
# Listener Feedback #99

**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-263.mp3
Quarter size (16 kbps) mp3 audio file URL: http://media.GRC.com/sn/sn-263-lq.mp3

---

**Leo Laporte:** This is Security Now! with Steve Gibson, Episode 263, recorded August 25, 2010: Your questions, Steve's answers, #99.

It's time for Security Now!, the show that covers all those important security issues with Mr. Steven Gibson. He's here from GRC.com, the man who discovered spyware, coined the term, wrote the first antispyware program. He wrote SpinRite, the world's best hard drive recovery and maintenance utility. And he is a wizard in technology. That's why we ask him to come and explain this all. This is a Q&A segment. Hello, Steve.

**Steve Gibson:** I do love technology.

**Leo:** I loves me some computer stuff, yeah.

**Steve:** When I explain myself to people, when they're trying to figure out what the heck are you, I say, "I just love technology." I do. Applied science.

**Leo:** Yeah. I always point out that people often say, well, I don't like science, but I love technology. And I always point out that that's the same thing. It's applied.

**Steve:** Exactly. Exactly. It's you take the science, and then you do something with it. That's technology.

**Leo:** Right. These iPods don't work on voodoo. So today a Q&A?

**Steve:** It's Q&A week, and we're - need I, dare I mention we're approaching another benchmark, another milestone. This is Q&A 99, so of course we know what happens in two weeks.

**Leo:** Yeah, another Q&A, 100. We're going three digits.

**Steve:** Our 100th Q&A. Boy, they do go quickly, Leo, a hundred of these.

**Leo:** Well, before we get to questions, and I have a bunch of them, I know you have security news. And as always, we begin the show with the latest updates.

**Steve:** We've got updates and news. Not too much in the way of updates, although Adobe was once again forced to release an out-of-cycle update in order to respond to the new problems that were revealed at the recent Black Hat and DefCon conferences. Adobe, for some reason, as we've discussed before, believes they will only need to update their products quarterly, despite the fact that everybody else has to update them continually. So far Adobe has not succeeded at doing that. There's been some rumbling that they were going to switch to a monthly patch system, like Windows has with Microsoft. But nothing so far.

So I just wanted to let people know that they can expect to, if they haven't already, seen Reader and Acrobat updating on all platforms - Windows, Mac, and Unix; that they should be at 9.3.34 for Reader, 9.3.4 for Acrobat. And if you're still back on the v8 train, that's 8.2.4 on all the platforms. So once again, Adobe - I'm not even going to bother with all the stuff they fixed. Just update.

Chrome also got a big update. Google's Chrome browser is now at 5.0.375.127. They fixed 10 vulnerabilities, two which were critical, and six were high risk. And what I found was interesting was that Google chose to block public access to its bug-tracking database specifically to prevent the flaws from being exploited. So here we're sort of seeing arguably a problem with the whole open source model, the "we're going to do everything in public view" approach, because Google themselves are saying, uh, we don't want people to see this because this makes it too easy to exploit these problems. So I thought that was interesting, that they had chosen to do that. One of the two critical flaws causes memory corruption, which of course can potentially be matured into a remote exploit of some sort; and other one causes a crash when the browser is shut down. So, anyway, those are fixed.

And then Apple did a relatively sizable update. When I turned my Mac on, as I do actually generally only weekly, unless I'm playing with iTunes and my iPad stuff, I got a 64MB security update - Apple called it the 2010-005 update - which fixed a number of different things in the ATS system. CFNetwork is their core services foundation networking. They discovered that a default had been set wrong, which I thought was interesting, which was permitting anonymous SSL/TLS connections. Several people reported that, and they fixed that with this update.

Their ClamAV, which is part of their server line products, had multiple vulnerabilities

fixed. The one of about 10 of these that caught my eye I thought was really interesting. In the libsecurity package, what they said, the impact they cited was an attacker in a privileged network position who can obtain a domain name that differs only in the last characters from the name of a legitimate domain may impersonate hosts in that domain. I thought, huh? And then they go in to describe it in more detail. They said an issue exists - or did. After this update, this is fixed.

So an issue exists in the handling of certificate host names. For host names containing three or more components - like www.example.com, for example, those being, each of those chunks being a component - in a name containing three or more components, the last characters are not properly compared. It's like, what kind of a screwy bug is that? And in their example they say, in the case of a name, for example, containing exactly three components, only the last character is not checked. For example, if an attacker in a privileged network position - whatever that means, they don't really explain that - could obtain a certificate for www.example.con, c-o-n, the attacker can impersonate www.example.com, c-o-m. This issue is addressed through improved - and I love this, finally, the last sentence. "This issue is addressed through improved handling of certificate host names." It's like, yeah, like properly checking to see if they're the same or not. But anyway, so they fixed that, which is of course good news.

**Leo:** Security news, as opposed to updates.

**Steve:** So, yeah. Once again we're with Microsoft and Windows, not surprisingly. A big new problem that's got the security community buzzing because it's not directly Microsoft's problem, although it relates to the way Windows works. Apple knew about this four months ago, in March. And one of the fixes they made to iTunes fixed it. The problem is that as many as more than 200 Windows apps are implicated in this problem.

So here's the story. In the past there's been various ways of malware exploiting the order in which Windows searches the hard drive for pieces of applications that are loading. For example, certainly, probably all Windows users have seen these DLL files, Dynamic Link Libraries. The idea is that many applications have an executable portion, the so-called EXE, the E-X-E; and then also may have more code that's not in that EXE, but are in DLLs. And when the application runs, Windows looks to see what other DLLs are necessary. Some applications load the DLLs that they need dynamically, thus the word "dynamic link loading." They load them, like, explicitly. If they know they're going to need it, then they'll say, hey, I need the following DLL.

Well, Windows has a sequence that it goes through for searching for the DLL that an application has asked for, when the application uses something called LoadLibrary, which is the function in Windows that applications use, asking Windows to please load this library for them into their application space. Windows looks at the directory from which the application was loaded first. If it's not there, then it looks in the system directory. If not there, it looks in the 16-bit system directory. If not there, in the Windows directory. If not there, in what's called the Current Working Directory, which is sort of like the current path that you're logged into, for example, if you're using a DOS box. And then if still not found, it looks through the path environment variable, which typically has tons of different directories that are enumerated.

So what malware guys have exploited in the past is the idea that, if there was some way for them to get a malicious DLL named the same as a good DLL, and somehow get it in one of those places upstream in that sequence that Windows uses for searching, then they could get their DLL to load first.

**Leo:** Oh, that's a clever man-in-the-middle.

**Steve:** Yeah, it is. And then, in order to hide the fact that they had done that, since they would want to mask the fact that they've come in, then they would turn around and load the proper DLL themselves so that the DLL that was supposed to get loaded ends up getting loaded, but they get themselves loaded in the meantime.

So get this. What has been discovered, and a security firm called Acros, it's a Slovenian firm, they disclosed last Thursday that what they call "binary planting," other people call "application DLL load hijacking," they disclosed that this was a flaw in iTunes which Apple had fixed, but that another 40 applications that they had discovered were doing the same thing.

And so what happened was our friend HD Moore, who does the Metasploit Framework and has been very active in the security field, he had apparently run across this himself when recently looking in detail at that recent Windows .LNK exploit. He had been planning on quietly advising a number of companies that their applications, things like Photoshop, and I discovered CorelDraw, and many other very popular programs were also having this flaw.

So what some applications do is, when they load a specific item, like when Photoshop loads a .PSD, a Photoshop drawing file, for some reason which to me is unfathomable, they set the current working directory to the directory where that item is found. Just they don't have to, but they do. And not all apps do, only some do. Well, it turns out that, if malware were also installed in the same directory as where that asset was being loaded from, and if that malware were a specifically named DLL, which that program, like Photoshop or Corel or whatever, was then going to load in order to process that asset that you've asked it to load, then of course, for exactly the reasons we've just explained, that there's this weird DLL searching sequence that Windows goes through, this is a way of a bad guy to get their malware to load.

Now, the bad news is this works over fileshares and WebDAV accesses, which is to say, out on the Internet. Because unfortunately WebDAV is this HTTP-based protocol that allows you to create connections over the Internet, very much like file and printer sharing, essentially. And all Windows recently has had the WebDAV client present and running by default, which creates a big vulnerability.

Now, HD Moore just created in the last day an auditing tool which anyone can run to check their system for vulnerabilities to this. And so as part of doing my due diligence for reporting this, I ran this thing. I don't recommend people do it.

**Leo:** You do it so we don't have to, Steve. That's the idea.

**Steve:** Just like Jerry has always said. It's horrific. First, it pulls every file extension that your system has registered, and then it starts launching exploits against those file extensions, which cause most of the apps in your system to launch. Meanwhile, Process Explorer is running in the background, logging the DLLs that each of these processes attempts to find. You then export the log that Process Explorer generated as a CSV, a comma separated values file, back to the directory where this auditing tool is running. You run a second JavaScript against it, and it then parses the CSV file produced by Process Explorer, in order to create proof-of-concept exploits against everything that it's

found. Which is how I learned that CorelDraw is one of these things.

Well, most troubling is that the two main scripting engines, cscript and wscript, that are in my system and in everyone's Windows system, are also vulnerable. So it's not just obscure, like, upwards of 200 third-party apps, but even Microsoft's apps are vulnerable, which is known by the industry at this point. Firefox is vulnerable. WinRAR is vulnerable. Wireshark is vulnerable. Which is to say that, well, what this essentially means is that, if you execute a shortcut which refers to an asset out on the Internet, which is a shortcut for any of these upwards of 200 executable applications, which many of us have on our systems, I mean, we all have cscript and wscript installed. That's the Windows scripting host executable. And if the file goes out to the Internet, reaches out to get it, because of the way these specific applications are coded, they set the current working directory to that remote location.

Then they call LoadLibrary, asking Windows to load a specific DLL. And if that DLL - and the bad guys, unfortunately, can easily determine all this just doing the same thing I did because Moore's auditing tool builds these little DLLs for you and leaves them all behind in a big directory structure that it creates. So nothing is unknown. There's no mystery anymore about this problem. So when a piece of malware is properly named, it will get loaded by Windows. And part of what happens when a DLL is loaded is there's a standard DLL initialization routine called by Windows in the DLL that gets it running and allows it to initialize itself. That will run, and then your computer is owned.

**Leo:** So it runs automatically.

**Steve:** Yes.

**Leo:** Windows does it for you.

**Steve:** Yes.

**Leo:** How convenient.

**Steve:** Yes. How friendly. Now, Microsoft has responded. There's a knowledge base article 2264107. So that's support.microsoft.com/kb/2264107. This is one of a number, I mean, Microsoft's scurrying around now. What's interesting is that they have told people they're not going to fix this. They've said something about maybe in a future service pack, but that they're not going to fix this. Now, the problem is they kind of can't because fixing it would mean changing the order in which DLLs are found, which everything is dependent upon.

**Leo:** Right.

**Steve:** I mean, who knows what would break? I mean, it would just be a disaster. This is like - this has been the way Windows has always worked. And so this is one of those things that you just can't come along afterwards and say, well, we wish it hadn't always worked this way, but it does.

**Leo:** This is kind of a nightmare scenario, where you have a functionality that's critical and is an exploit.

**Steve:** Yes. And so what can be done, what Microsoft has done in this knowledge base article, there is a something, a patch that you can download, one for every different version of Windows, and change or set some registry keys that this patch will take advantage of to specifically block the most dangerous instances, which are shared folders, remote shares, and WebDAV. So that, if this becomes a problem, I mean, everyone's now expecting that a week from now on Security Now! we'll be talking about this having hit, gone into the wild. As far as we know, it hadn't been exploited yet. But it's probably at this point just a matter of days, if not hours, before these start getting emailed to people, and the bad guys figure out - I'm sure they're working on it right now. So...

**Leo:** Geez, Louise.

**Steve:** The problem is, looking at this patch that Microsoft has offered, it seems like an unclean fix to me. I've looked at it.

**Leo:** Unclean, unclean [laughing].

**Steve:** I can't even recommend what setting to use that really protects people because I don't understand really the way Microsoft has designed this. It doesn't look to me like any of the options they provide in this knowledge base article are ones I would want, and I don't understand why. So I wanted to let everybody know that we've got this problem.

**Leo:** There's nothing we can do about it.

**Steve:** It's a problem intrinsic to Windows. Really what Microsoft is saying is, everybody else has to fix this. Photoshop - naturally, of course, Adobe. Photoshop, Corel, they've got to fix their wscript EXE and cscript EXE executables. And WinRAR needs to fix it, and Wireshark has to fix it, and Firefox. Firefox is apparently vulnerable, also. So everybody who uses the LoadLibrary function to dynamically request DLLs, as a consequence of loading specific assets, like displaying a Photoshop drawing, apparently Photoshop calls for some DLLs that it doesn't explicitly provide the pathname to. They just assume Windows is going to find it for them. Everybody needs to fix their EXEs. And...

**Leo:** Can I just say one word? Crap.

**Steve:** Yeah.

**Leo:** This sucks. Three words, sorry.

**Steve:** Yeah, it's not good.

**Leo:** Wow.

**Steve:** Yeah. Now, Microsoft is saying, yes, everybody should have already fixed this. That is, they're saying only apps which are not loading their DLLs safely are going to be vulnerable to this. And I guess you could say, well, that's why there aren't hundreds of thousands of apps that are vulnerable. It's only a few hundred so far that have been identified.

**Leo:** Woohoo.

**Steve:** Yeah.

**Leo:** That's good news.

**Steve:** So that's our big security news for the week.

**Leo:** Wow.

**Steve:** It would be good if Microsoft gave us a better fix. I've studied what they're offering, and I can't figure out myself what setting I would want. So again, it's support.microsoft.com/kb/2264107. And I'm sure I'll have more to say about this next week. And probably, with any luck, we'll be talking about applications which are being updated to fix this. I would say I wish that there was a better auditing tool.

What HD Moore put together, he put together quickly using a couple of batch files which invoked JavaScript. My system wouldn't even do that because I don't have associations to JavaScript. So I had to create a .js association, change some registry stuff, make it all go. And then it was really sort of horrific to watch this thing run. It took about 45 minutes. You know, it was, like, launching everything on my system; and, yeah, I don't recommend it. But if people are concerned, there is that, that would give them some sense for what applications are vulnerable. Maybe at the corporate IT level that's what they should do to see whether their own programs that they're dependent upon...

**Leo:** That's a good point. It might even be also in the house line of business stuff that they use that's vulnerable.

**Steve:** Yes, yes. I mean, so this could be - this is the kind of thing that, you know, the so-called "weaponized email"...

**Leo:** Right.

**Steve:** ...or "spear phishing"...

**Leo:** Spear phishing, yeah.

**Steve:** ...would take advantage of. So this is now really in the news and in the security community's crosshairs, which means it's in the bad guys' crosshairs, too. And I'm sure we'll be talking about it.

**Leo:** Okiley-dokiley.

**Steve:** In the ongoing, unfortunately never-ending, Google WiFi Snoopinggate story...

**Leo:** Which we, by the way, we should just point out have both agreed is kind of a tempest in a teapot.

**Steve:** Really is. But now a privacy group in Spain has sued. And so a Spanish judge is dragging Google into court to explain themselves. The multiple, I think it was eight class-action lawsuits that had been filed have been consolidated into a single one, thank goodness. And they may apparently be joined by five others. Some sort of California court has been given the venue for this, so we'll see what happens there. I hope this, again, just goes nowhere.

**Leo:** You know, in Britain they examined it, and they said, no, there's nothing here. Move along.

**Steve:** Yes. They're leaving their options open. And Germany has recently received a new tool from Google which was original going to be available for four weeks, and Google doubled that to eight weeks, which allows people to opt out of Street View showing their homes. So using this, you're able to stick a pin in a map somehow and say, here's my home, I want it blurred. And then so before Google takes their Street View service public in Germany, they're giving people two months' time to identify specific locations which they want blurred out, and Google will make that happen. So it's a big mess.

And many people sent to me through Twitter and also in email this note that it has come to the attention of the world that this plane crash, the Spanair's 2008 plane crash, which killed 154 people out of 172 that were onboard, was apparently not the fault of malware, but the reporting system which should have notified authorities in time, after three instances on this particular plane, that the takeoff flaps and slats had failed to extend as they should have. That was reported three times.

Malware that was apparently infesting the reporting system, it is now believed, caused it to fail to report this problem. Had the malware not been present, it's believed that notification would have been logged and noted in time, and this plane would have been grounded, pursuant to them figuring out what was going wrong with the flaps. So that was in the news. Apparently it was a Flash drive-based problem on a Windows-based system.

**Leo:** Now, Ed Bott, who I know you know, in his Microsoft report said, "Fact check: Malware did not bring down a passenger jet." I haven't read the article. So it may be there is some question about this.

**Steve:** Yes. And they're looking into it now. So, many people mentioned it. I just wanted to let people know that I had seen it and to pass it on. And there were two - this is just my own little grumble - two graphics-related kernel problems recently found in Windows 7. Not much is known about them yet. They've been assigned CVE index numbers. And so perhaps in a week we will know more. I'm grumbling about it because it really does seem wrong to me that graphics-related problems are in the kernel.

This is one of the things that Microsoft did that I think was a fundamental security mistake, was they moved the GDI, the Graphics Device Interface, from user space into kernel space because the user space to kernel space transitions, where you have to cross this boundary of privilege in order to get the kernel to do things for the Graphics Device Interface, they said, oh, this is many years ago, many generations of Windows ago. They said, oh, let's move that into the kernel to make Windows faster.

Well, when you did that, suddenly graphics stuff became kernel stuff. And then graphics programming errors then become kernel, privileged kernel errors, which is exactly what we're seeing now. We've been seeing problems like this before. Now they're affecting Windows 7. So it's just - it's like fundamental architectural policy decision that Microsoft made which was wrong and is now biting them. And unfortunately, biting all of us Windows users, as well.

**Leo:** I'm going to have to find this article. I just read an article this week by a security guy who said really the fundamental decision where we went wrong goes farther back than that, although that same kind of decision, which we decided to go with the von Neumann architecture, where data and program were intermingled.

**Steve:** Right.

**Leo:** And there are certainly advantages to that. For instance, you could run code out of the data space. But that's the problem also.

**Steve:** Yeah, flexibility. And thinking, just going back to the Spanair thing, I want to be careful. Maybe I said the right thing because I didn't say that malware brought the plane down.

**Leo:** No, you were right about that.

**Steve:** Yes. As I understand it, it's that malware may have prevented a recognition of the problem that the plane had.

**Leo:** Right. Although here's what Ed Bott says. You should read his article on ZDNet.

He says in 2008 two of the mechanics involved in that crash were brought up on manslaughter charges. He said the malware was a symptom of a larger problem in that facility, where it wasn't just the malware. They were just not very - they weren't good at what they were doing, and they weren't...

**Steve:** Ah.

**Leo:** And of course, if you've got machines that have malware on them that are on the Internet and are doing mission-critical stuff, that is certainly a sign of, as we've said many times, not such a hot setup.

**Steve:** Yeah, exactly.

**Leo:** So I don't - there's more to it. If you want to know more, read the Ed Bott thing. But the facts that you stated are the facts, as I understand them.

**Steve:** Right. And I just have a short little note from a thankful SpinRite customer: "SpinRite saved my butt." I think not literally, but figuratively. I hope. Otherwise we've found a new purpose for SpinRite. He says, "About two months ago I bought SpinRite when my hard drive failed. Steve, SpinRite saved my butt. I was writing my end-of-the-year term paper and had to go to class. So I shut my computer off. When I got home from class and turned my computer on, it failed to boot. After nearly having a" - he actually wrote "hard" attack, and I guess it was a hard disk attack, but in this case I'm sure he meant a "heart" attack - "I bought SpinRite and let it run all night. When I woke in the morning, SpinRite had finished, and the drive was completely repaired and restored. I was able to back up my hard drive and save my term paper. Thanks to you, I turned in my last term paper of the year on time. SpinRite may be expensive" - and I guess, for a student - he said, "but it's worth every penny. Thank you in advance for your help in this matter."

**Leo:** Yay.

**Steve:** Yay.

**Leo:** Yay. I like happy endings. Yay.

**Steve:** We get happy endings with SpinRite. And in this case it's a different kind of ending.

**Leo:** So now let us get to our Q&A, if you are ready.

**Steve:** Absolutely.

**Leo:** Got some great questions. Steve has compiled these. We should tell people, if you want to ask a question of Steve, the best, really the only way is to go to GRC.com, that's Steve's website, GRC.com/feedback. There's a form there, fill it out, Steve reviews those. And while we may highlight a person, individual person for the question, often Steve picks questions that many people ask.

**Steve:** Right.

**Leo:** So we welcome all the questions. And I guess, Steve, you pick the topics based on what people are most interested in this week.

**Steve:** Sort of what's happening, things that refer to recent shows. Sometimes they're not questions. We have some things that are just comments here, which are just like useful observations. So it's our listeners' opportunity to be heard, too.

**Leo:** Here we go with number one, Nick in New Brunswick, Canada. He's asking about the math behind password strength. We've been doing a lot of math live on the show lately, calculating numbers of bits. Steve, I love the show and love the way you explain complex issues. I was wondering if you - and actually, I wasn't going to say anything, but I had the same question - if you could explain the math behind password strength sometime, and how bit entropy relates. I've been doing a lot of research, discovering more questions that need answering. For instance, when someone says "NIST recommends a 128-bit password," how is that calculated? I understand that bit entropy is calculated by log2 of a base - well, he's way ahead of me - where a base is the number of possible characters. So if it's 128 characters, it would be log2, 128 characters. And by multiplying that result with the number of characters in the password you achieve a bit entropy length for the password. Well, that clears it up [laughing]. But is it the same as stating "My password is X bits long?" In other words, are those equivalent calculations?

**Steve:** Okay. So say that we had an alphabet of just two characters, like 1 and 0. Then it's very clear that the number of possible passwords made with that alphabet is two raised to the power of the number of characters in the password.

**Leo:** Oh, okay. That I understand, yeah.

**Steve:** Yeah. So, for example, if we had - say that we just had a two-character password, that is, two bits. And they're bits because the alphabet from which we formed the password is only 0 and 1. Then we know that there's four possible combinations. This is probably, like, the weakest password ever invented.

**Leo:** Yeah. But easy to understand.

**Steve:** Because we've got 00, 01, 10, and 11. Those are the four possible combinations. So...

**Leo:** Two to the twoth.

**Steve:** Two to the twoth, exactly. Two bits that can have two states. Now, in a normal password that we've talked about, we've got the good news is an alphabet, that is, the domain of possible characters from which the password can be formed, many more than just two. So there might be, for example, if we had lower case, that gives us A through Z, which is 26. If we add uppercase to that, and the case is sensitive, that means it matters what case we use, then we get another 26, bringing us to 52. If we add digits 0 through 9, now we go from 52 to 62 because there's 10 different digits, 0 through 9. So now we're at 62. If we were to add two more characters, like plus and minus, that brings us to 64.

Now, I've put us at 64 because that's one of our special, easy to think about, powers of two numbers. 64 is the number of possible combinations of six binary bits. That is, in the same way that two binary bits gave us four combinations, six binary bits gives us 64 combinations. So you could say that a one-character password, where it was an alphabet of 64 characters, meaning upper and lowercase alphabetic, the 10 digits, and also the plus and minus characters, that's 64 characters. So you could say that that password, if it had just one character, had an entropy of six bits because there are 64 possible passwords and six binary bits gives us 64.

So, similarly, if we had two characters in that same alphabet, a 64-character alphabet, with two characters, well, each character, as we've just seen, gives us six bits. So two characters would give us 12 bits of strength, of password strength. Four characters would give us 24 bits, and so forth. So that's really - that's the way to think about this. If we had characters in our password that were 128 possible characters, that is, all kinds of special characters and maybe smiley faces and other things, it's hard to get up to 128, which is the next power of two.

So normally we're at some odd point somewhere. We're, like, maybe 94 possible characters, if we add all kinds of special characters and things. And even though it's not as easy to calculate the entropy when you have an alphabet from which you're making your passwords which does not have a power of two number of characters, you can use logarithms, which I've done here on the fly in the past, to create, essentially determine how many bits of equivalent strength a password is. It's easy when you have a domain, an alphabet of 64 characters. Then it's just how many characters times six bits. Or if you had 32 characters in your alphabet, how many characters times five bits and so forth. But you can calculate it for arbitrary alphabet sizes, as well. And so there you go.

**Leo:** There you have it. You asked, and now you know.

**Steve:** That ought to be clear.

**Leo:** That's - as mud. No, I'm kidding. Joshua Backes, Shreveport, Louisiana, believes he got "rebound": I believe that our Netgear router at my job, where I am the computer tech, had fallen victim to this new type of rebound attack. A few weeks ago our computers started randomly redirecting to a few different websites, as well as a Google analytical site - or actually it was google-analytical.com, which is not a Google site - and would not load the page intended. After reinstalling Windows on

two machines, we discovered they began redirecting within a couple of minutes. Ooh. Wow. So now they know it's not the machines.

**Steve:** Right.

**Leo:** Our final resolution was to reset the router to default, and then the rest of the computers began working fine. Well, that's a good diagnostic. Sounds like it was in the router.

**Steve:** And it does sound like, I mean, it sounds like something reconfigured the router. And that's what a DNS rebinding attack does. It may well have something logged into the router, and probably change the router's DNS so that these people using this router were going to a foreign DNS server, which was then playing who knows what kind of games. I mean, it's exactly what this sort of thing sounds like. So I just thought it was interesting. I tossed this in here because here was somebody who actually had that experience of rebinding that we were talking about.

**Leo:** Right, right, right. There you have it.

**Steve:** And I'll mention, it's not yet public, but I'm, like, a day away from finishing. I declared it finished yesterday, and a couple of the testers in our newsgroup reminded me of one more feature that I had promised, which I had forgotten about. GRC's forthcoming DNS Benchmark now also tests for rebinding vulnerabilities. So it will show when remote DNS servers are protecting their users from rebinding attacks, as OpenDNS has that option to do. So I just added that last week after we did the story on it.

**Leo:** Aren't you amazing. How does he do it, friends?

**Steve:** Cool.

**Leo:** It's nice to have a friend in the programming business. Thorar - oh, boy.

**Steve:** Yeah.

**Leo:** Thorarin Bjarnason - that sounds like a name straight out of "The Lord of the Rings."

**Steve:** Good job.

**Leo:** Thorarin Bjarnason in Vancouver, BC, Canada is concerned that Michael McCollum's Wikipedia page is being considered for deletion. What? This happens

from time to time on Wikipedia. In fact, we had it happen to us. There was a discussion over whether to delete FLOSS Weekly. And you know you're not supposed to have a Wikipedia page for anything unless it's important, generally important. And so the discussion on FLOSS Weekly was, well, is the show important? I think we pretty much justified it. It's kind of a form of subtle vandalism to say, well, this isn't important.

Hi, Steve. You pointed me toward the Gibraltar Series. Two thumbs up. Steve and I both love that. I downloaded Gibraltar Stars today, did a wiki search on the author, Michael McCollum, only to find his wiki page is being considered for deletion. I think Michael's page should be kept, not only because I think his Gibraltar Series is great sci-fi and worth noting, but also because his business model is interesting. Absolutely, he sells only on the website, but sells it in every form possible for e-reading. He sells easily copyable PDFs directly to customers who can choose, rightly or wrongly, to distribute the digital content immediately and widely. In other words, he doesn't use DRM.

**Steve:** Right.

**Leo:** Which means that the reader gets to choose how he wants to read something. I think his trailblazing methods of selling his wares is of potentially more note than his literature, even, and this alone should justify his existence on Wikipedia. Perhaps you can help summon the Security Now! army to keep his page on Wikipedia. Maybe the more literate among us can contribute to his page. It's of course Michael M-c-C-o-l-l-u-m. And essentially it's a democracy. Somebody proposes that this should be deleted. If you go to the discussion page you'll see where this happens. And the trick is to weigh in in a responsible, informed manner.

**Steve:** Not flaming.

**Leo:** Not a flame war. But say, no, no, this is why I believe - oh, I've got to pull this up here - why I believe that this should be kept on Wikipedia.

**Steve:** Is worthy of staying on Wikipedia.

**Leo:** And they have - this doesn't mean it's going to be deleted. Somebody, some editor somewhere said, "Who's this guy?"

**Steve:** Right.

**Leo:** And just, lookit, there's no question in my mind. But I like what this guy just wrote, what Bjarnason just wrote to us. That's the kind of thing that you would put into that article's entry on the considering for deletion. "Although I own the Antares Trilogy," some idiot wrote, "he's not really notable."

**Steve:** Yeah, well, I love his stuff. The Antares Trilogy is fantastic. The Gibraltar Series Trilogy. I mean, he's got some great books that are standalone. I mean, I love Michael's stuff. And I know that a lot of our listeners have been glad that we've referred them to him in the past. So I just wanted to say, hey, if you've got something, as you say, Leo, respectful and...

**Leo:** Factual. You've got to give it some, you know...

**Steve:** ...factual and useful, I'd love to have our listeners help out, keep Michael's page there. I don't know if it matters at all to him from a financial standpoint. But he's a real sci-fi author, and his science fiction is wonderful.

**Leo:** There's lots of authors, I'm sure, on Wikipedia who have not sold as well as Michael has sold.

**Steve:** Yeah.

**Leo:** And it's kind of stunning that, here we are in Wikipedia, where people are saying, well, he doesn't, you know, he's not a signed author, he doesn't have a publisher, so he's not real. Just the irony, it's dripping with irony.

**Steve:** Welcome to the year 2010.

**Leo:** Yeah, you know? It's like a rock band, "Well, they're not with a label." Well, that's not really how we judge people anymore. And Wikipedia's a perfect example. Wikipedia's not with a publisher, either, I hasten to note.

**Steve:** Well, and he is being sold on Amazon. And I know that his stuff is selling well. So...

**Leo:** So that would, I'm sure anybody - you don't have to go mass trash this discussion page. Just go there, if you're a fan, and say I'm a fan.

**Steve:** If you really believe, yeah.

**Leo:** Harold Kravatsky in Florida found a - notice I did that name perfectly - found a Windows LNK checker - that's for that LNK problem - that works with Windows 2000: Steve, I have Windows 2000, and I wanted protection from the .LNK exploit. Sophos had a program that only worked with XP, Vista, and Windows 7. I tried it, and it wouldn't let me install on Windows 2K. I searched further, found a program from G Data that runs under Windows 2000. After installing it, I had to restart Windows 2000 to complete the installation. The icons still look normal. So it's a better fix than just kind of turning that off, the rendering off. Below is more from G Data. Harold

Kravatsky, Happy SpinRite Customer. And he has a link to gdatasoftware.co.uk, or he says just Google "g data link checker," you'll find it.

**Steve:** Many, many of our listeners, I'm surprised actually how many, were distressed that this was not going to function, that is, that Microsoft of course is not reaching, well, not even back past Service Pack 3, not to SP2, certainly not to Windows 2000. Yet there are still people with Windows 2000 systems that like them, if for no reason other than probably they don't have to be activated by Microsoft, so they feel a little more liberated and free. So I just wanted to point out that there was a fix for this, since this is a widely exploited vulnerability that shows no sign of letting up soon or going away soon. So there is some solution for Win2K users.

**Leo:** Great news. Moving right along to Toby Wilkins in Wales, United Kingdom, rightly worried, he says, about - or you says - about contactless payment systems. Oh, you see these everywhere now. McDonald's has them. You just kind of wave your hand. Hello, Steve. I have some information you and Security Now! listeners may be interested in regarding a new "feature" for my bank in the United Kingdom: wireless credit card payments. Barclays Bank is a very large bank chain in the UK. Yeah, everybody knows Barclays.

Today I received my new Barclays debit card. I opened the letter to find a small booklet boasting Barclays' new contactless wireless payment feature built into the card. He says: "Uh-oh. Alarm bells." The booklet claims payments of up to 15 pounds - that's about 25 bucks - can be made from any new contactless enabled debit card by simply holding it close to the newly released reading device. No PIN is required. So all you need is the physical card. Holy cow. You know, I see people with these on key chains and things.

I called up the information number, freephone (toll-free) 0800 009 4220. The polite lady confirmed the above, stated this feature is being rolled out with all new Barclays cards. I asked her what is to stop a thief walking around a busy railway station with a reader - just holding it up to people's pants. Her defense was these devices are physically big - well, you'd notice, wouldn't you? - but admitted she'd never been asked this question before. We know that readers are only going to get smaller. It's probably just an RFID reader, which we know can be made very small. And I'm sure it's only a matter of time before hackers rustle up a nifty little reading device to take advantage of it. When asked, she said she didn't know if the technology used RFID.

He says: Black Hat and DefCon spring to mind. So only 15 pounds will get stolen. That adds up to a lot of money when taken from hundreds of passersby in a public location. What happens if the card is pinged, or virtually swiped a number of times? What about if it's cloned? Signatures and PIN numbers, card fraud and skimming earn thieves big bucks. Adding a wireless, no-PIN feature is only going to make this game much easier for the bad guys.

In the UK nearly all credit card and debit card transactions take place by inserting a card into a physical reader and typing your PIN into the device. That's actually not the case in the U.S. for credit cards, but it is for debit cards; right? When I visit the U.S., this does not seem to be the system used. I've never understood why the U.S. has not adopted this system as we have in the UK. I hope you found this information interesting. Great fan of the show. Recently graduated from university with a

computer security degree with first class honors results. Congratulations, Toby. He says: I'm sure listening to Security Now! was the reason for this great result. Well, I'm sure maybe you had something to do with it.

**Steve:** Okay. So I'm just flabbergasted, Leo. I received new replacement cards from Chase, and they had something called "Blink," which was this new feature. And that's what this was. And so I contacted them, and I said, "I don't want this." And they said, "You don't?" And I said, "No."

**Leo:** But it's so convenient, Steve.

**Steve:** Exactly. And they said, "Well, we'll send you out replacement cards with no Blink." I said, "Please. Thank you very much."

**Leo:** Well, it's good you can do that.

**Steve:** Yes. I mean, but Leo, how is this possible?

**Leo:** [Laughing] Have we learned nothing?

**Steve:** No, I mean, I'm truly, I mean, the fact that it's limited to $25, or in the UK 15 pounds, to me that says they understand it's dangerous. So, yeah.

**Leo:** Yeah. Yeah.

**Steve:** I really, I'm just dumbfounded. I mean, you don't have to press a button on the card. You don't have to do anything. There are no buttons on the card. You just bring it close to a reader, and it takes money from you. This is the dumbest thing I've ever heard of in my life. I'm not kidding. I mean, this is just unbelievable. I'm [stammering]...

**Leo:** Well, and it's happening here with cell phones, too. The next thing is that they're going to do that with...

**Steve:** That near field technology, I know.

**Leo:** And it's also PIN-less; right?

**Steve:** It's like, yes, isn't it convenient. Yes, children, it is.

**Leo:** It is. For everyone.

**Steve:** Oh, my god.

**Leo:** Including the bad guys.

**Steve:** Oh, I guess we're just going to have to learn a lesson. And I noticed that it's a debit card, against which you have - I understand that you have no recourse when funds are extracted. Unlike a credit card where you can challenge the charge and then get it refunded. The debit, it's gone from your account. I'm just...

**Leo:** I think that they have in the U.S. passed additional banking laws with some protection for debit cards. I think the same kind of thing happens now where you - I think $50 is the maximum you can lose and blah blah blah. But this is terrible.

**Steve:** Oh, it's just - it's so brain dead. I mean, it's unconscionable. You just go, oh, look how convenient, just wave it in the air. Yeah. The bad guys are going to have a gun that, like, pings your card at a distance. I mean, it's entirely possible. RFID technology, unless you, I mean, for example, there are passports that use this. But they have them enclosed in an RF-safe wallet, or they have leaves on the front and back outer side so that you have to open the passport in order to get internal access to the RFID chip. And when it's closed, it protects you. And there are third parties that sell RFID shields for credit cards. But most people are not going to use those. They're going to have them in their wallet, in their back pocket. And it's trivial, I mean, it's the reverse of a portable dog killer. You have a gun, and you shoot somebody in the butt with this thing and take 25 bucks from them. It's crazy.

**Leo:** Okay. Okay. Well...

**Steve:** Anyway.

**Leo:** We'll see. It's interesting. I think the bank is going to lose money, too. But I guess they've decided they make more money because of the convenience than they potentially will lose.

**Steve:** Well, and the inconvenience, then, of having us all having to go through our statements, making sure - looking for anything that we don't recognize that just bled us for $24.32. Who knows how much money they're going to take because I guess it's up to them. Ugh, unbelievable.

**Leo:** I'll take all 15 pounds, thank you. Antonio Lorusso in Swindon, UK has a thought about our last episode, Strict Transport Security: Steve and Leo, you spoke of one small problem with STS in that, if a computer connects to a fraudulent site,

say a site trying to imitate PayPal.com before it has connected to the real PayPal.com to receive the STS token, the user will not be protected. Now, here's one solution. If I were operating an STS site I would ask for browsers that support STS to come preinstalled with an STS token with a large expiry date for my site. This would not even require browser manufacturers to take the burden of verifying the validity of the request for a preinstalled STS token simply by insisting that the request is digitally signed for the site requesting the preinstallation of the STS token. Preinstalled STS tokens could also be added or updated by browser updates.

The only theoretical fly in the ointment for preinstalled STS tokens that I can see is that this requires the provision of browser software and browser updates be secure. This is never going to happen, by the way. However, if browser software is not being provided in a secure manner, we have more serious problems than STS systems being compromised. But it would be something to bear in mind with this preinstall system. What do you think? Kind of like certificates. You come preloaded with STS tokens for PayPal and banks and things.

**Steve:** And Chrome does.

**Leo:** What? Really?

**Steve:** Yes, yes.

**Leo:** Kudos to Google, once again.

**Steve:** Yup. Chrome is now...

**Leo:** Really. I said it will never happen, and it already did.

**Steve:** ...has STS tokens preinstalled. And I wouldn't be surprised if it ends up being increasingly common in the future. If STS takes off - and, I mean, it already has - we're going to see it in Firefox 4. I imagine the other browsers are going to follow. Chrome has had it since v4.something or other. And Chrome does preload a large and growing number of STS tokens.

**Leo:** But it's site by site; right? Or is there a certificate authority?

**Steve:** Site by site.

**Leo:** Oh.

**Steve:** No, it's site by site.

**Leo:** That's a lot of tokens.

**Steve:** So basically PayPal says to Google, we want you to just pre-embed an STS token. We are 100 percent SSL. We don't want anyone to ever contact us otherwise. And this does solve that first contact problem, by having the browser, essentially, if you use Google's Chrome, you cannot connect even the first time, without being secure, to PayPal. Which is a benefit for Chrome.

**Leo:** Sure.

**Steve:** And it just makes sense.

**Leo:** Another reason to use Chrome.

**Steve:** That is going to be happening in the future.

**Leo:** Wow, that's really surprising. I mean, I guess you can do it now because there aren't a whole lot of sites probably that use STS. What happens if every site starts - you're not going to put a token for every site in.

**Steve:** Yeah, it's a very good question. Random Ma and Pa Kettle's site is going to be - that's burdensome.

**Leo:** Yeah. I'm sorry, Antonio. I should never have doubted you. Thomas Crowe, Virginia Beach, Virginia, worries about a Self Denial of Service attack on STS. I like that. First of all, I want to say I've been listening, Steve, to Security Now! since the very beginning. Well, maybe since Episode 10, quickly caught up. Thank you for the great podcast.

After listening to your latest, #262, STS, a second time, I started to think about enabling this on my own website. But I realized that I could easily shoot myself in the foot if I were ever to decide not to keep up with my site's SSL certificate. They are expensive, too, of course. Another troubling scenario in general would be, what if a domain name changes ownership at some point? That domain would not be accessible by someone who sells it unless they use SSL for the next 40 years or so, or whatever the last STS token was set to. So, yeah, so if I used STS for TWiT.tv, and then sold it to somebody, they would have to - there would have to be some handoff of the certificate, I guess.

**Steve:** Well, and you'd be obligating them...

**Leo:** To continue doing it.

**Steve:** ...to continue doing it because all your visitors who'd ever been there and received a 40-year STS token, they'd still have that.

**Leo:** We could do like Skype did to eBay. We could sell them the site without the token. It would make sense somehow to tie this to DNS, where the ownership of control of the domain is actually implemented. Oh, that's interesting. It doesn't make nearly as much sense to put this at the HTTP level where it is now. I think the browser should somehow check against the DNS expiration date or see if it was renewed. As it is now, just seems to be a temporary fix, not a real solution to the problem. Any thoughts? Thanks for the show. Enjoy listening every week. Yeah, what about moving it upstream to the DNS server?

**Steve:** That's already in discussion.

**Leo:** Wow.

**Steve:** And now the problem is DNS is not secure. But it is expected that, when we get DNSSEC, when we have signed DNS records, then that would provide the security we need in order to be able to add something like an STS record to DNS. And so what that would mean would be that that also solves the first contact problem because, when your computer looks up the IP for the first time for PayPal.com, then in getting the IP it would also look for, for example, whatever type of DNS record they created. It wouldn't be an A record. That's for addresses. It might just be a text record, and there would be some text entry. The point is, it would not be spoofable. That's what we have to guard against is, is this being spoofed, for exactly this kind of denial of service reason.

So by having signed DNS records, then not only are we able to rely on the IP address that we got courtesy of DNS security, but we can rely on all the information which is then being published through the DNS system in general. And STS is a perfect candidate for that. Which would mean that the browser would receive this STS token at the same time that it got PayPal's IP and say, oh, I'm authorized to use SSL, and I'm never going to do otherwise. So it's actually very prescient, in this case, on Thomas's part.

**Leo:** We have such smart listeners.

**Steve:** We do. We've got great listeners.

**Leo:** Thought of two improvements to STS that are already coming along.

**Steve:** And I'll say one thing also. In the spec there's been a suggestion that, relative to the expiration of the SSL certificate, that maybe the expiration of the Strict Transport Security token match the expiration of SSL, that is, of a given certificate that the site is currently protected by. So that instead of, I mean, you'd really want 40 years if you're committed, as PayPal is, to always having secure connections. But if you're not, if you might, you're not sure you're going to renew your certificate, then you're certainly better than nothing to set your Strict Transport Security header to the same number of seconds in the future as when your existing SSL security certificate expires, so that they die

together, if you choose not to renew.

**Leo:** Let's see. Let's go on to Question 8 here, Matt Bender in Madison, Wisconsin. He's wondering about adoption delay: Steve, he says, every now and then while listening to Security Now! you make a usually proud reference to the fact that you're still on XP. And not too long ago we know you were still using Windows 2000. So, like you, I'm cautious about adopting new technology the minute it comes out so it can get the bugs worked out. Look behind him, Steve's still using PDP-8s. No, not for anything serious. For example, I would never buy a new model line of car the first year it comes out. That makes sense. >From what I can remember, your reasoning in not adopting the latest technology or operating system is just that very reason. It's too new. Bugs need to be ironed out, as well as possible security implications.

But based on your reasoning, if you're still using XP, why have you adopted the iPad? It's a new technology, running a relatively infant OS that has some proven security flaws. I'm not bashing the iPad, or any technology, for that matter. In fact, I really like it, although I don't have one. I'm just wondering what your thought process is on adopting new technology both for you personally and for use at GRC. Take care, keep providing quality work. Matt Bender.

**Steve:** Well, Matt's right, certainly, about my feeling for something like my main workstation PC, where the crown jewels reside, and where I spend all my time, and I've got all this stuff going on. I'm very cautious. But I bought, I preordered the Kindle, also, which is a computer running Linux. And I'm not worried about it. And very much in the same way the iPad. For me, those are appliance devices. And it's a little island sort of all unto itself. And it can't - if there's a problem there, it's pretty much contained there. It's not able to escape from the iPad and do any great damage to the rest of my computing infrastructure and ecosystem, unlike something nasty getting into my Windows machine, which is hooked into my LAN and does have access. So I guess the reason I feel differently about the iPad is just that it's security constrained by nature of the way I use it, how it connects to the rest of my world. And it lives down in my car most of the time, and I take it with me when I'm running around out of the house. So it's sort of safe.

**Leo:** Yeah. It's because of its limited use. You wouldn't use it as a general purpose computer, perhaps. And I would also add what John C. Dvorak always says when confronted with this kind of thing: "Foolish consistency is the hobgoblin of small minds." In other words, hey, you know...

**Steve:** Adapt.

**Leo:** Adapt.

**Steve:** Adapt.

**Leo:** Steve in Florida worries that STS will block the administration of his router because the Linksys cert doesn't match. Great show on STS. I've been using it in NoScript for a long time. But whenever I log onto my router's administration page, I

get a certificate mismatch error. Essentially it says, "You're trying to connect to 192.168.1.1," the router's gateway's address. "However, the name on this certificate is Linksys." I click past it. But from what you said, I wouldn't be able to do that when STS is fully implemented. That happens to me in a lot of other situations, as well. So it's true, there are some situations where the certificate doesn't match. But you expect that.

I've configured the router's admin page to accept secure connections only, to help prevent my wireless network being used by a bad guy to mess with the router. Oh, that's why he's getting that warning. It seems I have to disable that, allowing insecure connections to the router, or else I'd never get past the certificate mismatch. Of course the default password's been changed, but I still hate to change the security settings on the router admin. Any thoughts? It is, it's kind of you gained a little bit, but now you're losing a little.

**Steve:** Well, and that's funny because, as I was thinking through this, I thought, okay, and what has he gained, exactly?

**Leo:** Well, that's a good point because...

**Steve:** If he's clicking past a certificate mismatch error, and he's only using secure connections to his router to somehow thwart a bad guy, well, then, the bad guy can do the same thing. So the only thing he'd be doing by using an SSL connection to his router would be preventing a bad guy from eavesdropping on his conversation with the router because that much...

**Leo:** Oh, that's encrypted now, yeah.

**Steve:** ...would be over SSL. But a bad guy could still connect to the router. So I guess, okay, he has protected his password. So if he uses SSL, and he's established a secure tunnel to the router, then he uses that to log on. So it's true that passively sniffing his network, if his network were not also encrypted - and I would imagine Steve in Florida's network is encrypted if he's gone through all this otherwise. But then if there was a bad guy who could crack his encryption - and he certainly could not get into the SSL connection. And that would prevent the bad guy from seeing him log onto his router, obtain his router's logon credentials, and then be able to get in and do the same thing himself.

So, yeah. I guess I can see that Steve has achieved something by doing this. And my first thought in reading this note was, well, okay. You can't have everything. I mean, if you want to allow a certificate mismatch, you absolutely cannot use STS because it won't allow that. So you have to tolerate certificate mismatch in order to get SSL, or somehow - okay.

One thing you could do is, what's happening is he's logging in by using the IP address of the gateway. But apparently he's saying that the certificate name is Linksys something or other. Which means that the router does have probably a self-signed certificate. Or maybe it's a certificate authority-signed certificate, which would be very cool. But what that means is he needs to put an entry in his hosts file for the name of the certificate on

his Linksys router, which he can view when he gets this mismatch error. He puts that in his hosts file and has that mapped to the gateway IP, 192.168.1.1. Now, after doing that, he can log onto his router using its proper name, rather than the IP, which will then cause the certificates to match, and he'll get no more SSL errors.

**Leo:** Clever. So, good. Best of both worlds.

**Steve:** Solved the problem.

**Leo:** Our final question, Steve. Shall I break out the vuvuzela? No. Maybe not, no. We have Steve and Marie Kimbrough in the studio, and Steve's vigorously shaking his head no. He's sitting right across from me. He'd get the full blast.

Our last question, David Jaundrew in Victoria, BC, Canada came up with an STS-based denial of service scenario. Here you go, Steve. Great discussion on Strict Transport Security. I was very excited to hear about this new security feature, although I have thought of a scenario that could allow STS to be incorrectly enabled for non-HTTPS sites using a man-in-the-middle attack. Here we go.

A Starbucks WiFi hacker sets up a man-in-the-middle attack for a user connecting to the Starbucks open access point. The user now attempts to connect to this site that does not have HTTPS support - just some random Leoville.com. The hacker intercepts the HTTP request, because it's not encrypted, returning a page that redirects the user's browser to the same site, https://. The user's browser then attempts to connect to the HTTPS URL, which is again intercepted - and that's key because otherwise you'll get a warning; right?

**Steve:** Yup.

**Leo:** Which is again intercepted by the man-in-the-middle attack, likely using on-the-fly, self-signed certificates. The hacker now sends back an HTTPS page with the STS header, thus enforcing and requiring the use of HTTPS connections. So he's turning on STS.

**Steve:** Right.

**Leo:** The user clicks through the certificate warning. Ah, see?

**Steve:** Uh-huh.

**Leo:** So you've got to click through that certificate warning.

**Steve:** Yup.

**Leo:** And the browser reads the STS header, adding the site to its list of STS-enabled sites. The user is now no longer able to connect to the original, nonsecure Leoville.com from any Internet connection because their browser says no, no, no, we've got an STS token here. You need to use HTTPS. And because my server or the nonencrypted server doesn't support it, they can't get on.

Now, granted, the application for this is strictly a denial of service attack on the individual user because once STS is enabled, the browser would then be forced to require proper certificate authentication for the intercepted site. My two questions are, one, are the STS headers able to be initially set when the site is using a self-signed certificate? And, two, where has my logic failed me? Thanks for the podcast. Congratulations on five years.

**Steve:** And the answer is, good news, the people who designed the spec were absolutely clear that no STS header will be accepted if there is any deviation from a perfect SSL connection, specifically self-signed certificates or expired certificates or domain name mismatched certificates, nothing. No error at all will be allowed.

So in this scenario the glitch is, the thing that prevents malicious STS headers from being accepted, and STS tokens from being embedded on people's browsers, exactly like this and for this reason, is that unless you have a legitimate certificate that is completely correct, such that no warnings of any sort come up, only then will the browser say, okay, I believe the headers that I have received over this SSL connection for this purpose because I got an absolutely legitimate certificate connection, certificate for the server, from the server at the other end. And so one of the key things that Strict Transport Security enforces is it completely removes from the user any ability to short-circuit the process, which is where a lot of its value comes from because users do just click through these things. And so this is meant to say, unh-unh. We're PayPal or whomever. We're serious about security. We'll take responsibility for providing the certificate at our end, and we want the other end, the client end to do as good a job as we are of enforcing and holding up their end of the bargain. So that's what we get with this, which is why it ends up being so good.

**Leo:** That's great. That's good news, yeah. Well, Steve, a great 10 questions. A great, of course, 10 answers, as always for Q&A #99, Episode 263. Next week, do you know yet what you're going to talk about? Or is it a mystery?

**Steve:** I think it's an issue that's been around for a while, which, again, many of our listeners already know about, have sent me, "Hey, Steve, have you seen this" sorts of notes a while ago. It's a sort of an investigative technology that the EFF has put together for tracking users without cookies.

**Leo:** Oh.

**Steve:** Yeah.

**Leo:** That's not good.

**Steve:** It's a way of - it's an interesting exploration for how our systems identify themselves, even without cookies. And it turns out it's too easy to do.

**Leo:** Yeah, I saw this article, and I'm glad you're going to address it, yeah.

**Steve:** Yup.

**Leo:** So perhaps, tentatively, that's what's next week unless something horrible happens.

**Steve:** Unless some horrible cataclysm erupts.

**Leo:** Unless a fire breaks out.

**Steve:** Although I imagine we'll cover it at the top of the show in any event.

**Leo:** You always get the news. And again, I am now posting show notes. Steve does copious show notes, which he puts on his site at GRC.com, along with 16KB and full 64KB audio, transcripts, I mean, that really is the place to go is GRC.com. However, I am also putting that in my blog because I've always put out a post on Buzz and Twitter and soon to be on the TWiT Facebook page, as well, that a show has begun, and here's the notes. But the problem is that that's kind of ephemeral because it just scrolls on by. So I'm going to put it on the blog where you can get it. So people can subscribe to my blog also, either RSS or email, so you'll get those great show notes that Steve provides, automatically. Or you could just watch for it on Twitter and Buzz, and you'll get a link back to the blog post. So that's another place you'll get the notes now.

**Steve:** Are we going to be calling this the Henry's Doing His Homework blog?

**Leo:** Yes. You heard that. Well, what happened, I really got kind of an epiphany when I realized that all this stuff that you put on Twitter and Buzz and Facebook just kind of scrolls away forever. And there's stuff like that, like the show we just did, I want to keep that permanently. So that's what the blog should have been for all along. So I'm going to use the right platform for the right stuff. So that's where those notes will begin. But I'm still going to pump it out to Twitter and Buzz and all the other places. In fact, I'm using a commenting system that's also supposed to pull comments back. It doesn't seem to be getting them from Buzz right now. I'll have to figure that out. But...

**Steve:** Wow.

**Leo:** It does get them from Twitter. I notice we're getting some Twitter comments

back there. So we'll figure that out.

**Steve:** Connectivity, connectivity.

**Leo:** The idea is, and this has always been the idea of the podcasts, too: any way you want it, everywhere you want it, in any form you like.

**Steve:** You can't get away from it. You can't escape it.

**Leo:** No, I just want to give people what they want. I don't, you know, I'm like...

**Steve:** The TWiT Army is after you.

**Leo:** They're before me. Thank you, Steve. Have a great week. We'll see you next Wednesday. We do it live every Wednesday, 2:00 p.m. Eastern, 11:00 a.m. Pacific, 1800 UTC, on the TWiT stream, live.twit.tv. The chatroom is always active whenever Steve's on the air, 600 people in there right now, so you can always join that conversation, irc.twit.tv. And of course the show notes. And let's not forget SpinRite, the world's finest hard drive recovery and maintenance utility, at GRC.com. See you next time, Steve.

**Steve:** Thanks, Leo.