what about HTTPS Everywhere? You've got to talk about that. What do you think about that? Well, it's cool. But it's sort of limited. It's exactly as you said, Leo. What's happening is it's not redirecting. Redirection is a reserved term meaning that you went to somewhere, and the server said go somewhere else. So that's redirection.

What it's doing is exactly as you said, Leo. It's doing an on-the-fly rewrite of the URL you put into your browser. So what this extension does, essentially, and it's very nice and clever, is it has a file of JavaScript regular expression matching. So that it has sort of a knowledge base per website. So Wikipedia's got some expressions. Google's got some expressions. Foursquare, I mean, Fourspace, you know, whatever, all the different things it knows about. It knows - so it has to have some domain-specific awareness of, like, which pages go where and sort of how to get the equivalent page. Many times it's just a matter of putting the "S" on the HTTP. And that's the way to GRC. GRC's pages are symmetric. Some sites are a little trickier. And it uses regular expressions, so you can match complex phrases in a sophisticated way.

So I would say it's a nice add-on. It doesn't do anything to sites it doesn't know about. And it's user-expandable. So if there are sites that you care about, you can edit that template to augment it so that you can just put in HTTP and it'll automatically do the right thing. Which is, you know, I think it's very cool. So it's not like the total answer to securing the web because it only is able to munge the URLs that it has permission to, that it knows something about by virtue of them being in this template file.

I imagine over time this file's going to grow, and it will acquire an increasing knowledge base of sites that it's able to make secure. And, I mean, it would really be nice if just all web browsing of all kinds was over HTTPS. From a privacy standpoint that would be great. We've seen, like, there's some downside because people that want to be able to filter content are unable to do that. But still it would be nice if it were possible.

Leo: I'm sorry I left you out.

Steve: No problem.

Leo: And I'm glad we could get that in.

Steve: Good, good question.

Leo: So Christoph Angerer in Zurich, Switzerland asks why encrypted security is all or nothing: Steve, I was just listening to 254 - our last episode - your discussion of open vs. encrypted wireless routers in the Google case. What I was wondering and wanted to hear your take on is this: Why do the standardized security solutions always have to be 100 percent, always on; or always off, nothing at all? I understand, for securing a private wireless router, encryption plus a password is absolutely necessary. But why isn't there any router that only encrypts the traffic, like HTTPS does, so that sniffing the data is prevented, but you still have the convenience of an open Starbucks hotspot? Maybe this wouldn't prevent somebody from spoofing a Starbucks router or something; but if the alternative is to send everything in the clear, then I think it's still way better.

For example - I'm trying to get my head around what he's saying here, Steve. But maybe this example will help. As a website owner, which I am, I have a similar issue with the certificate in HTTPS. To prevent a man-in-the-middle attack, the certificate is necessary. Okay, I understand that. But it still would be nice if there were a standard way to just encrypt the data on the wire - for example, user passwords - without having to buy expensive certificates and the implied fixed IP - well, that makes sense to me - and without having the browsers complain about self-signed certificates. My little site doesn't have to be as secure as my bank, but I don't want to make stealing user credentials too easy, either.

As a summary, I think there ought to be a middle ground in security where one is allowed to just lock the door but not be required to construct a full-fledged Fort Knox out of everything. Love your show, listen to every episode the second it's out. Christoph. How could you do that on a router? I'm trying to think what that would mean.

Steve: Well, he's really saying sort of like SSL without authentication.

Leo: Right.

Steve: He's saying the reason we pay VeriSign...

Leo: Just encrypt it.

Steve: Yes, encrypt it, but don't worry about authentication.

Leo: Right.

Steve: Now, first of all, I agree. It would be - we could certainly have a protocol which, during the connection, the sides exchange some randomness which allows them to agree at the beginning of the connection to a symmetric key, and then they both use that for encrypting and decrypting the traffic. That could absolutely be done. And he understands, clearly, that what's not preventable if we do that is an active man-in-the-middle attack because, if you don't authenticate, then you don't know that you're not setting up an encrypted connection to a bad guy, who's then setting up an encrypted connection to your destination. Meanwhile, he's able to decrypt and then reencrypt and see your stuff in the clear.

But he's completely right that, if we understood that we're not authenticating the endpoint, there's no reason you couldn't have a lightweight simple protocol, which unfortunately we don't have, which simply chooses a random key and uses it as the cipher for that connection. Could absolutely be done. It would mean that, like, the Google problem of sniffing WiFi, it would mean that passive eavesdropping would completely go away. Active man-in-the-middle eavesdropping, which is arguably much more sophisticated and difficult, it would not solve that problem. But we don't have any solution for that problem for non-SSL now anyway.

So, yeah. It's really a good point. If we know we don't get authentication, if we know that we're not having protection, we don't really know we're connecting securely, I mean, we'd need a different protocol. We couldn't show a padlock and give people the idea that they had a secure connection to the bank. So the padlock would have to mean SSL authentication. Anything else would just be, like if there were, like, some optional protocol that servers started to support, that clients supported, where it's like, okay, look, can we just - here's a key. Let's use it so that what we're doing is not in the clear. And we're going to agree, no padlocks, no green bars, nothing indicating security to the end-user; but at least the traffic going through the air or over the network, it's been scrambled. Hasn't been, you know, made bulletproof, but sure is better than it just being in the clear. And totally doable.

Leo: Interesting. Maybe we should do that. I think there are hotspots where you can go, and it's encrypted, but you don't have to log in. Seems like there are some - there is something like that. I don't know. You have nothing to say? We're out of time, so I think we're going to table the rest of these questions. And there are so many good ones, I hate to do it. But we must move on. My fault.

Steve: Well, they will keep for two weeks. And we'll pick up where we left off.

Leo: And if you have questions for Steve, you go to GRC.com/feedback, you get them answered. Well, maybe you'll get them answered. He doesn't - I don't think you answer personally, do you?

Steve: No.

Leo: Who has time?

Steve: And we get hundreds. And so I read what I can; I answer what I can. And but please, I mean, that's the source for all of this. So we do need input and feedback. And it really helps me to sort of profile the show. For example, next week is finally, at long last, the full cryptographic system presentation of LastPass.

Leo: Oh, I'm so excited about that. That's great.

Steve: Yes. And it's only because people have written in and said, hey, what about LastPass? You were going to talk about LastPass. You going to talk about LastPass? You said you were going to talk about LastPass. It was like...

Leo: All right, I'm going to talk about LastPass.

Steve: But if I didn't have that feedback, I wouldn't know. And it would have - might have fallen through the cracks. You guys didn't let it happen. So next week, LastPass.

Leo: There you go. Thank you so much, Steve. Steve Gibson is at GRC.com. That's the place to find 16KB versions of this show. You can get transcripts, too, thanks to Steve and Elaine, who writes this all down and then posts it. And of course SpinRite, the world's finest hard drive maintenance and recovery utility, a must-have if you have a hard drive. You can also find lots of free utilities, Perfect Paper Passwords, and more. GRC, Gibson Research Corporation, dotcom.

Steve and I do this show every Wednesday, if I don't start late, at 11:00 a.m. Pacific, 2:00 p.m. Eastern Time, that's 1800 UTC. You can watch live and even participate live at live.twit.tv. There's a chatroom link right there you can watch. In fact, we have an iPad app, an iPhone app, many other apps that you can watch this show live. We're getting more and more live streaming out - in fact soon I hope on the Roku. But you can always download it, too, from all of the general places you get your podcasts, or just go to TWiT.tv/sn and you can subscribe there, TWiT.tv/sn. Steve, thanks so much. We'll see you next week...

Steve: Right-o, Leo. Thanks very much.

Leo: ...on Security Now!.

Copyright (c) 2006 by Steve Gibson and Leo Laporte. SOME RIGHTS RESERVED

This work is licensed for the good of the Internet Community under the Creative Commons License v2.5. See the following Web page for details: http://creativecommons.org/licenses/by-nc-sa/2.5/