SECURITY NOW!

Transcript of Episode #234

Listener Feedback #85

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-234.mp3
Quarter size (16 kbps) mp3 audio file URL: http://media.GRC.com/sn/sn-234-lq.mp3

Leo Laporte: This is Security Now! with Steve Gibson, Episode 234 for February 4, 2010: Your questions, Steve's answers #85.

Time for Security Now!, the show that covers all the issues in security. And with us right now, Mr. Steve Gibson. He is our guru of security, and he has come down from the mountaintop to share the latest findings with us.

Steve Gibson: Or out of my cave, as the case may be.

Leo: Yeah. And those of you who watch - we do this show live Wednesdays at 11:00 a.m. Pacific, 2:00 p.m. Eastern time at live.twit.tv. And it's fun to watch because Steve has now three PDP-8s behind him, blinking, but not really doing anything, are they. I mean, they're not - are they, I don't know, are they simulating global thermonuclear war? I mean, what are they up to?

Steve: Oh, they're just there entertaining me.

Leo: It's great. It looks so good. And what I enjoy is the speculation in the chatroom about what the heck that is. Some people say binary clock. Some people say, oh, no, that's Steve's server running in the background there.

Steve: What's hysterical is the amount of time and ingenuity and trouble I went to, to get just exactly the right feel to the do-nothing blinking light display.

Leo: So they're running a program.

Steve: It's called Deep Thought. And, you know, I wrote it in the last few months. It's brand new fresh PDP-8 code running on those little PDP-8 kits. In fact, I have a note here in our errata. When I put the pages up, the guy who designed the kit said, well, you know, Steve, if you collect at least 50 people, I'd be willing to, you know, I mean, that's enough to justify another complete kit run. And we have collected something like 57 people. They're now letting him know. And the window closes on January 8th. Monday morning, January 8th, at 8:00 a.m. Pacific.

Leo: I hope it's February 8 because...

Steve: Oh, yeah. February 8, sorry.

Leo: Whew.

Steve: He said, I have to set some sort of specific time or we'll, you know, we just need to be able to say everybody who wants to get one, get your order in by then. So that is happening. And I'm excited because basically there are a lot of people who have run across, I mean, I just wish it were possible for people to get them whenever they wanted to. But the problem is we just need a critical mass. And then it doesn't make sense to make a lot more because the kids are really high quality, but in such a low quantity that they're still expensive. And so neither he nor I want to just, like, sit on inventory of unrequested kits. But it is, it's really cool to see them going.

And in fact some people wrote and said, well, what could I do with one? So I put up a new page in my PDP-8 area that answers that question: What can you do with a PDP-8 today? And what's neat is that it achieved such a critical mass back in its day that, when the Internet happened, people who were collectors and archivists put all of their material on the 'Net. So the OS is there and running, and Pascal and Focal and Fortran II and IV and Basic and all the editors and utilities, all of that stuff is still available. So, I mean, you can actually play with this just like we did back in the late '60s and early '70s.

Leo: And Steve has published the source code for Deep Thought.

Steve: Yep.

Leo: He's actually done an open source program, ladies and gentlemen...

Steve: For a PDP-8.

Leo: For the PDP-8 assembler. But what the hell; you know? If you want to learn how to write assembly, look at every line has a comment at least one line, many lines for most of them. I mean, talk about beautiful code. Just looking at this would be an education.

Steve: Well, I created it for that reason. So, you know, to help people who wanted to mess around with it.

Leo: So let's get - we're going to do questions and answers today. We've got 10 great questions and answers from Steve, questions from you. And we're going to do those in just a bit. But as always, before we get too deep into the weeds there, I thought it would be good to check in and see what security news you have.

Steve: Have some. Not too much this week. I did want to touch base with - we've talked a couple times about some of the ridiculous RIAA lawsuits against individuals?

Leo: Yeah.

Steve: You know there was a college student who got fined $850,000 or something, which was really ridiculous. But even more was there was like a mom in Minnesota, who's a mother of four, who was using Kazaa. And without her knowledge there were 24 song tracks that her system had on it somehow. I mean, it really looks from all evidence like it was completely inadvertent. And of course the RIAA just dropped on her with both feet, used the letter of the copyright law. And a jury last week awarded damages to the RIAA, that is, that this mom had to pay, of $1.92 million. Now, there's a provision that a judge has when a jury completely goes off the rails like this, where the judge can just say, okay, well, thank you for your opinion, you strange set of 12 people, but that's just ridiculous. So he reduced it by his own declaration to $54,000.

Leo: Which is still a lot of money.

Steve: I mean, for, arguably, I mean, maybe this wasn't even a crime. I mean, she had them on her computer. She was using filesharing software. So there's some responsibility. But also there's no evidence even of any damages, no proof that anyone actually got these from her machine, as I understand it. So...

Leo: They can't prove that. That's the irony of it.

Steve: Right. And so then the RIAA offered - they came back, after the judge reduced this judgment from $1.92 million down to 54K, the RIAA came back and said, well, we'll settle for $25,000 if you will ask the judge, Judge Davis, to vacate his decision to reduce the penalty. So, I mean, here's the RIAA playing games, saying we'll settle for $25,000; but we want to be able to say that this case was decided and the judgment was against you in the amount of $1.92 million so we can threaten the rest of the world, the rest of the country certainly, with this onus. And happily...

Leo: These guys are so blatant. They don't even make an attempt to hide their greed and their...

Steve: It's really just awful. And so Jammie is the mom, Jammie Thomas-Rasset, said no. I'm not going to accept your settlement. And so she's - and she's still saying, I didn't do anything wrong. I don't know how those songs got on my machine. I'm not a music pirate. And the thing that really does annoy me, Leo, is there are certainly really egregious music thieves operating on the 'Net. Go get one of them, not some mother of four in Minnesota who is probably innocent of this.

Leo: Steve, Steve, I know you're upset about this. I don't know if it's caused your nose to bleed, but I think you - is your left nostril bleeding a little bit there?

Steve: What?

Leo: I mean your right nostril?

Steve: Are you kidding?

Leo: Your left nostril, yeah. It seems like it might be. No?

Steve: No.

Leo: They're telling me on the chat, maybe there is a - there's a rust-colored streak on your mustache, then. I don't know what it is. It could be an artifact of the video. I'm going to edit that out, obviously.

Steve: Nothing dripping off the ceiling on me, so.

Leo: Okay. We have a doctor, we have a doctor in the audience. And so I just thought, you know, she's concerned about your health.

Steve: Maybe I'm getting so upset that I'm blowing a blood vessel.

Leo: Those RIAA losers. Anyway, you know, I agree with you. I think it's just unconscionable. Now, I'm not sure how innocent Jammie is, Jammie Thomas. But as you point out, you can't prove intent here. And I know that that's a key part of law in general is intent; right?

Steve: Yeah.

Leo: And you just can't prove it.

Steve: Yeah. It just, it really seems like an abuse of power. I'm pleased that the judge just was reeled by this $1.92 million judgment. And that was based on a fixed dollar amount per song is the way the RIAA was going after this.

Leo: Well, let's be honest, or fair. It's the copyright law, as well.

Steve: Yeah.

Leo: So the law gives them the right to ask for fixed damages. Of course this law was written in effect by the recording industry, the DMCA and so forth. But that's what they're going for. They have a set amount that they can ask for, and they did.

Steve: Yes, they did. We have a mysterious new and troubling problem for IE. I mean, when don't we?

Leo: Uh-oh. Again? Jeez.

Steve: This one a Jorge Luis Alvarez Medina just demonstrated this at the Black Hat Conference in D.C. The title of his paper which he presented, which is available online if anyone is curious, is "Internet Explorer Turns Your Personal Computer Into a Public File Server."

Leo: [Laughing] Okay. Yeah, maybe you want that, I don't know.

Steve: It's not good.

Leo: No.

Steve: It turns out that this is actually not the result of a vulnerability, but a clever interaction of properly functioning features, which Microsoft certainly did not intend. Microsoft has acknowledged this report. Jorge has not released all the details because he's waiting for Microsoft to fix this somehow. But it's sort of a blended collection of behaviors, which he just sort of talks about without getting into any specifics at all, involving the way zones interact with some protocols that IE makes available.

The bottom line is that there is a way which he has shown for any website to present, or web page, meaning email also, if you're using IE as the viewer for your email. We've talked about how that happens, where the user clicks on a link. The result of clicking on the link is that a filesharing connection, a so-called SMB, server message blocks, an SMB connection is established from your computer to a bad remote site. And over that connection the remote site has then full access to your file system.

It's literally, it's just like what originally caused me to create ShieldsUP! and tell people that their server ports were open and that they were sharing their C drive with the entire world. This, here we are in 2010, and this is a vulnerability in IE. Certainly Microsoft will look into it. They'll come up with a workaround of some sort. He's not going to disclose details until they've done so, after which he'll say, okay, now that it's been fixed, here's how you do it. So anyway, just another little blip on our IE radar.

Leo: Just when you thought it couldn't get any worse.

Steve: And we do have a question in our Q&A about someone who had trouble completely getting away from IE that I want to talk about a little bit.

Leo: Oh, good.

Steve: And the only other real bit of security news is that for anyone still using Real player, and I don't know anyone who still is, I mean, I stopped using it because it was a catastrophe back when you were still seeing lots of .RM, Real Media files, on the 'Net. The good news is it's pretty much faded out. But you probably know, if you are a user of Real Player, maybe your corporation only publishes things for internal consumption, who knows what. But there is a security update. And anyone who knows they're using Real Player and is still using Real Player ought to go to Real and update themselves. The rest of us, anyone who has it installed and thinks, hey, I forgot I installed that a couple years ago, and I haven't used it, just get rid of it.

Leo: Piece of junk, yeah.

Steve: You probably don't need it, yeah. Just, I mean, it was a real exploitive...

Leo: No, what's sad is that there's still sites, I go to them all the time, that require Real Player to playback audio. I guess they just never updated it or whatever.

Steve: Wow.

Leo: And it's just, you know, I went to a site the other day that gave you two ways you could listen to audio. One was Real, and the other was Windows Media. And I thought, you guys never heard of MP3, huh?

Steve: Oh, yeah, wake up. So in errata, I wanted to just acknowledge to our listeners that I have heard everyone's request for the LockNote security analysis. LockNote is that cool little notepad that I talked about some many weeks ago. And I said, because the source had been posted over on SourceForge, that I wanted to check it out to make sure that they were doing things the right way. I have no doubt that they are. I believe they are.

But since the source is there, and since there's no statement from them about how they are processing the password - remember that we just talked recently about a really bad use of a password where you gave the password, remember, it was a family of password-based AES-256 encrypted thumb drives. But the way they implemented it, once you gave the password, then the same key was always being used to decrypt the data. So it was completely bogus. I mean, it meant that it was trivial to bypass this. That's the kind of stupid mistake you just want to make sure no one's making.

So I have seen a bunch of reminders from our listeners - hey, Steve, whatever happened about telling us about LockNote? I've got the source. I tried to get into it for this week, but I just didn't have time. So next week for sure I will be able to say yes, definitively, I've looked at the way LockNote is handling the password. I'll be able to explain exactly what it does and give people the warm and fuzzies that they're looking for. Because it's a cool little app.

Leo: Yeah. I have your notes in front of me.

Steve: Well, we have a little iPad discussion.

Leo: So let's talk about iPad. Now, you and I both use the Kindle.

Steve: Yes. In fact, I have to say, Leo, I was thinking about this because I knew we were going to talk about this, there is no other single piece of technology that I use more, or even as much as, the Kindle.

Leo: Wow.

Steve: That's really true. I mean, it is, when I leave the house, I have it with me. And I spend hours a day reading stuff, mostly periodicals, with the Kindle. David Walker, who is the ex-head comptroller, and he was running the general accountability office recently, he's got a new book out called "Comeback America," talking about getting our fiscal house in shape. And I ran across a mention of him relative to the U.S. budget that just came out recently. And so I grabbed that book, and so I'm going to read that. But it'll be the first book I've read in a long time. Mostly I use it for periodicals, for which I think it is absolutely perfect.

Leo: See, I would disagree. I think that, while I agree with you, I love the Kindle, and I do the same as you, I carry it everywhere, I don't find it perfect. It's just the best that I could find so far. And where it really lacks to me, it's great for reading linear things, but I like to jump around. Like when I'm reading a newspaper, I like to scan the front page and then go to something, and go to something. And the Kindle does not make that easy.

Steve: Well, I agree, it does require a different approach. Now, Paul and I are apparently - Paul Thurrott and I are in sync with this. He said he's been reading The New York Times on his Kindle, and he loves the experience. I'm the same way. I subscribe to The New York Times, the Financial Times, and The Wall Street Journal.

Leo: Wow.

Steve: And you do have to get used to sort of this tree-structured approach. And I know what you mean, the idea of, like, glancing at an actual front page where you can see it all laid out and very quickly, you know, scan visually, looking for topics that seem interesting. And then presumably tap on them, and that'll zoom you into that and so forth. So it's certainly the case that the limitations of the Kindle's eInk screen have forced a design compromise on the reading experience that takes you from the way you used to do it to the Kindle approach. For me the tradeoff is that it is very lightweight and easy to hold. The battery life is so long that you really - it's not even a factor. It's long enough that that sort of leaves the equation completely.

So, I mean, I will - I've notified Apple I want to be notified as soon as the iPad is available. One of the things that I wanted to say was that for $499 it's an unbelievable value. And I don't think that Jobs and Apple want anyone to buy that one because they're making much more money on the 32GB and the 64GB at $599 and $699 than they are at $499. I mean, a Palm Pilot not that long ago was more than $499. And it doesn't hold a candle to this thing.

Leo: Right. No, I was impressed by that price point, too. Although I am going to buy the $499. I think many people will.

Steve: Oh, I am, too. See, I'm going to buy it because I don't think I'm going to use it.

Leo: You're holding off for the one-month-later 3G version.

Steve: No, no, no. I don't think this answers a problem that I have.

Leo: I see. So you think you're going back to the Kindle after you try it.

Steve: Yeah. I mean, I have to have it because from everything I've heard - and I haven't held it, but I've listened to Andy, I've listened to you - that it is a transformational thing. And the thing that I think it may have for me is being an instant-on web browser when I'm in a WiFi area. When I'm at Starbucks or in various other places I hang out that have WiFi, even in my home, like watching television I'll see - something will pop into my mind, and it would be really nice to be able to really quickly jump on the web and type something in and find something. And I think it's a feasible, I mean, I'm sure it's going to be a very practicable web browser.

And then there are other things. Now, I do - this whole issue of Flash is an interesting controversy. What most people think of when they talk about Flash seems to be video, which is not what my focus has been on at all. What's interesting about Flash is that it is a powerful interpreter. And it bypasses the iTunes store. The reason Apple doesn't want Flash on their devices is that you can write applications in Flash which are fully functional, useful applications.

Leo: And just serve them up as a web page.

Steve: Exactly. Serve them up as a Flash application. And Jobs and company doesn't get money for your purchasing the little button.

Leo: To be fair, they've encouraged people, in fact that was the original plan for the iPhone, to write web applications. And that's how Google's gotten around the Google voice blocking. You don't need Flash to do that. You need JavaScript, you need CSS, you need all of the tools that Apple has allowed. I think that you hit the nail on the head - this is a security show - that it is a powerful interpreter. That's why Flash is a security problem. That could be one reason. And of course it's one company that owns it, and that could be another reason. I think you could do everything that you're talking about without Flash.

Steve: Well, except there are websites that are written in Flash.

Leo: Well, of course there are. But I think what Jobs is saying is let's get rid of those. Let's move on.

Steve: Well, so I do take issue with him saying you are holding the Internet in your hand, and you can surf anywhere you want to.

Leo: And you don't have Flash, right. No, you're right.

Steve: If you don't have Flash, then there are sites...

Leo: He should have addressed this. He should have just said, "We have decided not to put Flash on our phones and now on the iPad because...." And he has a credible story to tell. He shouldn't lie and say the web in your finger, in your hands, because it isn't.

Steve: Right. And certainly it'll put some pressure on sites that were written in Flash, somehow not to be written in Flash.

Leo: That's already done because so many smart phones don't support Flash. Android doesn't really. So in effect, if you want to support mobile computing, if you're Hulu, for instance, you're going to move on. And the good news is there's an alternative. In fact, YouTube now has a beta program where everything is done in H.264 and HTML5, and it works great. And it works fine on the iPhone and the Android phones, and it's all you need. In fact, we have a streaming, an HTML version of our video in prototype that streams using H.264 and has no Flash. Because in the long run we're dependent, heavily dependent on Flash, if you want to watch it live.

Steve: Well, and you're probably aware, I'm sure you're aware of the as-yet-unsettled question of HTML5 and support for H.264 versus the Ogg format.

Leo: The Aura, yeah.

Steve: Yeah, because of patent issues. I mean, the MPLA does, I mean, H.264 is patent encumbered.

Leo: Patent encumbered, yeah, yeah.

Steve: There's no doubt about it. And so organizations like Mozilla are saying, you know, we don't - how do we support that?

Leo: Right. No, and I'm with that, and I'm sure that'll get resolved. Apple does so many closed things, I mean, so much of what Apple does is highly proprietary and closed. This is one case where I think maybe for proprietary and business reasons. But for whatever reason they're supporting open technologies over a proprietary technology. And I just think that they ought to be applauded for that. It's certainly a bad business decision in some ways. Maybe they're doing it for other business reasons.

Steve: Well, and I really - I love the idea of them putting pressure on the serving side of the industry to move video to HTML5.

Leo: That's all that has to - if someone, one big player - Microsoft could have done it, but did not. Mozilla could have done it, but did not. All it takes is one player, one of the big three, to do it.

Steve: And, fortunately, Google has decided that they're going to do H.264, which is the far better codec to use over Ogg.

Leo: That's what YouTube is using, yeah.

Steve: Yeah, exactly. The other thing is there was a lot of issue about camera, the lack of a camera on the iPad. And my feeling is that it's an example of a decision they had to make just for cost reasons. While, if you think about it, the things that are on the iPad, which is to say not much, get 100 percent utilization - that is, the touchscreen, the screen, the processor, the battery - what is there is absolutely utilized fully. Whereas anything else they were to add would always represent a fixed cost for every single device that they've produced, but lots of people wouldn't be using it.

So I also think of that as the way they sort of got this add-on, accessorized deal, where if you want the USB connectors, you've got to get an extra gizmo to plug that in. If you want to import video, then you've got another little add-on. What they did was, again, they minimized what the base product does to completely minimize their fixed cost. And then they've accessorized this so that those things that people do want to add, they can plug into the docking connector, where they're really going to make - where they're going to need those functions. And those are also other little profit centers for Apple.

So anyway, I'm stoked about it. I'm glad it exists. I can't wait to get my hands on it. Maybe I'll use it, I think I probably will use it as my web browser. And I'll have to wait and see how I feel about it as an eBook reader. I mean, I love the Kindle. The idea that the battery life is as long as it is does make a difference to me. But maybe, if the experience with the iPad is good enough, enough better, then I'll switch to it. But I just don't know.

Leo: Yeah, we won't know until we get it and spend hours trying to read on it.

Steve: And I have to say the idea that Kindle has opened their device up to other apps? It's like, what?

Leo: Amazon, it's really interesting because Amazon clearly sees the iPad as a shot across their bow. And now there's this big war going on. You heard the McMillan thing, they pulled all the McMillan books off because McMillan wanted iPad-style pricing on Amazon, and Amazon said no. And then they flip-flopped the royalties. It was 70-30, now it's 30-70 in favor of the publisher. And now this, they've offered an SDK so you can write - but you're right. You can write apps on the Kindle?

Steve: What are you going to do? I mean, the problem is, again, I think this is nutso. Unless - I just don't know what you can do. I mean, that screen is so limited that it's not like an iPad at all. You can't animate things. You can't have little things walk around. I mean, and the moment you do anything, your battery life just goes to hell. I mean, we know that listening to music on the Kindle drains it in a matter of an hour.

Leo: Well, and the irony is, of course, all the people wanting multitasking, there's no multitasking on the Kindle, either.

Steve: It barely does single-tasking.

Leo: Talk about a limited platform. But maybe somebody will do something interesting.

Steve: Maybe, like, crossword puzzles or something not very animated. I mean, it can't be with that kind of screen technology. We'll see. I just - to me it seems really strange. It's like, okay, well, good luck with that.

Leo: Good luck with that. So we'll talk in two months.

Steve: Oh, can't wait.

Leo: We'll talk in two months.

Steve: I mean, it looks just like a spectacular little toy.

Leo: Well, it's a good toy. Toy is a good word. In fact, maybe that's the word we should start using because people keep comparing it to computers. It's not a Netbook. It's a toy. But it's...

Steve: And the other thing people talk about is how many different devices they're having to carry. And they look at it in terms of a consolidation. The problem is, my Blackberry, for what the Blackberry does, it does it perfectly. The fact that I can get a little trickle, I get email trickling in, and instant messaging from friends trickling in, I mean, what it is, it is perfect for that. I believe the Kindle, for what it is, is perfect for that. I've got my ultimate laptop is this little Lenovo X200s. What it is, it is perfect for that.

And so I do have a collection of individual devices, each perfectly optimized for the way I use them. And I've given up on the idea that one single thing is going to replace them all. It's just not going to happen. So I may add another. I'll add an iPad, and so I'll have a quick web browsing gizmo. And maybe something to play with puzzles and things. I mean, it just sounds like it's a beautiful piece of equipment.

Leo: Well, exactly. I mean, think about crossword puzzles on it. I can see you bringing it - you're bringing the Kindle to the Starbucks. This is the perfect Starbucks computer, don't you think?

Steve: Yeah.

Leo: I mean, that's - and I don't care about 3G or a lot of extra memory because how much memory is on our Kindles?

Steve: Exactly.

Leo: Hardly any.

Steve: 16GB is something I'm not...

Leo: It's not the memory.

Steve: And I did want to mention also, you were a little annoyed by the non-HD form factor. And I think that my take on that is, having messed around with video a lot, 1024 pixels of horizontal resolution is really a lot for video.

Leo: Okay. I've been kind of more convinced. Andy kind of convinced me on MacBreak Weekly. It can't do - it can't be ideal for both.

Steve: Exactly, and that's just it, is you really do need a more square aspect ratio for many other things. And it's only HD where the - and remember that it was only recently that cinematographers figured out that wide was better because it was more like the way we see, rather than square, which is what televisions have always been historically.

Leo: It'll be fine for a lot of things. And we're going to do a TWiT application. We're actually - Houdini7, who writes our TWiT application for the iPhone, is already working on one for the iPad that will in effect have multitasking. You'll have video and chat. It's going to be a great way to watch TWiT Live. I can just think of a lot of ways that you can use it.

Steve: I noticed one of the strangest things, too. On the screen during the Apple presentation was - the pixels were not square on the screen. Whenever they showed the iPad oriented in landscape mode, it looked really wide. But when they rotated it and showed it in portrait mode, it looked really squatty and almost square. And it was like, it just struck me as really strange that no one would, like, fix that.

Leo: I didn't notice that when I held it. I think one thing about this, you know, the Fat Nano was like this, too. When you saw pictures of it, it was like, this is ugly. What is this? If I look at pictures of the iPad, I go, I can see why people would go, gee, that's not very appealing. When you hold it, there is a different experience. I don't know if it has square pixels or not, but I would expect it would.

Steve: Oh, no, no. I mean, no, it was the projection screen.

Leo: It was the projection, okay.

Steve: That's what I meant. When it was big up onscreen, whenever it was laying on its side, it seemed really long and stretched out. And it's like it sort of - and it accentuated that aspect of it sort of unfairly because then when you turned it upwards, it was sort of squatty and more square. And so it's like, okay, that's a little interesting bit of showmanship there on Apple's part.

Leo: Well, they've never released that video tap for the iPhone, and I presume they use the same one for the iPad, that lets them put it on the screen. No one knows how they do it. It's magic. And apparently it's not perfect.

Steve: Or it's probably not in the production models. It's probably just in...

Leo: Maybe that's it. Maybe they build a video-out into one Steve edition, the Steve edition with video-out.

Steve: I did have a fun SpinRite story that I wanted to share. Mark Jones is a listener and wrote to us. And his subject was, "Should I Be Happy That I Have a SpinRite Story?" He said, "I'm a loyal listener and SpinRite owner. I finally have a SpinRite story. My neighbor was having trouble with a Vista computer and took it to have it serviced at a national chain. I'm not sure they want me to give out the name, but it rhymes with Meek Pod."

Leo: Meek Pod, Meek Pod, Meek Pod. Oh, yeah, yeah, them, yeah.

Steve: Of course we know that that's the Geek Squad. Oh, Mark didn't say that, I did. "They replaced her hard drive, saying the old one was defective, and offering as proof of its failure Western Digital's diagnostics failure during a complete scan. They installed a new disk and convinced her to upgrade to Windows 7 as a clean install. She commented to me, her neighbor, that she was happy I had told her about using an external disk for backup. She was only going to lose about a month of pictures and other files, even though even that was a problem for her. The broken drive is a SATA drive. My desktop has external SATA connections. So I put the drive on and ran SpinRite. By morning it announced that errors had been repaired. You can guess the rest. The drive is perfectly fixed, and I moved all of her files off of it. She lost nothing. Way to go, SpinRite."

Leo: That's great.

Steve: "In addition to her Windows 7 install, she was sold antivirus and antispyware software. I suggested that before purchasing such items in the future she should talk to me." Mark says, "Keep up the great work. Thanks to you and Leo."

Leo: Well, I hope he recommends some antivirus, antispyware software.

Steve: Well, I'm sure he's probably thinking of Microsoft's Security Essentials, which is what I'm telling everyone is just use that. Problem solved.

Leo: Yeah, yeah. All right, Steve Gibson. I have questions for you, if you have answers.

Steve: Let's do it.

Leo: All right-y. Starting with Question #1. This is from Van A. Eash here in Laredo, Texas, he says. I've been a listener of Security Now! for a year or more. Love your show. I look forward to every issue. My question is regarding switching to Firefox. I have attempted to switch from IE to Firefox. However, the company I work for uses a couple of applications that are built on .NET. That's the Microsoft programming technology that does in fact use the Internet, and I guess Explorer. When I attempt to run them in Firefox, they don't work. Is there any way for me to run .NET applications in Firefox? It's the only reason I can't leave IE behind. What do you think, Steve?

Steve: Well, there was for a while a .NET framework assistant that was available for Firefox. And we talked about it some time ago, that there was a known security vulnerability in it. And Firefox, the Mozilla folks, disabled the use of the framework assistant until Microsoft got it fixed. I looked around for it and its compatible version under the current 3.5 and was unable to find it anywhere. So I thought, okay, well, maybe Microsoft didn't make it available under 3.5, or I don't know.

But this gave me essentially an opportunity to explain to Van that it's certainly the case that it's likely not possible to completely leave IE behind. I'm still using IE when I manually go to Microsoft's site to check for Windows Updates because I like to do the custom version, not the express. I want to look through them. You know, there are some things that I just do not want added to my system which Microsoft keeps trying to push on me. And so it's like, eh, no thanks. So I've got some of those declined and hidden so I'm not being bothered about them all the time. So unfortunately IE is tightly enough integrated into Windows that you can't get rid of it all the time.

So I just sort of wanted to suggest that the goal should not be to absolutely, positively never need it. And specifically, for example, if there are in-house applications running on .NET that need IE, well, they're not a problem. I mean, they're not maliciously created, and they're not exposing you to danger. So the idea would be, feel completely comfortable with using IE to visit Microsoft when you need to, or to use your corporate internal .NET applications. That doesn't represent a problem. Just stay in the habit of choosing Firefox when you're doing your normal web surfing. And so it's really - you really don't have to absolutely never use IE. You just have to only use it when you have no choice.

Leo: A number of people in the chatroom are saying, oh, yeah, there's an IE tab extension for Firefox. But that's just - that's not solving - that's saying use IE in a tab in Firefox. And it...

Steve: Yeah, exactly, yeah.

Leo: If you want to - you can just open the window, IE window, too. I mean, you don't need - I guess it means you don't have to leave Firefox, whatever that means.

Steve: Yeah. And, see, I would also decline that, just on the grounds of wanting to keep Firefox as simple as possible. And who knows what new exploit will come up for the problem of running the IE tab in Firefox. You might have it running and - you're just asking for more trouble, I think.

Leo: Let's go to Uppsala, Sweden for our next question. He commends everyone to virtualize. That's Hans in Uppsala. I've been following Security Now! as much as I've been able to for the past month or two. I think it's a great show, and I'm recommending it to as many friends as possible. I do have a comment on the subject of having a separate banking computer. We've talked about that quite a bit, actually. He says having a separate computer or, say, a dual-boot system is safe and all, but it can be a hassle to achieve - that's for sure true - and/or incur a high cost to achieve the best security level.

I would suggest that most people could make do with a dedicated virtual machine for banking. It's far cheaper than an extra computer, and much easier and less annoying than a dual boot system. Use VirtualBox - he says Microsoft's virtualization system. Actually VirtualBox is from Sun, and free - or the VMware player/server/workstation, that's free. Microsoft has a free version of Virtual PC, as well. He says: I use VMware workstation quite a lot and will never set up a dual-boot system again unless I absolutely have to. And I think I've asked you that, Steve. Is virtualization an adequate solution?

Steve: Thus this question.

Leo: Yes.

Steve: No.

Leo: No.

Steve: There was originally a big flurry about the security benefits of virtualization. And what followed not long after was a bunch of smart people punching holes in not only those - not only in those virtual machines, but in those concepts and theories. In this case it's careful to think about the attack problem. That is, running a banking instance of the operating system in a VM could potentially protect you from stuff getting into that machine, into that virtual machine. But that's really not the threat model. What you want is to prevent anything from being able to play games with your session that you've established between that virtual machine and the bank.

And the problem is the virtual machine is running inside of your real machine. And it's your real machine that could have some malware installed, for example, sitting, monitoring all the packets that go in and out of the physical adapter. So even though you've created a virtual container and built a mote around it so that nothing can get to it, well, the data still has to cross out through your real physical computer, which could be under the influence of some malware, and thus you lose that protection.

So I just - this was a great question because we touched on it before, but I just sort of wanted to accent this a little bit and make sure that people understood that the idea of creating a little protected zone where the data then crosses a non-protected zone, well, that's - we've lost the value of our protection. Not all of it, but a critical aspect of it. And so, for example, a perfect instance would be that, if there was a hack which took advantage of the SSL/TLS renegotiation failure, that's exactly where it could insert it. That's a man-in-the-middle attack that allows other things to stick themselves on the front of your connection, which is exactly what you want to avoid in a banking scenario. So that kind of, essentially, your own external machine would be the man in the middle that would be attacking the communications coming out from the virtual machine and making that really an unsafe solution.

Leo: Yeah. So there you go.

Steve: Yeah.

Leo: Francois Pominville in Montreal suggests a Linux boot CD might be dangerous. We've talked about that as an alternative. He says "Power down before booting. Something you forgot?" First, love the show. Listening to it from the beginning, keeps my paranoia alive and well. On a previous show you told people - actually I think it was me, I don't think you said it - to boot form a Live Linux CD so that you're in a fixed environment. But if you had a virus in Windows that is memory resident, meaning that it'll keep running even after a reboot, wouldn't that be a problem? I'd recommend that the computer is powered down during 30 seconds before booting from that Live Linux CD. Does that make sense? Is there something that can survive a reboot?

Steve: It's a good question. I tried to think of some way that something could stay around when you're changing - assuming you're on Windows normally, and you're changing to Linux. I mean, even a boot sector virus that installs itself on the first track...

Leo: That wouldn't work because you're not touching it.

Steve: Right. It's very OS dependent. And so I can't see a benefit to powering down.

Leo: It's conceivable you could have a BIOS virus. But that powering down wouldn't help in that case.

Steve: And neither would, I mean, nothing would help at that point.

Leo: Nothing, if you're that screwed, yeah.

Steve: You've got something really deep into your machine. So I don't see a benefit. I think that, you know, maybe pressing the red reset button - of course, computers increasingly lack actual physical reset buttons, because that sort of does a hardware reset. But that's, again, I can't see any benefit to powering off the machine. And I dislike power cycling things because that tends to break them, too. So I think that just rebooting a Linux CD is enough.

Leo: Yeah. Whew.

Steve: Yeah.

Leo: I had enough trouble with that. Moving on to Question 4, a listener needing anonymity, so we won't say her or his name - in Michigan, though, that's a pretty big state - makes a good point about broken SSL renegotiation. He says: I was listening to "Let's Build a Computer," which was last week - can't wait for the rest, by the way - on my way to work. I heard you and Leo discuss what happens when one side of the connection has SSL renegotiation disabled, as in the case of Apple's recent update to its broken SSL/TLS. In the discussion that followed, you described the unlikely instance of SSL sessions that last a month or more, which is correct.

However, where this issue also arises in more practical terms is with client certificate authentication, which is a use case which you touched on when you previously discussed session renegotiation. You might have forgotten to mention it this time.

Steve: Yup.

Leo: At least with Apache, the behavior of client certificate authentication depends on whether you apply the directive on a per-server or per-directory context. In the per-server context you have to supply a valid client certificate to establish the SSL connection to the server. In a per-directory context, you establish a non-client certificate authenticated connection first. Once you request a directory requiring certification, Apache forces a session renegotiation before giving the client the data. So you start with an insecure connection, try to go to that directory. Apache says, no, wait a minute, it's secure, let's renegotiate so that we can have a secure connection. He gives a link to the Apache docs for this.

http://httpd.apache.org/docs/2.2/mod/mod_ssl.html#sslverifyclient

In the case where Apache is compiled with OpenSSL 0.9.81, thus breaking session renegotiation, client certificate authentication in a per-directory context no longer works. Clients are unable to access the directory protected by client certificate authentication.

Thanks for the extremely informative podcast. Keep up the good work. Don't say my name on the air because my position in the industry is sensitive. Obviously this is somebody who deals with this kind of thing because that's a very deep understanding of what's going on. Is he right?

Steve: Yes. And when we originally talked about renegotiation, I remembered this as an instance where renegotiation could occur. So let me sort of explain it sort of more in common terms. The idea is that it is - first of all, we're talking about client certificate, meaning not the server certificate, where we're authenticating the certificate, but where the client has a certificate to prove its authentication. So it's a nice way of establishing very good security on a site.

So, for example, you might have a public site like Google, where any random person, anonymously, is able to connect, just like we all do when we bring up Google's home page. But Google employees might have a special certificate installed on their laptops, for example, which allows them to get access to specific directories on the Google domain that nobody else, no matter what they try, who lacks that client-side certificate, can access.

So the mechanism for this does require renegotiation because - so the sequence would be an individual connects over SSL/TLS, gets a secure connection to Google's domain. Now, what's significant about that is that remember that this connection is established before any data flows, that is, before the request for a page. So the server doesn't know what page, that is to say, for example, what directory the connecting client wants to visit. So a normal, non-authenticated SSL connection is established, that is, where the server is authenticated with a server certificate; but the client isn't because it could be just you or me, Leo, wanting to hook up to Google with HTTPS.

Leo: Right.

Steve: So that secure tunnel has to be established. Then the request for a specific page is sent through the established SSL connection. If the server sees that that's a protected directory, then it has to say, oh, only authenticated clients are able to access that directory. So it then issues a renegotiation request to the client browser, saying you've got to prove to me who you are if you want me to honor the request you just made. So that's where, on an established SSL connection, an SSL renegotiation occurs on the fly to establish a new security context, which in this case is authenticated at each end with the client has the proper certificate.

So the point of this is that, for example, Apple's recent update to SSL, which we talked about last week, which removed renegotiation completely from the protocol, would prevent this scenario. And this is an established scenario. So Apple made a decision that said, well, it's so uncommon, versus the danger of malicious use of renegotiation, that until we get the fully fixed TLS next version, whatever it's going to be, probably a 3.1 or something, we're just going to shut this down completely. Almost nobody would be inconvenienced by it. But it is possible that somebody would.

And so anyway, that's what this anonymous listener was bringing back to our attention. I did discuss this when we talked originally about SSL renegotiation. But I failed to mention it when we were just recently talking about, oh, yeah, Apple updated their security, yay, because they've removed renegotiation from the scenario. Well, that could cause a problem for some people.

Leo: How common is per-directory certificates? I mean, is that a common way to do it?

Steve: Yeah. Yeah. I mean, there's lots of man pages on the Apache site about here's how you set it up. And you can, I mean, I could easily imagine a situation where - and I know that some corporations do this, is they give employees certificates that allow the employees access to private areas because that's more secure than just a username and password.

Leo: Right.

Steve: It requires that physical machine with that certificate installed. And you say, okay, if you're going to go down this branch of our web domain, then you've got to prove who you are.

Leo: Right. Question, is it 5 or 6? It's 5. Joshua in Perth, Australia suggests Live CDs are not necessarily impervious. Also wants to talk about Live USB: I've been meaning to post this comment since hearing about the recommendations for Live CDs in banking. We were talking about that earlier. Admittedly, Linux is so much less of a target than other operating systems. But I just wanted to point out that it is possible to modify the Live CD and/or your hard drive, possibly making persistent changes. This is, after all, how CD installers work. And it's usually trivial to get root admin on a Live CD, possibly already running as root, known password and simply prompting the user, after which point you can write to anything, potentially even the CD itself, if the OS is running in RAM. We make use of this at work in updating otherwise fault-tolerant systems which appear not to be writable to avoid accidental writes but temporarily allow specific deliberate writes.

An additional point is that many people, Netbook owners perhaps, will use a Live CD on a USB key, which is obviously trivially writable. My suggestion is, either make sure you put your banking Live CD into a CD-ROM drive - I think that's what I was suggesting - or use a USB key with a physical hardware switch lock on it. If you can't find one of those, SD cards usually have the switch and should boot from USB card readers. And for the ultra-paranoid, add a physical switch to your computer which disconnects the hard drive. What do you think about that? Is that necessary?

Steve: Well, what he's talking about is technically a possible vulnerability. And I liked it just because it's something we hadn't looked at. He said CD-ROM, and he actually meant ROM, meaning not a drive which can even write. Because he's assuming that we're...

Leo: Oh, I see, CD-ROM drive. I get what he was saying, okay.

Steve: Right, right. Not a CD-RW, for example.

Leo: Right.

Steve: And so he's assuming that maybe something could exist which could alter the, literally, the contents of the CD.

Leo: Most Live Linux disks are finalized and cannot be rewritten to.

Steve: Exactly. So you do want the disk to be closed so that it will not accept any modifications. And then, you know, I suppose a CD-RW could erase the disk or make some changes. But it just - this is so far out there. I sort of liked it from a standpoint of...

Leo: He's thinking about all the possibilities.

Steve: Which is exactly - that's good security protocol is okay, wait a minute. If it's a CD-R, then it's a recordable CD. And how do we know some bad thing isn't going to come along and record something else on it. So technically you're right, you know, you want it to be absolutely a ROM. And certainly the idea of putting a Live CD image on a USB key is a little less safe because it's writable. Again, with all the hoops that somebody's going through, these seem way, way out the bell curve in terms of something one needs to worry about. But this is the way security thought goes.

Leo: And he is right about the hard drive. There are a lot of Live CDs that save data on the hard drive. And presumably that's something a Live CD can easily, if you boot from a Live CD image of, say, Ubuntu, that that wouldn't be necessarily so hard to figure out how to do.

Steve: Right.

Leo: So that is actually, seems to me, a legitimate criticism. Fortunately we don't see virus authors planning for that situation. They pick low-hanging fruit.

Steve: Exactly.

Leo: They go after people running Windows 98 with no security. Ben in Brea, California, makes a terrific and troubling observation about the Firefox master password prompt: Like many, I use a master password to protect my login passwords when using Firefox. We've talked about that before, you absolutely have to do that because Firefox otherwise stores the passwords in the clear. He says: The way it behaves for me is that I'll just be browsing around across many tabs when all of a sudden the "Password Required: Please enter the master password for the Software Security Device" window pops up. That's actually the way Firefox works. It doesn't pop that up right away, but often does later, and certainly does before you ever need a password.

I've gotten used to quickly and automatically typing in the master password so I can get on with browsing the site. What worries me is that the master password window could pop up at any time, often for a tab I'm not even looking at, and it looks like any other JavaScript text input popup. How do I know Firefox made that popup window? Seems to me any website could easily phish for my master password. I just type it right in, and boom. I would feel a lot safer if Firefox only prompted me at startup, before any websites are loaded. Am I missing some way that Firefox is protecting me from this?

I suspect a black hat with my master password would also need actual control of my computer to do any damage. But perhaps its compromise might be one step in some future blended attack. I apologize if you've covered this before. I searched the archives, but I am not fully caught up on listening. Love the show, thanks. That is a good point.

Steve: It's a fantastic point. I mean, and again, it's another example of really correct security thinking. If there are any Mozilla developers within the sound of this podcast, it would be a terrific feature to add to Firefox. And the feature would be, if you've configured a master password, give the user the option of entering it only once when the browser is launched. The user would have to understand the liability that, if only issuing the password once, that unlocks the password set for the entire browser during its session. But then they should never get another popup asking them to enter their master password. And if that ever happened, they would know that it was coming from a script running in a browser window that was trying to trick them into entering their master password.

I mean, this is a classic example of something designed to increase security which ends up happening all the time. The user gets accommodated to entering the master password, and, exactly as this listener says, it's possible to get spoofed and to enter it when it's not actually the browser asking, but rather something running in a browser window. It's a great, great thought.

Leo: Good point, yeah. Let's see, moving on. Question 7, Bill in D.C. wonders about forcing your lawyer or accountant to use security. By the way, my lawyer and accountant are very secure, and I checked that. I was wondering if there's a way to force someone to use security and protect our personal information when we give it to them. This is really geared towards professionals like lawyers and accountants, realtors, anyone else who keeps a lot of private information on lots of people.

I recently moved to the D.C. area and ran into the situation where two professionals, my realtor and my tax guy, do their official business on Hotmail. Jiminy. Both these gentlemen are very good at their jobs. But they're not up to date on computer security; nor, and this is more troubling, do they show an interest in "making things more complicated." When I think about potential hacking targets, these guys would be higher on my list since their email accounts contain all kinds of data on lots of people; whereas most individuals' accounts just contain their own private information.

Since I really doubt I'll be able to convince them to use, say, PGP, I was considering sending them information in an encrypted zip file and just telling them the password over the phone. That way at least my information is secure in transit and when it's residing on the Hotmail servers. I'm sure they're going to store it on an unencrypted personal computer, so I'll just have to hope they keep their work computers patched and don't do any dangerous surfing on them.

Any other ideas? I know in the past you've talked about some of your interactions with your lawyer and not even discussing sensitive information over the phone. By the way, I don't have any SpinRite stories to share, but I use it on all my hard drives at least once every other month. I'm a huge fan of GoToAssist.com, and in the first 30 days I used it on several family computers all over the U.S. My mother appreciated the help so much she offered to pay for my subscription. Which, he says, is a mixed blessing and curse because now I'm always on call for family tech support. That's wonderful. Thank you for that, Bill. We appreciate it. He raises a good point, too. And it does worry me. I see this all the time.

Steve: He does. Now, for what it's worth, I guess I have a security conscious banker, at least. There have been times when I have begged my banker to send me some details of something I'm doing by email, and he absolutely refuses. And he says, "Steve, frankly, nothing I could send to you would even make it out through Union Bank's email security. It would just be completely shut down." And he said, "I can fax it to you, or I'll talk to you over the phone."

And our listener Bill remembers correctly. Back in the old days of analog cellular phones, I was unwilling to even have a conversation with my attorney over analog cellular because scanners at that time were able just to simply scan a frequency, and there was no encryption at all. It was absolutely in the clear.

Relative to his question of what do you do to get the high-value professionals you work with to be secure, I would think that the greatest motivation would be to just sort of point out to them that they would decimate their entire practice if their accounts were compromised. If their account base got loose, if their Hotmail account was hacked, and the history of all their conversations became available to a bad guy, I mean, it would be game over for that kind of a professional.

I mean, we see this to an exactly analogous degree among large organizations whose servers are compromised, and all of the details of all of their customers, their personal identifiable information, credit card information and so forth gets loose, and it's a huge PR scandal. The organization is big enough that they're damaged by that; but they're not, you know, it's not game over. I would argue that if you have a much smaller, your lawyer or your accountant has an equivalent breach in their security, which seems to me everything we're seeing in terms of trends is aiming in that direction, that is, these kinds of threats evolving and moving downstream to smaller entities being attacked, really does create a vulnerability.

So I would think our listeners know enough about security and the dangers that are really present to just bring it up with your lawyer and accountant, say, you know, some of the things you're doing I happen to know are not safe. So think - you want, you need to appeal to their self-interest. And their self-interest is, consider what would happen if. It absolutely could happen. And I would imagine a professional who's reading the news, who's aware of what's going on, sees stories of this happening in various forms to other people, and thinking, well, there but by the grace of god, you know. And so you might say, eh, you know, there's some things you could do to tighten up your security. So bring them into the loop, I would suggest.

Leo: Let us move on to - yeah, bring them into the loop. Make them listen to this show. Move on to Question #7.

Steve: Eight.

Leo: Oh, no, 8. Greg Christopher says, "Ummm, no" to self-signed certificates: Steve, I thought I'd start off with your favorite word, "No." Unfortunately - Steve has even got a T-shirt, although I have mug that says "No," too. Unfortunately the title is in reference to a recent response you had to a listener who was concerned about self-signed certificates for the website Shadowserver.org. While recent conversations on the show might lead one to believe that identity verification and encryption are strange bedfellows and need not play soccer on the same field, this is actually not true. Without getting verification as to who is on the endpoint, the attack, which you explained on your ARP poisoning page, is very trivial.

The problem is, anyone can create a self-signed certificate to match the Shadowserver's self-signed certificate. The warning will not look any different in the web browser because we can easily match the text in the certificate - U.S., California, Shadowserver Foundation, mail.shadowserver.org - in our own certificate. That's if the user even bothers to click "show details," which he or she probably won't.

So with a situation in an open wireless caf or on the same Ethernet LAN or even a closed wireless network with a known password, you do indeed have a TLS connection established - to the hacker - who will phish away your email, username, password information and the like. Hopefully you don't use the same password for other sites you use. Twitter just posted a big warning about that to people, that their passwords have been compromised. And since people use only one password for all these sites, you might want to change your password.

Anyway, secure connections are only secure if you truly understand who is at the other end. Which is why Certificate Authorities actually perform an important role. And while a CA may not do a complete security analysis of those requesting certificates, they at least have the capability to make sure that you are not requesting a certificate with a name like "micros0ft" with a zero instead of an "O."

I must say it is really difficult to catch any mistakes on the show. And I also wanted to let you know that my copy of SpinRite has saved my bacon, and I'm an avid proponent of both SpinRite and Security Now!. Thanks very much, keep up the good work. Greg. Is he right?

Steve: He's absolutely right. And I thought that this bore repeating. He mixes definitions a little bit loosely for me, that is, when he says, "Anyway, secure connections are only secure if you truly understand who's at the other end." Well, okay. They're only authenticated if you truly know who's at the other end. But they're secure, meaning they're encrypted. So using the word "secure" sort of tries to straddle encryption and authentication. I'm always very careful to disambiguate the two.

We know that encryption means that the data looks like pseudorandom noise. And we know that authentication means we've identified, we've authenticated the identity of the other endpoint. So it is absolutely the case that it would be trivial for a man-in-the-middle hacker to spoof a self-signed certificate. Which is why I thought that Greg's point was worth making.

In fact, you could even automate this. If you saw, on the fly, you saw a connection being established to Shadowserver.org, and if malware or something else, if malware knew that the certificate was self-signed, then it could synthesize the interception certificate on the fly and send that back to the browser. That would pop up the warning that this is a self-signed certificate. But if the user knew that Shadowserver.org's mail server used a self-signed certificate, they'd go, oh, yeah, I know, I always get that when I send mail through this server. And they'd click on yes, fine, I know that. When in fact they've authenticated against a bad guy who can then intercept all their traffic.

So Greg's point is correct. That is to say that self-signing does open you to any kind of man-in-the-middle attack, specifically because it's not a certificate authority that signed the certificate, but rather the certificate signs itself, essentially, which gives you encryption, but really doesn't give you any authentication. And so this is the full bad news of what it means not to have any authentication, is that even though your data is encrypted, it could still be intercepted. And so that's a really good point.

Leo: Question 9, couple more to go. Rob McLean in Saskatoon has an idea for getting energy for nothing: Hey, Steve. Hey, Leo. On the recent CES episode you discussed a device which could power devices from WiFi signals. Was that you, or was that somebody else?

Steve: That was me.

Leo: It's laughable, obviously. I recently started studying electronics, and while doing so this same idea occurred to me. It seems that, if you took an AC signal from an antenna and ran it through a transformer, you could then turn a few millivolts into several volts. If you then step it through another transformer, you could ramp up the amperage. I haven't had the chance to test this out, but from what I read it seems to work. In the podcast you mentioned the math wouldn't work out. In the spirit of the current series on the podcast, could you explore why or why not this system works? Also, while researching I discovered Tesla explored this idea - yeah, I remember that - so it doesn't seem so farfetched. As always, thanks for the great podcast. Could you just step it up?

Steve: Okay. The trick here is to discuss the relationship between voltage, current, and power. Power is the constant, and power equals voltage times current. So by definition, for example, a watt of power is a certain amount of voltage at a certain amount of current. So when you - and, for example, a transformer with a differing number of windings on its primary and its secondary coils is able to change, for example, the voltage of a signal on the primary versus the secondary. For example, when all of us grew up running, who had a train track set, a set of...

Leo: Lionel. I had Lionel, yeah.

Steve: Exactly, Lionel. So we had a big transformer there which was taking the 117v AC signal and reducing its voltage to a much lower level, which then made it safe to stick it on the little three-rail tracks so that we wouldn't shock ourselves.

Leo: Right.

Steve: And it was a useful voltage for the train. So, but the problem is, a transformer doesn't create power. And in fact it's a lossy process. The transformer gets a little warm, which means you're actually losing some of the power during the transformation in the form of heat. And you may have noticed it buzzes sometimes, too. So now...

Leo: Yeah, and you smell ozone.

Steve: Yeah, you're actually even losing some acoustic energy. So there's a mechanical energy being lost. So it's a lossy thing to do. So if you step up the voltage, then you're going to get more voltage, but at less current, because the power minus the losses of conversion will be the same. So there isn't a way to, like, by hooking up some transformers in series, you can't step up the voltage and then step it back down and get something for nothing, essentially. And in fact in each of those processes you're going to lose some power.

I did misstate something in last week's episode, which a number of listeners mentioned. I used the term "a flow of voltage." And of course I meant current. I know the difference. So it was just something people who were listening very carefully to what I said picked up that I mentioned a voltage flow, when in fact voltage doesn't flow. The best analogy that we've used in the past is to think in terms of fluid, like water under pressure. If you have a hose with water, the pressure of the water, if you think about it, the pressure is different than the flow. If you could put your thumb over the opening of the hose, you can have a great deal of pressure behind there, but no flow. If you release your thumb over the opening of the hose, then water begins to flow.

So the word "current," just like it sounds, is like the current in a river or in the hose. Current is actually the volume of the flow of the water, and the voltage is the pressure that is sort of behind and enabling, inducing the current of water flow. So together you multiply the pressure and the flow, the voltage and the current, in order to get the power which that amount of pressure and flow are able to do. That is, the amount of work they're able to do. So there's a close analogy between the electrical circuit operation and something we're really familiar with, like the way water flows through a hose.

Leo: And one of the reasons Tesla's research never panned out is because he couldn't ever get it to work.

Steve: Yeah, he was a master of super high voltages and also high frequency. High frequency and high voltage. There's something called "skin effect" which happens with high frequency where the electricity stays on the outside, that is, on the skin. And so one of the ways he was able to do things that looked like it would just electrocute somebody, was that it was super high frequencies and very high voltages. So he'd have, like, fireworks dancing off of his fingertips, and his hair standing on end, and all kinds of crazy stuff going on. He was a master showman, also.

Leo: They both, both Edison and Tesla were archrivals.

Steve: Yup.

Leo: Our last question is really more of a plug for what's to come. Kenneth Musante in New York City, speaking on behalf of many listeners who wrote in, says "I'm so excited about the 'How Computers Work' series." This is what we started last week. I've always been curious about how computers work, but since I've only been around for about 30 years, the modern machines I've always known seem so far removed from the computers of yesteryear like your PDP-8s. To me they've always just been black or beige boxes. I have been using computers since I was a kid in the early '80s. My first machine, a Coleco Adam. Oh, yeah. Remember that?

Steve: Yeah.

Leo: I also know a lot - that was the stringy floppy that had. I also know a lot about computer history - Babbage, Colossus, ENIAC, and so on. And I know most of the basic principles of electronics. However, the conceptual gap between those historical machines and the iMac sitting on my desktop seems insurmountable. I would love to really know what's going on inside there. Even in college, no one was able to explain it to me in a way I could understand. I loved your talk about how the Internet works, and I'm certain if anyone can explain what appears to be such a complex topic, you're the guy. Looking forward to learning. Thanks so much to you and Leo for doing this for all of us. I second that emotion. And I know many other listeners do. It's fascinating.

Steve: I got a lot of really great feedback about last week's episode. And next week we're going to talk about - and we understand now about sort of the basics of logic gates. And I wanted to give people a sense for how bulky these things were, how with the lack of integrated circuits, what the challenges were that the designers faced. So we're going to look at the design of an early minicomputer in detail and understand when we come out the other side of next week's podcast exactly what is machine language.

Leo: Wow, that'll be fun.

Steve: It's gonna be great.

Leo: I like that, yeah. What language does that thing speak?

Steve: What is machine language?

Leo: Steve Gibson is the man in charge at GRC.com, a great website for people who love technology. Of course it's the place where you get SpinRite, and you really ought to go there just to pick up SpinRite. It's the most insanely useful program you'll ever use. If you've got a hard drive, you need SpinRite. GRC stands for Gibson Research Corporation, that's how you remember it. He also has a lot of free security utilities there, including ShieldsUP!. More than 85 million people, or shields, have been tested there. Lots of freeware. Information about that PDP-8.

And in fact, if you go to, in the other menu, the PDP-8 computers page, you can learn all about what Steve's doing - there's lots of great videos there - and find out how to order that front panel. If you're one of the people who wants to get Bob's front panel, last chance to do that. This Monday will be the last chance to order.

That's GRC.com. Also 16KB versions of the show are there, and Steve's notes, and Elaine's transcriptions and all of that. GRC.com. Thank you, Steve. It's been a great week. We'll see you next week...

Steve: Thanks, Leo, very much.

Leo: ...on Security Now!. Bye bye.