SECURITY NOW!

Transcript of Episode #214

Listener Feedback #75

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-214.mp3
Quarter size (16 kbps) mp3 audio file URL: http://media.GRC.com/sn/sn-214-lq.mp3

INTRO: Netcasts you love, from people you trust. This is TWiT.

Leo Laporte: Bandwidth for Security Now! is provided by AOL Music and Spinner.com, where you can get free MP3s, exclusive interviews, and more.

It's time for Security Now! with Steve Gibson, Episode 214 for September 17, 2009: Listener Feedback #75. This show is brought to you by listeners like you and your contributions. We couldn't do it without you. Thanks so much.

It's time for Security Now!, the show that covers all things secure - privacy, computers, the Internet, all of that stuff. And our guru of security is here in his locked-down fortress somewhere in Southern California, Mr. Steve Gibson of the Gibson Research Corporation.

Steve Gibson: Yes, Leo, it's great to be back with you again, as always.

Leo: Good to see you. I love - for those of you watching the video portion of our show, you'll see Steve's T-shirt. He bought the T-shirt he talked about a couple of weeks ago.

Steve: It just says "NO."

Leo: No. With a period. Which makes it very emphatic. Not an exclamation mark. Just a period. No. Just no.

Steve: No, unh-unh.

Leo: Does it say anything on the back?

Steve: No, it doesn't. In fact, some people have, like, walked around me in circles wondering if there was, like, some clue.

Leo: What's the question, yeah.

Steve: Some clue to the no-ness. And it's like, no. It's just, you know, will I fix your computer? No.

Leo: No. Will not, no.

Steve: No. No.

Leo: And no matter what you ask, the answer is no.

Steve: Yeah. It's the crusty old curmudgeon T-shirt. I like it. Cranky. It's the John Dvorak.

Leo: Just exactly what - why is it, when you say "crusty old curmudgeon," the name John C. Dvorak leaps to mind?

Steve: I was at a UCI lecture last night about the future of inquiry-based science education in K through college.

Leo: Oh, interesting. Oh, how fascinating.

Steve: Really interesting talk about, you know, just in general, unfortunately, that this country is - sort of has an anti-science orientation now, where it's - I don't know if it's that back in the '50s scientists over-promised things. I think that people want easy answers for complex problems, and complex problems often don't have easy answers. So demagoguery, which purports to give you an easy answer, is more appealing than complex, value-based, evidence-based, non-black-and-white answers. Anyway, it's a really interesting lecture. And the friend that I was with saw the T-shirt. And he said, "So what's with the shirt?" And I said, "It's the cranky old curmudgeon shirt." And he said, "Well, I know you think you are, and I know you think I am. But you're not." It's like, okay, well, guess he didn't buy it, no.

Leo: You're not a cranky old curmudgeon.

Steve: No, I'm not, really. It's just...

Leo: In fact, if anything, I'd say you're an optimist and an enthusiast.

Steve: Yeah, it's what makes this sort of funny, I think. My own...

Leo: You probably say yes far more than you would like to care to admit.

Steve: It's my own joke.

Leo: Yes.

Steve: Yeah.

Leo: Well, I think it's kind of neat that you go to lectures at UC Irvine and study up on stuff. I mean, you could just sit back and read sci-fi all day. But no. You're expanding your mind.

Steve: Nice to know what's going on out there, yeah.

Leo: My question, I wonder, I mean, I certainly recognize that anti-scientific bent. My question is, is this new? We interviewed on Dr. Kiki's Science Hour the author, I wonder if this is the guy who was speaking, of "Unscientific America"? Was it him who was talking?

Steve: No. He's sort of a politician/educator who spent about 14 years in Washington. I mean, and he is a scientist. At one point in the Q&A afterward someone asked him something. And his immediate response was, well, I'm not aware of that. Do you have - is there any evidence to substantiate that?

Leo: Oh, I love that. That's a good question.

Steve: Oh, it was wonderful. It was like - and frankly, I mean, he had no patience for talk radio and talking heads arguing with each other. I mean, he really saw sort of - he just came back, he explained that two days ago he'd been in China, and he was meeting with some of the top political management people in China on the topic of health and healthcare. One of them said that they weren't really sure what was going to work, but they were doing something one way over in this province and a different way over in that province...

Leo: Yeah. They experiment. It's really interesting how they work there.

Steve: Well, and he said that most of the management of China, for lack of a better word, are engineers, that the upper level political infrastructure are a bunch of engineers, and that they're applying scientific principles rather than just...

Leo: Interesting.

Steve: ...than arguably maybe large corporate lobby-driven politics, which seems to be what our system has fallen into.

Leo: Not that they're the paragons of virtue there.

Steve: No, there's lots of problems, too.

Leo: Just don't be an ethnic minority, everything's fine. But I do have to say that it's a very big country. And I don't know how one manages a country of 1.3 billion people spread over that kind of land mass anyway. It's a very difficult thing to even contemplate, especially with a planned economy. Must be very, very difficult. But I just think of - and I think back to the '20s when you had people like Aimee Semple McPherson, we had these radio demagogues. It's not - it kind of seems to be a strain in American life.

Steve: Well, one of the points that he made that really...

Leo: The Scopes Trial.

Steve: One of the points that he made with me that really stuck was he said that the science textbooks that we're using have about - the way he phrased it, about 1,600 - he didn't use the word "jargon." I can't remember exactly what his term was. But 1,600 words that are used in textbooks which are unfamiliar to the kids reading them. And his point was - and he used the word "analyze" as an example. The word "analyze." And at home that word is not in use. It's not being used.

And so what's happening is, kids are opening these textbooks which were written by well-meaning educating authors. But they're fundamentally using words like "process" and "analyze," which there is no context for, unfortunately, in the mind of the reader. So the message is not getting through. And there was a sense of his own frustration that there wasn't enough educational substrate from the home bringing kids into the school, so that it was difficult for them to get any traction with the material that they were being offered. So sort of this concern about the dumbing down of the populace.

Leo: Right. Well, certainly always something to pay attention to, that's for sure.

Steve: And not a problem we have here with our listeners.

Leo: No. In fact, this show is challenging for almost everybody who listens. But that's - you know, your brain will grow if you listen. So today's a Q&A day.

Steve: Yup.

Leo: 214, it's even, mod 2. So we have questions. Do we have any security news or updates?

Steve: Got a bunch of stuff, yeah. Remember that last week, just as we were recording, because we were recording a day early in order to move the recording out of the way of the big Mac event, the iPod fiesta, that we knew that this was - that was going to be the second Tuesday of the month, and Microsoft was releasing things. While we were recording you had looked them up because they had been published just as we were going live and recording. And the thing that I stuck on among those that you enumerated was a bad, apparently bad TCP/IP flaw that was going to be fixed.

Leo: Right.

Steve: Well, it turns out that this was something we have covered in the past, that surfaced last year when it was sort of repopularized, although it was originally discovered in '05, so four years ago. And that was the Sockstress problem, the idea that there was a denial of service attack, generally powerful, that required low bandwidth, not a flooding, just completely overwhelm some spot of the Internet with huge amount of inbound traffic, but rather this was taking advantage of some of the inherent proper functioning of the TCP protocol where there's something called the TCP window, which is sent, or as they use the term, "advertised," whenever one end is acknowledging the receipt of data to the other. There's this TCP window which says, oh, and by the way, this is how much buffer space I currently have that allows the other end to asynchronously send ahead. That is, it's one of the ways that TCP so gracefully and nicely deals with the delay of packet traveling across the Internet. If it was necessary for each end to acknowledge the receipt of every single packet, then the round trip time would limit the amount of data, the rate of data that you could send. Because if one end sent a packet and then waited for confirmation of its receipt, then that would obviously be a big problem.

So the designers who understood this, this whole notion of packets moving from router to router between end points, they said, okay, we need a way of allowing us to send ahead, to like know how much we can send and not need everything acknowledged, not need to wait for this roundtrip time from the destination and back to the source. So they created this notion of a window where the recipient is constantly saying, okay, here's how much buffer space I've got. So the way it works is, it's a guarantee of how much can be sent at the time that that window is received by the sender. The sender looks and says, oh, okay, good. The other end is claiming that he's got 16K, so I can send - I know I can send at least that much safely without there being any problem.

Well, what happens is, if the other end says, oh, hold on a second, I have no space, I'm full right now, then the sender is blocked from sending anything. And then if no additional change occurs, the sender will periodically send what's called a "window probe," send an acknowledgment to traffic that's already been received. And that induces the other end to acknowledge that and in the process get an update on the window status, hopefully finding that some new buffer space has been made available. So this is fundamental to the way TCP has always worked. And it's something that just sort of wasn't really on anyone's radar until this notion - it was really just sort of a repopularization of an old problem, the whole Sockstress that we talked about, about a year ago.

So it turns out that that's what Microsoft just fixed. And they didn't fix it on their older OSes. Their claim is that the stack, the TCP/IP protocol stack, as the term is used, in Windows 2000 is too old. It doesn't have the required flexibility. And in fact that's also the case, I believe, on XP. So it's not until Windows 2003, Vista, 2008 Server, and Windows 7 that they've got whatever it is that they're doing in those stacks. My sense is that Microsoft just figured, well, it's old. Here we are in 2009. We're not going to worry about 2000 anymore. And so they did not fix that. There is no fix for that. In order to fix it, you would need some sort of third-party firewall box or something on the outside of the server essentially being a prophylactic to protect the server against this kind of attack.

So Microsoft fixed it. Cisco also fixed it. And this is still an outstanding problem. Until, I mean, it's tricky to deal with because you need some sort of an overseer because it's legal for the other end to say I've got no buffer space. So you just can't drop a connection that says that. You need to detect that there's malicious intent, which you would detect by a succession of connections from the same IP, all saying that they've got no buffer space. And at some point, in sort of an overseer mode, you'd look at all of the connections you had coming into your server and say, wait a minute, something looks fishy here. And then you would proactively drop those connections and probably blacklist that IP so that it was no longer allowed to obtain any connections to the server. Or maybe just terminate the oldest ones and allow it to keep making new ones, which would prevent there from ever being a buildup. So that was the issue that came to my attention when you enumerated the problems that were fixed in last week's update.

Leo: Okay. All right.

Steve: Apple, we talked about Snow Leopard v10.6 and how it was downgrading the version of Flash Player, Adobe's Flash Player, that it installed. And you probably know the Snow Leopard has been updated almost immediately to 10.6.1 to incorporate the latest version of Adobe's Flash Player. So Apple responded very quickly to that. It was obviously an easy thing for them to do and something that was likely an oversight or - it certainly generated a lot of press. And so they responded to it immediately.

Firefox has also been updated. It's now - the 3.5 version thread is now at 3.5.3. And the 3.0 version thread is at 3.0.14. So any Firefox users, you may want to just check to make sure that, you know, just check for any updates, and Firefox will let you know. And it's just, you know, your standard security fixes and stability improvements.

I did move a couple of my machines, Leo, from 3.0, where I had been sort of stubbornly staying, over to 3.5 because a couple times Firefox was saying, hey, we've got something really new here, and what's wrong with you? I thought, okay, fine, I'll give it a try. And it's working just fine. And I did discover that when I reinstalled the various add-ons that I like to use, with a little bit of jiggering I was able to get them to work under 3.5. So I'm able to use 3.5 and have the various add-ons that I like.

Leo: Yeah. Yeah, I love 3.5.

Steve: And something interesting happened. I don't know if you had picked up the news about a major incursion into Apache Software Foundation.

Leo: Uh-oh.

Steve: Yeah. This is...

Leo: Everybody uses this for their web server.

Steve: Well, yeah, it's not that - it's not the web server itself. It was their network got hacked. The Apache Software Foundation network got hacked. What happened was, the exploit that we talked about several months ago in Linux, the kernel-level privilege escalation, privilege elevation exploit, that was a local root exploit, was used on - and this is really interesting, I mean, the way this attack happened. It was used against sort of a off-the-mainstream, sort of ancillary server, ApacheCon.com, A-p-a-c-h-e-C-o-n, which is the server for the Apache Software Foundation Conference. And it was DV35.ApacheCon.com that was running an unpatched version of the CentOS, Linux CentOS, which still had that vulnerability that had never been patched.

So some bad guys were able to get into that. They fully compromised the machine, got root, destroyed the logs, which meant that for the Apache guys, figuring out what had happened exactly was challenging. Then they used an SSH key which belonged to the backup account to gain access to the main server, which was People.Apache.org. And so there was a, sort of on the subsidiary machine was a - it had backup privileges on this main machine that allowed them to get the SSH key. And the machine that they got it to was what Apache calls a "staging server."

So what happens is there's like a staging server for their software, and then a regularly scheduled rsync process copies the staging server to the production server. So they were able to compromise the staging server and install CGI scripts, which they added to the document root folders. And then when rsync happened, even though they had no access to it, that put all of the changes that they had made to the staging server onto the main Apache.org public server. And the CGIs allowed the bad guys to obtain remote shells.

So the Apache Software Foundation guys said, you know, our software, our source code is fine. We've determined the limits of this incursion. And they were very open about it. In fact, the security community has congratulated them with being so open about these are the things that we did right; these were the things that we did wrong. Interestingly, one of the things that they felt - that turned out coincidentally to be right is that they do not have a homogeneous server farm. They have a heterogeneous server farm. They've got Linux, they've got Sun, they've got...

Leo: And that's a good thing; right?

Steve: They've got Free, yes, they have FreeBSD.

Leo: It's not a monoculture, yeah.

Steve: Exactly. And so it was the fact that they had a FreeBSD 7 system that did not have this problem, and the Sun didn't. Only this one sort of off-to-the-side Linux machine did, even though there still was a route in. And I think for our listeners that's sort of the coolest thing is that - well, I mean, cool in an unfortunate fashion - is that even though you get a little bit of foothold off on some machine on the side, once you're there, then that machine on the side may have some sort of privileges or view into some other machine. And little by little you sort of gain entry into the main network, which is exactly how this happened.

So the Apache guys, I think, are to be complemented for saying, okay, this is what happened. This is how it was done. And so what they did was, one by one, they actually went completely offline. They brought their systems down. They did have mirror servers, so they were able to bring up sort of a backup presence while they took a good, long, hard look at all of the machines that might have been infected. And then one by one, once they were sure that those machines were clean, they brought them back up online and learned some valuable lessons about how to be better safe in the event that one machine in their network gets compromised.

And, for example, they use these CGI scripts to dynamically generate the web pages that people visit. And what they found in looking closely was that they had the ExecCGI enabled globally, even though it was not being used globally. And had, for example, it only been enabled where it was needed, this exploit would not have been possible. So...

Leo: That's a common error on a lot of web servers, actually.

Steve: Yeah. Again, it's sort of - it's a little bit like the old approach that firewalls took of being open, and then selectively closing problems, instead of the other way, which the modern approach is that you close everything, and then you selectively open only those services and ports that you know you need. So I thought that was interesting. Also The New York Times got hit with a JavaScript-based scareware attack.

Leo: Yeah, I saw this. Oof.

Steve: Yup. And it's interesting because it's a problem that, again, was foreseeable. The New York Times, like many other websites, are serving ads from a third-party server. We've talked about this often. It's the way cookies are used to track people is the much-maligned DoubleClick.net company are an active server of ads. Well, if bad guys are allowed to submit ads to third parties, those then get served to some public server. And unfortunately it's possible to have scripting in an ad, of all things. And that was then - it was scaring people, popping up a window, redirecting them to some other server that popped up a window and said, oh, you've got some bad stuff on your system. It's infected with malware. Click here to scan, and we'll take care of you. And of course it was 100 percent malicious. And so The New York Times is going to be modifying their own behavior and somehow coming up with some way to control the ads, which they obviously didn't have before.

Leo: It was interesting because the people who were doing the malware posed as - I believe it was Vonage. The New York Times won't say, but the reports are they were posing as a legitimate company, and I believe it was Vonage, for a whole week, putting what would appear to be normal-looking ads up for a whole week and then...

Steve: Oh, to establish a baseline.

Leo: Exactly. So they were paying good money out for ads for something they didn't even own. And then the weekend hits, when I guess they figure, well, nobody - there'd be less oversight on the weekend.

Steve: Ah.

Leo: That's when they stuck it in. It's very interesting. And, you know, I talked about it on the radio show because on Sunday the Times put up this warning. And it was still happening when they put up the warning. It took them a while to get rid of it. So a very, yeah, sneaky - these guys are sneaky.

Steve: Yeah. I mean, we can count on that in the future.

Leo: Well, you remember it happened at MySpace. There was malware put in one of the rotating ad banners because a lot of these guys have automated ad systems running, but it can submit an ad; you know?

Steve: Exactly. So you just post the ad, and it's automatically accepted, and it goes into a bin and is rotated through all the websites that are clients of that provider.

Leo: It's pretty obviously a bad idea.

Steve: Bad idea. Not secure.

Leo: No.

Steve: Adobe has announced that they're going to be delaying their quarterly update from the 8th of September to October 13th. So of course that passed by already. But they would have normally done it on the 8th, and they're going to be moving it to October 13th because this Microsoft Active Template Library, that ATL problem which has been so pervasive and has bitten so many people, has also bitten Adobe. And so we will be seeing an update to Reader and Acrobat and, I would imagine, Flash come next month. But it won't be Adobe's regularly scheduled quarterly update. And you know how I feel about that. I mean, it just seems so ridiculous that that's the approach...

Leo: Quarterly is not often enough.

Steve: No. And they have - they've already had several emergency updates. We talked about how, well, that didn't even last a quarter.

Leo: The minute they announced it, pretty much.

Steve: Yeah, yeah. And in something a little creepy, there was another story that came across my radar. There's some web-monitoring software which is available at retail under the brand Sentry and FamilySafe. It's produced by a company called EchoMetrix. Deborah Yao, reporting for the Associated Press, learned that this company, who makes this consumer retail web-monitoring software, is reading the private chats of the children that are being filtered by this...

Leo: Terrible.

Steve: ...in Yahoo!, MSN, AOL, and other services, and selling the information from that to third parties.

Leo: Oh, my goodness.

Steve: So, and in recognition that there are federal privacy laws to protect anyone under the age of 13, in the fine print it says that in recognition of federal privacy laws, data on kids, on children under 13, it says the agreement states that the company has, quote, "a parent's permission to share the information" if the user is a child under age 13. Which just - several of the people who responded to this said that this one of the creepiest things they had seen in a long time.

And there is no mention of this in the licensing agreement. The agreement states that the company reserves the right to pass along data to, quote, "trusted partners," unquote, and that the confidentiality agreements with those partners prohibits them from sharing the information with others. But still, they are selling this information to third parties. And there's nothing in the relationship that the typical end-using consumer has with EchoMetrix, this company selling Sentry and FamilySafe, makes that clear. There's some provision to opt out if you go to their website, but not in the UI of the software at all.

Leo: Ugh.

Steve: Creepy.

Leo: Yeah. Oh, that's just appalling. Man, I think of how the parents must feel on that. That's terrible.

Steve: Well, hope the word gets around because it's the wrong thing to do. I did have sort of an interesting little SpinRite story that I thought people would get a kick out of, also because this person who asked to remain anonymous works in the cancer center of a major university medical center. And I guess he first tweeted because his note was, "Just thought I would retweet what I sent out today on Twitter." And so I guess what he said on Twitter was, "Computer gods bless SpinRite. Boss's PC fixed."

Leo: Yay.

Steve: "Into which volcano must I throw a virgin? Where do I find a virgin?" And I guess that used up his 140 characters.

Leo: That's 140 characters. But a good use of those 140 characters.

Steve: And so then he said the story beyond 140 characters continues. He said, "Steve, yesterday while the boss was out of town I stopped in his office to do the weekly check on his PC, make sure his mailbox was not filling up, run the monthly Windows patch push tested and packaged by our IS staff ahead of time, but refused as an automatic push by my boss who, like me, is old enough to remember these pushes causing problems." He said, "It's been years, but he's the boss. I ran the push, and the computer locked up. Doing a Ctrl-Alt-Del forced a reboot, and it went into Blue Screen of Death." And so he says, "A+ certified I may be. My main job is systems management, not PC maintenance. And more to the point, especially not on my boss's PC. He keeps all his documents and stuff on the hard drive." And says, parens, "(He is old and remembers...."

Leo: Hey, I resent that.

Steve: "(He is old and remembers server crashes, too.)" Apparently where he lost all the stuff that he had on the server.

Leo: Yeah, I remember that, too.

Steve: Yeah, I remember that.

Leo: I remember service crashes.

Steve: And he says, "More terrifyingly, I realized he had turned off the remote backup I had installed for him, apparently because it slows down his PC. I called in the institutional techs, but none of their full-time professional diagnostic repair tools and tricks could get around the Blue Screen of Death. We also did all the standard ones - last known good boot and last known good configuration - and could not even reinstall the OS because it would not recognize the existing XP SP3. We are a, quote, 'Reimage first, ask questions later shop here,' except in cases...."

Leo: Which is probably a good policy. Except for data.

Steve: Yeah. Except, exactly, except - and he says, "...excepting cases like this, so this was not surprising. To my relief, we were able to slave the drive to another machine, so I was able to get his data onto another hard drive. To be fair to Microsoft," he also noted that "some of the motherboard's capacitors looked bad, and that might have been the real source of the problem." Sounds like he changed tense there on me, but anyway. So he said, "Either way, I still faced having my boss come home the next day with no PC. I got out a laptop and began very roughly configuring it for him, knowing it would lack most of his specialized software. But as the tech finished his last effort, I remembered the copy of SpinRite I had purchased for myself." He says, "I've used it one time when it was unable to help a very dead drive, and another time when it saved the day, but mainly use it for my home preventative maintenance. It was time to go home, the end of the day. So I popped in my copy of SpinRite and left it saying it was going to take seven hours to run. I went home worrying if I was going to have a job the next day."

Leo: Oh, boy.

Steve: "Set my alarm early..."

Leo: Oh, poor guy.

Steve: "...to beat the boss in to work, and literally dreamed about the problem several times."

Leo: Oh.

Steve: He said, parens "(All the dreams could be called nightmares.) When I arrived, SpinRite said that it had completed. There was one unrecoverable sector. I booted and was amazed to see everything back to normal and running. Needless to say, due to the leaky capacitors, I am trying to get him a new PC before anything else happens, and I am backing up until it is safely installed." And then what I love was his final little note. He said - remember that he works for a cancer center in a major university medical center. He said, "P.S.: Everyone here is as crazy about Vitamin D as you are."

Leo: Really.

Steve: And then he said, "Withhold name if used, please."

Leo: I wonder why. Do they make Vitamin D, do you think?

Steve: No, because it's such a strong anti-cancer agent.

Leo: Oh, and they're anti- of course.

Steve: They're a cancer center.

Leo: Cancer center, yeah.

Steve: Yup.

Leo: So they know. That's very interesting. By the way, we've been looking at ratings. And the ratings for that Vitamin D episode were about, let's see, almost double. So I don't know whether - what I suspect has happened is that people who listen to Security Now! on a regular basis shared it with people who don't listen to Security Now!, but might be interested in Vitamin D. Because it just - the numbers on that thing were through the roof.

Steve: You know, Leo, we both thought I was a little crazed when I did the Q&A the week after, all about Vitamin D. But I was in this, sort of this fog because I got an unbelievable amount of feedback. I mean, it was just...

Leo: Yeah. You were on the right track, absolutely.

Steve: It was crazy, you know, so...

Leo: I mean, it's not something we want to do every...

Steve: No.

Leo: It's not a vitamin show. But at the same time I think, you know, everybody trusts you. They know that you're very level-headed. You're not a faddist. And you do the research. So when you were saying all this stuff, I think people shared that, they must have, like crazy.

Steve: Yeah. Well, and I had hoped to bring the same level of scientific pursuit to it that I normally exercise in my own life. And anyway, so it was valuable, and I was really glad we did it.

Leo: Well done. Shall we move to the questions and answers? Because I have some good ones for you.

Steve: Yes, you do. I happen to know.

Leo: You picked them, yes. No surprise to you, Mr. Gibson. Starting with question one, an anonymous GSM provider - hmm, phone company - in the UK, in England. We did an episode, of course, last week on GSM cracking. He or she responded: Hi, Steve. I just listened to your podcast on GSM cracking. I work for a 3G operator, and we are very - underscored, capitalized, and bolded - aware and wary of the issues with GSM's 2G vulnerabilities. I thought your assessment was mostly fair. So here are a couple of additional points:

Data GPRS - the packet radio service that kind of predates 3G data - doesn't use A5/1 - or as they say in the UK, A5 stroke 1 - it uses GEA/1. And yes, this is similar in structure to A5/1, but this would require a different rainbow table computation. Still vulnerable to rainbow tables, I guess.

Steve: Yup.

Leo: The design of GSM security was to give a similar level of privacy as is provided by the wired network. Where does that sound familiar? Oh, yeah, Wired Encryption Protocol.

Steve: Uh-huh. Doesn't that ring bells.

Leo: Oh, that rings a bell. So your attack on a competitor has been achievable in the past. All you'd have to do is open the manhole outside the building and tap into the analog wires, or the T1 - or E1 as they call it in Europe. Okay. Okay. That's, you know, they're setting a standard. A low standard, admittedly, but they're setting a standard.

Fixing the problem: GSM was designed with the ability to add new algorithms, so a total of seven algorithms are possible. About three years ago the GSM Association published A5/3 and has been moving this forward, albeit at a frustratingly slow pace. In our defense, it's hard to get 400-plus operators and many phone manufacturers to spend money on some theoretical threat. That's reasonable. That's really what's difficult about all of this is legacy hardware. We even, I think, addressed that.

Even with the current publicity, there will be operators, most operators probably, who will not implement A5/3 as this will cost them money. It's hard enough to convince the big European operators to spend money on implementing this. Happy for you to read this on your podcast provided you don't mention my name. And we did not.

Steve: Yeah, so this is, I mean, this is the problem we have is here's a direct message from a listener who is with a GSM provider, who acknowledges that, while more security is available, it is expensive for systems to be fit with it, to be retrofitted. Consumers' devices have to be upgraded at the same time, synchronously. He also, I loved how he talked about how, well, it's like Wired Equivalent Privacy. We only designed it to be as secure as the wire would be. And as we all know, you could tap wires if you wanted to. So this really wasn't meant to be super-encrypted security anyway. Just hard enough that, unless you were a government, you couldn't listen in on people's phone calls.

And it turns out, as we know, as technology has moved, because this is now so old, it's become increasingly feasible for people to have a few terabytes of rainbow tables built and stored, which suddenly makes the cryptography substantially weaker. And other James Bond-like sci-fi technology, like having a software-programmable radio, well, you can order those on the Internet now and download the software to make it work. So it's just become too accessible.

Leo: Right.

Steve: And as he says, moving forward, the problem is this is probably enough encryption for most people. And remember that when analog cell phones weren't encrypted at all, people still used them. I mean, they used them like crazy.

Leo: Yeah. Yeah. I remember people, well, we talked about it last week. I remember people just having scanners and listening in; you know?

Steve: Yes, yes. I have done it. I mean, it just - you turn the scanner on, and you're listening to someone's phone conversations, like whoa, that's a little too easy, yeah.

Leo: Austin Clark in Menomonee Falls, Wisconsin wants cookie management. Who doesn't? Steve, for the second week you mentioned your favorite Firefox cookie manager, but never gave us its name. I'm sure you're driving a number of your listeners like me crazy. Could you either tell everyone on the next show; or, if you don't want to promote it, could you at least email me the name? It could be our little secret. What is the name?

Steve: Okay. The name is just - it's just called Permit Cookies.

Leo: Permit Cookies.

Steve: Permit Cookies. And I need to warn people that it is the most feature-lean cookie manager there is.

Leo: Much like the name.

Steve: It's why I like it. In fact, some listeners recommended other cookie managers, and I thought, oh, okay, I'll see. And they're just - they've got more bells and whistles than I want. The way I have Firefox configured is I have it set to allow third-party cookies just for the sake of not breaking anything, but to remove all cookies whenever the browser session restarts, which is one of the options in the standard - under the privacy tab in Firefox. So no cookies, either first-party or third-party, are ever kept permanently. But then that's, of course, inconvenient because you'd like some sites to remember you. I don't want to have to go reauthenticate to Amazon every time, or eBay, or PayPal, or so forth. So it's nice if specific sites that I trust are allowed to create permanent cookies.

And so what I use, I use this Permit Cookies, little add-on. It just puts a little tiny little "C," C as in cookie, down in the Firefox tray. And when I'm at a site that I want to remember me, it's just a matter of right-clicking on that and saying trust this site. And so it's a simple UI into an existing dialogue in Firefox. That is, I could go into Tools, Options, Privacy, and make that site an exception. But that's many more steps. The other thing is that the little "C" will turn green if the site I am on is in my exceptions list. So it's easy for me to see, oh, yeah, okay, that's - I'm trusting this site. Typically, most of the sites I go to, I'm not. So everything works fine while I'm there. And as long as I'm using the browser. But there's no long-term accumulation that's like this infinite accumulation of cookies that you normally have since I use this delete them or keep them only for the current session option.

Now, Permit Cookies, if you put Permit Cookies into the add-on finder, it will - it's not in the first one that comes up. You have to say "show me all." Then it's on a page on the Mozilla site. And it says that it is not compatible with 3.5. And so it just - you can't even install it from there. But in fact it is. And if you go to the author's website, and there's a link there, he's got a whole bunch of little add-ons that he's written for Firefox. And a ways down is just this little simple Permit Cookies. So it's as simple and easy to use as could be. And I did come back to it from the more fancy cookie managers because there's really nothing I want to do more than that.

I just want to say, you know, me, I'm the lowest common denominator guy, writing things in Assembly language and wanting it to be simple and clean and not slow down my Firefox and not require infinite updates. This thing's never been updated. Well, I mean, it's at 0.6.2, I think, is the version of it. But it's not like NoScript. Every time I restart Firefox, oh, we've got a new version for you. It's like, okay, wonderful.

So that's the thing I use. I really like it. I recommend it. It's minimal, minimal, minimal. But it just - it allows my Firefox to remember the sites that I want to. Otherwise cookies are all sort of session cookies, even the first-party cookies, that are constantly flushed whenever I restart Firefox. So works great for me.

Leo: So just to recap, you turn off cookies completely.

Steve: No, no, no. I...

Leo: You block all cookies and then use this to unblock the trusted sites. Is that - did I misunderstand you?

Steve: No.

Leo: Oh, okay.

Steve: Yeah, Firefox has a neat option. I'm going to go under the Tools menu to Options. And you click the Privacy tab.

Leo: Okay.

Steve: And under Cookies I have enabled "Accept cookies from sites," and I have enabled "Accept third-party cookies." But the next option down is "Keep until." And so the normal option is "Keep until they expire." And I've chosen "Keep until I close Firefox."

Leo: Okay.

Steve: And so the beauty of that is nothing is stored permanently on this system. But over to the right there's an exceptions button. And if I look at my exceptions, I've got Amazon, Blackberry, eBay, GRC, UPS, and a couple others.

Leo: But those were added by the plug-in.

Steve: Yes.

Leo: Got it.

Steve: Now, you can add them manually, if you didn't want to use the little plug-in. But it takes just more steps.

Leo: So the plug-in explains this, how to do this? I mean, this is the way you're supposed to do it?

Steve: No. That's why I...

Leo: Oh, good. I'll put this in the show notes, in that case.

Steve: The plug-in is so generic, all the plug-in does is simply allow you - see, it knows what site you're on. So you're just able to say block or allow the current site. Just that simple. And so it just sort of makes it easier to put domain exceptions into Firefox. But in order to make the rest of this work, you have to go and configure it the way I have.

Leo: Got it.

Steve: Which is to say, allow first-party and third-party, but then set them to keep them until I close Firefox.

Leo: So that's interesting. So you've told us in the past to disable third-party cookies.

Steve: Yeah. And, for example, on IE, that doesn't have an option like this, I think that still makes more sense. And I could disable third-party cookies except that here there's no long-term tracking happening because they're all being washed away whenever I close Firefox.

Leo: Right. I'm looking at my Firefox settings.

Steve: But look at your cookies. If I look at my - oh, I'm sorry. Look at your exceptions. I have, like, 10. There's only 10 sites that I want to remember me on an ongoing basis. And they're able to. Nothing else can. Everything else just gets thrown away.

Leo: I like this. This seems like the best of all worlds, best of all possible worlds for cookies.

Steve: I think it's very clean. And...

Leo: And stuff works while you're browsing.

Steve: Right.

Leo: But it just doesn't remember anything about your previous session once you've closed it unless you explicitly say, "I want this one."

Steve: Precisely. So it's exactly what you want. It's an opt-in approach which is extremely lightweight.

Leo: Now, it looks like, unfortunately, Firefox has changed their setup in 3.5 so that this - you can't do this anymore. At least I can't figure out how to do it.

Steve: Right, I'm still in - on the one I was looking at, I'm at 3.0.14.

Leo: Because I go to Privacy, and it's all about history now. And, boy, they've really messed this up. Gosh darn them.

Steve: I do have 3.5 running on a couple machines. I'll take a look at it and make a note to see if I can see a way to do the same thing for next week.

Leo: And I'll put this in the show notes. We'll put this in the wiki. And as always, what we do - this is a new thing that we do is I have a FriendFeed room that I start for every show. It's on FriendFeed.com/twit-conversations. And as we talk, I put links in here so that I have links. So if you want - if you're listening live, and you want to get that information, it's there. But we'll also take that information and then put it into the - we'll put it into the wiki. And I'm sure you'll put it in your show notes, as well. So people can get the show notes at GRC.com, as well.

Okay. Moving right along. Mateus Del Bianco in Brazil wonders about GSM cloning: Hi, Steve. Listening to the podcast, the last episode on GSM cracking, got me wondering, how easy is it to clone someone's cell phone over the air? Now, cell phone cloning was a technique used for a long time, I don't know if it was pre-GSM, but as a way to kind of, you know, steal their phone, in effect. I can accept that GSM is secure, and I don't mind someone listening to my conversations, but I do mind if someone can use my line. Is GSM cloning possible? What about if someone has physical access to the SIM card? How hard is it to obtain that 128-bit preshared key you talked about? All good questions.

Steve: Well, I didn't - the reason I wanted to share Mateus's question is, it is absolutely one of the consequences of the cracking of GSM is the ability to clone. It was given in several of the examples of the papers that I read when I was doing the research on this. If you have physical access to the SIM card, it's relatively trivial, using current technology, to crack the algorithm. There's something, it's called COMP128, which is the authentication algorithm, which by default has been used by most providers. It was given as an example in the original GSM spec. And while not everyone has to use it, that's what everyone has ended up using because it was just sort of, here, here's an example of an authentication algorithm that you can use with GSM. And everyone said oh, okay, fine, we'll use it.

The problem is, it's very old, and it has been badly cracked by cryptographers who understand the weakness of it. So it is absolutely possible for somebody with a radio to basically ping somebody's phone. And they don't even know who they are. So you would, like, for example, you're in a coffee shop, and this person with a laptop and a strange-looking antenna and a little box to the side of their laptop has a strange grin on their face because they're pretending to be cell towers to every phone within range and are able to ping the phone and acquire the shared secret that is that 128-bit key that the subscriber has locked up in their SIM card, and after that be able to impersonate that person's phone.

Leo: Wow. It's easy.

Steve: So it absolutely is one of the consequences of this cracking that we're talking about.

Leo: Wow. And to go back, thanks to our chatroom, to go back to our question about Firefox...

Steve: Oh, Firefox?

Leo: It turns out that, if you go to the Privacy section, it looks like it's a very simple section. Firefox will remember history, never remember history. And then there's a use custom settings for history that gives you access to all of those previous settings that we've seen before - accept cookies from site, accept third-party cookies. And then you just change this "Keep until" to "I close Firefox." And that's all you do.

Steve: Beautiful.

Leo: Yeah. And then there's also a clear history when Firefox closes. But that's kind of separate from cookies; right?

Steve: Yes, yes.

Leo: We don't have to worry about that. Thank you to the chatroom. Once again, they're good. They're real good.

Steve: And just to reiterate, although nowhere does it say, even on the author's site, that Permit Cookies does run under 3.5, I am running it under 3.5, and it works beautifully.

Leo: Great.

Steve: So it just - there's no problem with it.

Leo: Excellent. Moving to question four, Dax Mars, which is a great name, it sounds like a science fiction name, visiting earth via second life - I guess it is a science fiction name. Quick question: Would I be insane to try running my own web server for my website? Cash is short. My hosting is up for renewal. My ISP's personal web space is very limited and not very reliable, and I'd rather spend the money elsewhere. I'm thinking Windows 2000 or XP on an old PC with Apache for Windows. What do you think, Steve?

Steve: I thought this was a really great question because it incorporates this issue of whether things have just become so crazy on the Internet that it is impossible for just a private citizen, an individual who doesn't have a huge IT staff and security people and all of the paraphernalia that any large organization will that wants to have an Internet presence, is it possible for just a random guy to set up a website and have it be practical? And obviously he's asking us because security of that is an issue.

Leo: Right, right.

Steve: And it really is a good question. I would shy away from Windows, that is, he's suggesting Win2K or XP on an old PC, using Apache for Windows. If I were doing this, I would use the securest version of UNIX available. I would use NetBSD or FreeBSD. And you can run Apache on that. There's a little bit of a learning curve. But either of those runs beautifully on the oldest PC you can find. It is just simple to do. And they're going to be state of the art. They're going to be very secure. And Apache is Apache, whether it's on Windows or on UNIX. If you also install the SMB support under UNIX, then it's very easy to look at your file system over on the UNIX machine from within the Windows browser, that is, you're able to - it's just another machine on your Windows network where you're able to open it up and look at it. It's the way I manage my UNIX machines, is they have SMB running on them. And that allows me to see the entire drive. Just like I'm using file and printer sharing, where I'm looking at other drives on my own internal network, I'm looking at UNIX that way. And of course we configure UNIX with text files. So it's very simple and practical.

The only problem, of course, is that you've got now a server deliberately exposed to the Internet. And the danger is that a bad guy could somehow use some compromise on the server in order to get root on the server, gain access beyond just being a casual web surfer, and then have access to your internal network. So this is where my suggestion of multiple routers comes in, the idea being to put the server on an upstream router and then protect your network behind a router which is located inside the network that the server's on, very much - it's not as much like the consumer grade of DMZ, where anything coming in goes to that IP. You really don't want that. You only want to allow web services to be mapped to the IP of that server.

But we've talked about using multiple NAT routers several times. And I would use that approach. And then I think you're probably pretty safe. I mean, it's sad that in this day and age it's just not easy to put a website up on the Internet and be able to do things. But the fact is, I mean, look at all of the problems we're continually seeing with this stuff. It really is difficult.

And the other little bit of advice I would have is absolutely, as much as possible, resist the temptation to make it more complex than it has to be. If you don't need SQL server, don't put it in. If you don't need PHP or any of the fancy scripting technologies, don't put them in. Those tend to be where today's problems are, more than just in the core web services. The core web services stuff that just serves up simple web pages are pretty solid and pretty stable now. It's all that extra fancy stuff that people are adding that we tend to see the leverages that allow people to gain a foothold inside a server.

So I think, yes, I think doing it carefully, minimizing your install - arranging to isolate the server's network from the rest of your network I would absolutely do. I would never have the server on the same network as mine. It makes administration of it a little more painful. But most people aren't changing the pages on their server all the time anyway. And then I think, you know, basically you've got a web server for free. No web hosting, no ISP or anything.

You do want to make sure that your ISP allows you to host a server. That is, you want to make sure that they're not blocking port 80. We know that many ISPs are now blocking the file-sharing ports, those that Windows use. Some are blocking port 25 to prevent spam from illicit SMTP servers on port 25. You do want to make sure that 40, I mean, sorry, port 80 and 443 are enabled if you want - 443 if you wanted to do SSL connections, but at least port 80. You can run a server, of course, on an alternative port like 8080. But that's not convenient for people because that's not what their web browser is going to use by default. So I would say yes. It's not easy. It's unfortunate that it's not easy. But it's certainly doable.

Leo: Yeah, and there might be - it's now all of a sudden, because you're running a server, you have to kind of keep an eye on holes and exploits and make sure you're patching it regularly. And I don't know - Windows now becomes a vulnerable target, as well. A lot of people would prefer FreeBSD or something more secure than Windows.

Steve: And isn't it sad, Leo, that, I mean, it's just not easy for someone to run a web server. Like...

Leo: Yes.

Steve: Like once upon a time.

Leo: Yeah, I mean, we used to do that. Anytime you run a server, it's not just the web. An FTP server, anytime you run a server, even a Windows Media server, you're always kind of now opening up a little vulnerability. I like the idea of isolating it using the routers. You also may have issues with your - if you don't have a static IP address. And then you have to use things like DynDNS to redirect because otherwise people don't know where to go if your address changes regularly. There are all sorts of little issues.

Steve: Yeah, in order to publish a domain name whose IP will change as your machine's IP changes, to the degree that it might.

Leo: Yeah. Question five, Tim in Rancho Cucamonga - oh, I love saying that - Rancho Cucamonga, California wonders about a router's password strength: Hi, Steve. I hear a lot about having a strong wireless passphrase, but what about the password that lets you into the router setup? The same password that will let you see the wireless passphrase unencrypted. If I use your Perfect Password maker for my wireless passphrase, then a relatively weak password to get into the router, isn't the router's less robust password the weak link? Or am I missing something? Should I use a Perfect Password generated by your site for the router setup, as well? Thanks, Steve. Great show.

Steve: I thought that was a great question because we've talked a little bit - we've talked extensively about the only current vulnerability known for the strongest WiFi is guessing the password. That is, there is still that vulnerability. That is the only problem that we currently know, for example, with WPA encryption, which is not using TKIP as its cipher, but is using AES, which is what you want to use when you can. Sometimes that's called WPA2, although that's really not the official name. But given that you're using the best wireless encryption available, the only known vulnerability is just guessing the password. And that is an offline attack, meaning that data can be captured and then taken home somewhere and pounded on by as much technology as is available, trying every possible password.

Leo: But don't they need physical access to the router to do that?

Steve: No, no, no. No, here I'm talking about just about cracking the WPA.

Leo: Oh, the WPA, yeah, right, right.

Steve: Yeah, the WPA password in the air. So we know that, and we've talked extensively about, the fact that that wants to be as unguessable as possible. So Tim is exactly right, that if you had an attack on the router's password, that is, the administration password, username and password for the router, then if that were dramatically weaker, and if you had access to it - and that's the point you were making, Leo - then there'd be a problem.

So we know there's the issue of WAN-side management. And one of the first things you want to do is make sure that your router is not manageable from the WAN, from the Wide Area Network, that is, from the outside, from the Internet. I don't think in this day and age there are still routers that ship with that enabled by default. I sure hope not. It's just incredibly worrisome and insecure to expose the management interface, even behind a username and password. I just can't think of a good reason to do it unless you really, really need to administer routers remotely over the Internet, in which case there it's extremely important that you use a strong username and password because nothing is restricting someone from just sitting on that connection and guessing username and password day in and day out until - hopefully they're never able to guess it. But when they do, they'd be able to log in.

But assuming that web-side management is not enabled, then the only vulnerability would be somehow accessing what is now typically a web browser interface from inside the LAN, which is the point you're making, Leo, is how would a bad guy get onto the LAN? And the good news is, unless there's malicious software running on a machine, it's probably not possible. You've got the catch-22 of the wireless aspect of getting on the LAN, which is if you had the wireless password, then that would get you on the LAN, giving you an opportunity to break the router's administration password. But assuming a strong WiFi password, you don't have that password until you break the router's administrative password, that as Tim said would give you in-the-clear access to the WiFi password.

So the danger - and this is something we have talked about before - is, like, for example, leaving the router's admin passwords alone. We now know there's malware that you can get on your system that is smart enough, they contain all the username and passwords for all the routers out there. And it will attempt to log into your router explicitly for this purpose. In this case it's not trying to steal your WiFi password. Normally it's trying to access your router in order to open ports in order to allow remote access into your network. So we know that it's important that you change your admin and username away from the manufacturer's default because there's definitely malware roaming around the world that knows, if it can get into your computer, it would love to take over your router. And that's the first step to doing so.

So Tim's right. It's not a huge problem because it's not something anyone - no one has access to your admin username and password, given that you've got WAN access turned off, the Wide Area Network access turned off. So they don't have access to your network as long as your WiFi password is good, and we're presuming it's good, and we're looking for the weakest link. But it's certainly worth making it as robust as you can. And certainly use another one of the Perfect Passwords from GRC. That's going to give you complete pseudorandom protection.

Leo: I've seen a number of routers that have, as you call it, WAN administration turned on by default. Which is shocking.

Steve: No kidding, yeah.

Leo: And we talked before, and I think I talk about this on the radio show a lot about the things that you need to do when you get a new router - change the default name, change the default password, turn off WAN administration...

Steve: Oh, of course.

Leo: ...turn off Universal Plug & Play.

Steve: Yes.

Leo: I think those are the - oh, and turn on, if it's wireless, turn on WPA2.

Steve: Yes.

Leo: Those are the five things, and you're secure. But you've got to - if it's on by default...

Steve: Yeah, you just cannot take it home and plug it in. And it's very distressing, Leo, if there's still routers that have WAN admin on. I mean, nobody needs it. Sure, maybe there are applications where some Soho IT admin wants to manage the routers of a few friends or something. But your typical end-user is plugging in a router to have the features of the router inside their LAN. Never do they need to get to it from the outside.

Leo: Right. Yeah, it's crazy. But, you know, this comes from the day when they would turn everything on so that they wouldn't get any support calls.

Steve: Yup.

Leo: And now fortunately I think most of the new routers are really being very smart and careful about telling people, this is what you need to do, walking them through it. They've changed their defaults. And I think that's all good. I notice the Ident port, for instance, is - and this is thanks to you and ShieldsUP! - by default is turned off.

Steve: ShieldsUP! probably put pressure on it, yes.

Leo: Chris in Iron Mountain, Michigan brings you the - wait a minute, no. I don't want to jump ahead. We've got a New Zealand question. Gary McCleery - then I'll get to Chris - in Oamaru, New Zealand, a Kiwi, wonders about proxy servers: Hi, Steve and Leo. Greetings from New Zealand. You're probably saying, "Where the heck is New Zealand?" No, in fact we all know where New Zealand is ever since "The Lord of the Rings." I run the school library as well as help look after the servers, desktops, and teacher laptops. Never a dull moment. Everyone accesses the Internet via an external proxy server. I think that's a good thing to do. Some of the students use other online proxy servers to access sites that have been blocked. Shame on you.

Steve: Uh-huh.

Leo: But of course it's a high school, and high school kids know how to do this.

Steve: They're going to find a way.

Leo: Yeah. Certain sites are blocked to provide a certain level of protection. My question is, can the use of these other proxy servers allow viruses, trojans, and other bad stuff onto our servers, or do the proxy servers simply mask the address of the sites the students are trying to access? Can our computers be compromised by the use of these proxy servers? Now, that's a great question. Love your podcast, been listening for years, always learning heaps from them. Keep up the good work. I'd like to know the answer to this.

Steve: Yeah. Okay. So a proxy server, as we know, is sort of a way station. Instead of the client going directly out to a remote web server, the clients within such networks are configured to use an intermediary server, a so-called proxy server. So that the client makes its connection, its TCP connection, to that proxy server, and then submits its request to the proxy server, which then turns around, and it generates the request outside to the Internet at large. And the process is pretty quick. It's not something for which there's lots of overhead. It's a little more overhead than not using one.

The one tip, I've never mentioned this before, but Internet Explorer defaults to looking for a proxy server whenever you start it up. And whenever I go to someone's machine who's using IE, and I launch it, and it, like, sits there for a while, I think, oh, they haven't turned that off. And it's easy to change the configuration if you - most typical end users in homes and small offices are not using proxy servers. Yet IE has it turned on by default. It has proxy server, automatic proxy server discovery, which stalls...

Leo: It's slow, yeah.

Steve: Every time you start IE...

Leo: Hate that.

Steve: ...if that's turned on. So that's just a tip to our listeners, anyone still using IE, and I hope there's only maybe one or two of you that haven't been listening to the podcast that long, because even I have finally switched over to Firefox. IE can be made much faster to start up if you just go down in the Internet connections dialogue and turn off "Automatically look for proxy servers." And from then on it's just way faster to get yourself going with IE.

So anyway, this intermediary is the one that then performs the connection. Well, the proxy server can have a number of functions. And in Gary's school in New Zealand, I guess in the library - or he runs the school library - but in the school's network they're doing other things than just proxying. They've got content filtering software which has a whole list of domains which are blacklisted that the kids that are using this proxy server for their connections are unable to access. And corporations certainly do this also. They may just block off, for example, all the social networking sites because their employees are spending too much time during the day poking around in Facebook and MySpace and Tweet and Twitter and who knows what.

Leo: No, that's exactly what they block at my kids' school - Facebook, Twitter, all that stuff, yeah. They don't want them messing around.

Steve: And so another thing that such servers could do is, and may do, is malware protection. They may perform some level of malware filtering. We talked about the Astaro Security Gateway for years here. And one of the features it offered was that it would be automatically synchronized by Astaro, and it would filter things coming and going so that your whole network behind there was protected. So similarly, it might well be that the school's proxy server is offering protection from malware and trojans, as Gary asks. However, if the students are being clever and not using the school's proxy server, but reconfiguring their clients to use some different proxy server, as I understand it he's saying can the use of these other proxy servers allow viruses, trojans, and other bad stuff to get into our network.

And the answer is yes because there isn't anything about the proxy server that performs any kind, by definition, any kind of filtering. It's additional features of a proxy server that may be doing that. The school's main proxy server, the authorized one, may be doing that. It may be, though, that other proxy servers would not be. So it's definitely something to keep in mind. If the kids are getting around the security and sort of the formal channel for accessing the 'Net, it may be that they're getting around more than just the school's filtering. They may be getting around the school's AV technology, which could represent a problem for the school.

Leo: Interesting. The software that they use at my kid's school, I know because I've tried to get around it, because I've been on the campus, and I want to tweet, also knows about proxy servers and seems to catch a lot of the most common proxies. But I'm sure, you know how teenagers are, they probably have figured out a way around that, too.

Oops, I jumped ahead a little bit. We've got another one. Poojan Wagh in Chicago, Illinois wonders whether security in hardware is a bad idea. He says: I was thinking about the security in GSM, and it seems to me the problem is that such hardware-reliant security implementations have the difficulty that they can't be modified in the future. They're stuck in the hardware. That's what happened with the DVD. It was part of the spec, and you couldn't change it once the CSS was broken. It was, that's it. Anyway, perhaps it's generally a bad idea to put security in hardware because of this limitation. Once it's in, you can't change it in the future. That also seems to be the case with 802.11's WEP, the shortcomings of which was inherited apparently by WPA-TKIP.

Now, one could make the case that back when GSM was being invented, they didn't have the capability of putting security in software. Couldn't one make the counter-case that maybe it would be better to live with no security rather than the false illusion of security and leave the true security as a software option later? After all, when the hardware security gets broken, software has to come along and fix it anyway. Your thoughts, Steve? Is hardware-based security for long-term standard products generally too risky?

Steve: It's really a good question. And reading it, I had to think about that for a while. It's like, well, is there anything fundamentally bad about security implementation in hardware? And I don't think so. It's probably the case that hardware - at any given point in time, hardware offers greater security than software, in two ways. It is generally much faster to implement security algorithms in hardware. That is, you can take an algorithm that's in software, and you can cast it into gates which, running at the speed of light, at electricity, can much more quickly process the fixed algorithm than software, which is inherently flexible. The power of software is it's soft. The power of hardware is that it can be very much faster. So for a given algorithm, putting that into hardware allows it to be far faster, or it will allow you to have a much more powerful algorithm at the same speed, thus potentially more security. So I really think that at any given point in time, hardware gives you more security.

Now, as he mentions, though, hardware is also fixed. That is, the gain that you got by locking it into hardware is its speed. Also note that it also makes it unchangeable, which is a good thing for security. Many of the problems we have with our software today is the softness of it. If we had technology that was locked in place, for example, email clients and web browsers that were in ROM and could not be modified, then they would arguably be much more secure than what we have now, where they're inherently infectable. So thinking about this, I don't really - I don't see a tremendous benefit for keeping things in software because, again, exactly as he suggests, if we had that as a requirement, because we would lose the performance edge of hardware, then we would have no security at all during that intervening time; whereas the extra leverage, the extra speed and power that hardware gives us allows us to have security where we otherwise would not.

Certainly we see example after example with, for example, as he states, WEP and also GSM, where this legacy hardware is now holding us back. We've got the power to do better security, and it's difficult to move forward. But I guess the flipside would be not having any security at all during that time, and I don't think that's - I think that's worse, as opposed to being better.

Leo: Yeah. And often you have a combination of hardware and software; or, I mean, there's ways to patch it. The original Xbox had hardware security built into it and was cracked almost immediately by Bunnie Huang at MIT. And then I think Microsoft used it as a learning tool, and Xbox 360 I don't think has been cracked. I don't think their mod chips were. So they learned what not to do. And it's hardware secure. Most hardware devices have hardware security; right?

Steve: Yes. And I think that at this point, I mean, sure, once upon a time, 20 years ago, we didn't really have a mature understanding of crypto. We didn't have algorithms. We were using pseudorandom bitstreams and XORing them. And now we have this rich understanding of how that's bad and why that's bad. So I think those are legacy problems that have been solved, and we're now at the point where we've got sufficient technology and speed that it's really no longer an issue. We don't have to choose. We've got algorithms that are strong enough, whether we implement them in hardware or software, to carry us hundreds of years into the future.

Leo: Good. Yay. I guess. Our last question is actually our Tip of the Week. Our Way Cool Tip of the Week.

Steve: Oh, yeah.

Leo: Way cool. This is Chris in Iron Mountain, Michigan with the Way Cool Tip of the Week. He calls it a Cool Google Feature: Safe Browsing Diagnostic Report. Steve, I was doing some unusual Google searching when I stumbled across a really cool page, maybe feature on Google. It's called "Google Safe Browsing" diagnostic page. I have no idea how to access it through Google's own site, but I've been actively using the site through a direct link which is a little long, but I'm going to give it to you anyway, and we'll put it in show notes.

Steve: Yes, you have to, Leo. This is so cool. Our listeners are going to go nuts over this.

Leo: It's google.com/safebrowsing, that's all one word, lowercase, /diagnostic?site=, and then you put the URL. Don't put the HTTP, just the URL. So he gives as an example google.com/safebrowsing/diagnostic?site=grc.com. And it brings up a malware diagnostic report of the site referencing the last 90 days. So when he did your site, he said it's currently not listed as being suspicious, and of the 10 pages it tested on your site, "0 page(s) resulted in malicious software being downloaded and installed without user consent." On sites that do have malicious content - I wonder what The New York Times site says right now.

Steve: I put it in, and it found it.

Leo: Oh, interesting.

Steve: Yes. And you're clean, and BitGravity is not.

Leo: Really.

Steve: Yeah, this is just too cool, Leo.

Leo: On sites that do have malicious content, it'll give statistics as to what type of malicious software it found, and how many there were of each type. Because Google has to index these sites all the time.

Steve: Yeah, so it's looking.

Leo: So if there's malware, it'll see it. For example, if I were to check out Tripod.com, I get the results, "Of the 6,400 pages we tested on the site over the past 90 days, 224 page(s) resulted in malicious software being downloaded and installed without user consent.... Malicious software includes 227 trojan(s), 187 scripting exploit(s), 61 exploit(s). Successful infection resulted in an average of 1 new process on the target machine." Wow. He says: Hope this is something new and unknown to you. I've not heard of this particular page before, but have seen the warning screens Google pops up every now and then if you are trying to get to a potentially dangerous site that pops up on a Google search. Both Microsoft and Google maintain databases of malware sites. I guess this is coming from that.

Steve: Yes. And again, our listeners are going to go nuts. You know, it's just google.com/safebrowsing/diagnostic?site= and then the domain name.

Leo: This is so cool.

Steve: And I've put in The New York Times, and it found malware on The New York Times. I put in BitGravity, just kind of, you know, out of nowhere. And it's like, whoops, there have been some problems there. Google has found malicious content there.

Leo: And this comes from sites being compromised, whether there's a server error or a...

Steve: Really anything. It's Google's search engines pulled pages and looked for malicious content for whatever reason, in an ad, in a blog that had been inserted through cross-site scripting, I mean, whatever the source, the idea being that if a user's browser went there, the user would be in trouble.

Leo: Wow.

Steve: And this is just - it's so neat to just try out different...

Leo: I'm putting all my sites in there. I want to make sure I'm not - wow.

Steve: Yeah. Yeah, and again, New York Times came up. I think it found one problem there, and it found a few on BitGravity. And so it maintains a 90-day history and shows you the statistics for the last 90 days. And this qualifies as a cool tip, a way cool tip of the week.

Leo: So cool. I'm going through all my sites right now. Let's see what Twitter, just out of curiosity...

Steve: Ooh, good.

Leo: Twitter.com. Of the 3,885 pages we tested, 0 pages resulted in malicious software.

Steve: Wow.

Leo: Malicious software has never been found. That's pretty good.

Steve: Try Facebook.

Leo: Ooh, yeah. Facebook. I know, well, I don't know. One time, Facebook. One page resulted in malicious software. Intermediaries distributing malware to visitors at site included [mancrushonmcflea.com ph]. Wow. This is really interesting. Let's try MySpace. That's got to be a...

Steve: A nest.

Leo: 68 pages in malicious software. 76 scripting exploits, 4 trojans, 1 worm.

STEVE & LEO: And a partridge in a pear tree.

Leo: Wow, that is great. I think people are going to spend the rest of the day entering sites into this.

Steve: I think so. It's just too fun to have an instant report on what has been found on popular and unpopular sites.

Leo: Thank you, Google. I'm putting this in my Google show.

Steve: This is a keeper.

Leo: This'll be my Google Tip of the Week because that's a great one. Wow. And thank you to Chris in Iron Mountain, Michigan for that way cool tip of the week.

Steve: Yup.

Leo: And thank you for sharing it with us.

Steve: Absolutely. We've got great listeners. They go to GRC.com/feedback and send me the things they're thinking about, questions they've got, ideas they have for shows and whatever's on their mind. And I really appreciate it.

Leo: That's it. We'll be back next week. Do you know what you're going to talk about next week?

Steve: Don't know yet.

Leo: It's a surprise.

Steve: We'll let the world determine it for us.

Leo: You know, there's never really a dearth of subject matter.

Steve: There's never a dull moment, Leo.

Leo: Steve Gibson is the man in charge at GRC.com, the Gibson Research Corporation, the company that gives you the great SpinRite, the world's best hard drive recovery and maintenance utility. It's a must-have. If you've got a hard drive, you need SpinRite. You'll also find at GRC.com a lot of great free stuff. Of course Shoot The Messenger, DCOMbobulator, Wizmo, the very famous ShieldsUP!. And you'll find this show, GRC.com/securitynow. You'll find 16KB versions, the little tiny ones for quick download. You'll find transcripts, which makes it very easy to search the entire show. We've got to do that on all the shows. It's such a good idea. Steve's show notes and more. And of course he's got great security forums there, as well. Highly recommended. GRC.com. Steve, we'll see you again next week.

Steve: Okay, Leo.

Leo: Safe surfing to you.

Steve: Thanks very much.