SECURITY NOW!

Transcript of Episode #249

Listener Feedback #92

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-249.mp3
Quarter size (16 kbps) mp3 audio file URL: http://media.GRC.com/sn/sn-249-lq.mp3

Leo Laporte: It's time for Security Now!, the show that covers your security, now, and privacy, too, with the guru of security, the man who started GRC.com, the Gibson Research Corporation, originally as a way to sell his one-and-only software product, SpinRite. But it's become a huge resource for anybody interested in security. Between the podcasts, of which there are now 249, but also the software that he gives away, the man who discovered the first spyware, wrote the first antispyware program, even coined the term "spyware," Steve Gibson. It's great to see you again.

Steve Gibson: Well, Leo, we set a pretty high bar for ourselves last week.

Leo: I should add to your credits "the man who created the portable dog" - I don't want to say "killer" because no dogs were harmed.

Steve: No dogs were harmed. And it's unfortunate that that's what - and when I was 16 years old, that's what I named the thing. And I considered, you know, fudging that for our audience and calling it the "portable dog trainer." But I thought, well, maybe I'll be forgiven and understood that, you know, I was 16. But...

Leo: We should call the "portable principal stunner."

Steve: Vice Principal Archibald.

Leo: Yeah. Have you heard from Vice Principal Archibald? Because I would think...

Steve: I actually wondered maybe if it would filter out to him. Haven't heard anything from him as a result of this. It could happen. He's probably still alive and kicking somewhere.

Leo: If you have not listened to Episode 248 of Security Now!, you may pause this one and go back and listen. Not that you'll need to for anything we talk about today. But it's just such a fun episode. And with an important moral, which is that you've got to encourage kids to create, to make. Not to sit passively and consume, but to create. And thank goodness we live in an era where the tools for creation, whether it's software programming, hardware hacking, are just everywhere. We've got a listener in the studio, Doc is in town for the Maker Faire, which is coming up this weekend in San Mateo, California, which is all about people making cool stuff. And I think that that movement is alive and well.

Steve: It's interesting, too, I mean, the fact that there is such a thing as a Maker Faire; that they're scattered around the country; that there's a magazine, Make, that supports it. I mean, that sort of says that there's something special about making stuff.

Leo: Yes.

Steve: I mean, it's not nothing. So I decided that we really needed to do a Q&A because it's been now several weeks since we've heard questions from our listeners. The mailbag incoming is full of really good stuff. So I didn't want to skip that, which was regularly scheduled for last week, which we did the special Episode 248 sort of in commemoration of the 50th anniversary of the invention of the laser. So I thought, okay, well, we'll change the parity of our Q&A and non-Q&A episodes by not skipping one, but just pushing it down a bit. So that's what we have for today is nine questions, some short, some long; some discussion; tons of security-related stuff to talk about. So I think we've got a really great podcast to follow, okay, nothing will be as great as 248, but we'll do the best we can.

Leo: It will be content-rich, and it will cover security this time, which is...

Steve: Indeed.

Leo: So item one in our security news.

Steve: Oh, Adobe, speaking of dog stories, Adobe in the dog house.

Leo: Again?

Steve: Yes, once again.

Leo: It's just - this sounds like a broken record at this point.

Steve: It's, well, I would say it's shocking, but the problem is with Shockwave. The good news is...

Leo: That's not widely used, really.

Steve: Exactly. That's the good news is that it is sort of their more powerful platform. It's more expensive to develop for. You need about a thousand-dollar designer program in order to create it. There aren't lots of third-party, in fact there may not be any third-party support for, like, alternative design platforms. So once again, this is a problem with Adobe Shockwave, not Flash.

Now, last time we had major problems with Shockwave, my advice to our listeners was just uninstall it. If you happen to have installed it by mistake or sometime in the past, it's likely that you don't need it. So hopefully you'll know if you do need it. Certainly, you know, a given corporation might have standardized on it. It might be something you have no choice to use. In any event, you could certainly upgrade to the latest.

Leo: Am I reading this right? There are 11 problems?

Steve: Yes. And I wanted to just run through these...

Leo: That's ridiculous.

Steve: ...to give our listeners a sense of how relatively horrifying these are. SANS put together in their most recent security vulnerability alert a sort of a summary. This is available on Adobe's site. It's available from other third-party sites. But I wanted to just quote from what SANS wrote because they sort of turned it into English in a nice fashion:

"The first issue is caused by a boundary error while processing Shockwave 3D block, which is one of the block formats in a Shockwave file. The second issue is a memory corruption vulnerability caused by a signedness error

while processing malicious Shockwave files. The third issue is a memory corruption vulnerability caused by an array indexing error while processing malicious Shockwave files. The fourth issue is caused by an integer overflow error while processing malicious Shockwave files. The fifth issue is a memory corruption vulnerability caused by an error while processing asset entries. The sixth issue is caused by a buffer overflow error while processing embedded fonts. The seventh problem is caused by a boundary error while parsing Director files. The eighth problem is a memory corruption vulnerability caused by an error while processing a four-byte field within record-type 0xFFFFFF49 within the 3D objects defined inside Director files. The ninth issue is caused when an application encounters signed values while parsing "pami" RIFF chunks."

Leo: Pami Riff?

Steve: Oh, you know her?

Leo: Oh, yeah. Went to high school with her, I think.

Steve: "The tenth issue is caused by an error while processing Director files during a memory dereference. We know about what memory dereferences are now from talking about pointers in our Fundamentals of Computer Technology series. The eleventh issue is caused by a signedness error while processing Director files." And, as if that wasn't enough, there are more "unspecified errors which can be exploited to cause memory corruption." In other words, just don't use this.

Leo: It's stunning that there are this many errors. I don't understand.

Steve: I know.

Leo: And these are - are they kind of related, or it seems like they're all different...

Steve: They're all very different.

Leo: ...parts of the code.

Steve: They're all very different. At the bottom of Adobe's page announcing the update, past, okay, past this vulnerable and prior, on both Windows and Mac, by the way, they have a list of thanks to all the various people that have brought these to their attention. So in fairness, these have been accumulating...

Leo: It's a rollup, okay.

Steve: Yes. These have been accumulating for a while from various sources. So this is their fix. So these vulnerabilities exist in version 11.5.6.606. Now, what's important is that this is not Flash. This is Shockwave. So the problem is there's some - and we've talked about this before - some jargon confusion because, for example, my Flash plug-ins in both Firefox and IE describe themselves as "Shockwave Flash." So that's not the problem. And those are back at version 10 something. If it just says Shockwave or Shockwave Player, then that's what you're looking for.

But I wanted to also remind our listeners of a cool service that Mozilla is now offering, but also a major news announcement about that. Mozilla has something that we've talked about before called the "Plugin Check for Firefox," where if you simply, in Firefox, you go to Mozilla.com/plugincheck, all run together, plugincheck - and you do need to, from a scripting standpoint, trust Mozilla.com. So if you're using NoScript you need to enable that. That's why it didn't work for me initially until I said oh, yeah, I've got to turn scripting on. Then it worked. And it will - it's just a beautiful facility they're offering to enumerate the plug-ins in Firefox and tell you how you're doing.

Well, when I did that this morning in prepping this, I realized, oh, I was fine except I needed to update my Acrobat Reader plug-in, which I did, and then everything was okay. The big news is they've decided to expand this service beyond Firefox, which is so cool. They're now offering it for not only Firefox, but Safari, Chrome, Opera, and IE.

Leo: Wow.

Steve: Yes. So Safari users, Chrome users, and IE, you can use - if you go, for example, and I did it this morning when I ran across this, if using IE you go to Mozilla.com/plugincheck, it works. It works in Internet Explorer. And it's a little less extensive...

Leo: This is so great.

Steve: Isn't this neat? It's a little bit less extensive in IE. And they explain that the reason is that IE's technology requires much more work from them, sort of individual customization per plug-in. So their support under IE is lagging behind what they're able to offer the other browsers.

But I just think, you know, hats off to the Mozilla group for doing this. I mean, really, they could easily be snarky and say we're not going to help anybody else because those browsers are competing with Firefox, obviously. But they're not. They're saying, look, this stuff matters. Plug-ins are hurting the web in general. And so since we acquire the technology and the knowledge, we're going to make it available cross-browser, which is way cool. So this is a major tip for today is Mozilla.com/plugincheck, no matter what browser you're using.

Leo: Man, I, thanks to it, found out that my Shockwave is out of date.

Steve: Yeah.

Leo: Wow. Maybe I'll just uninstall. When you download the Shockwave installer you get an uninstaller, as well. This might be a good time to run the uninstaller.

Steve: Yeah, I was going to say, you could keep it, you could certainly keep it around and put it back in if you find out you need it.

Leo: Right. Maybe you play Shockwave games or something like that. But...

Steve: Right. So if you do need it, you want to keep it current. But it makes no sense at all to have a vulnerable plug-in that you never use. That is, you may have - something may have said you need it three years ago, and inertia has left it in. I mean, another nice security feature we could dream about would be if things realized they hadn't been used for a long time, they'd pop up and volunteer to leave. Of course we're not going to see that anytime soon. But so we need to be proactive. And I would say if you've got Shockwave, and you don't know why, uninstall it because this is from Adobe. And unfortunately that's not good news.

Leo: Somebody in the chatroom asked a good question, I don't know if you have an answer, if User Access Control, the UAC in Windows will protect you against stuff like that. My sense is it would not because...

Steve: No, it will not.

Leo: ...you've already given permission to the browser to run.

Steve: Yes, in fact, someone asked also, I think it was a piece of mail that I scanned through and didn't quite make the cut for today, was would a firewall prevent plug-ins from misbehaving.

Leo: Right, right.

Steve: And again, a firewall won't because the plug-in is running in process. It's running in the browser's process. And we talked about also in our Fundamentals of Computing, in the multi-verse episode a couple weeks ago, what processes are. And these process boundaries are regarded as sort of autonomous units by things like firewalls. So when you have permitted Firefox, for example, or IE or whatever your browser is, to access the Internet, as you virtually have to, I mean, that's what the thing's for, you've given the browser permission to communicate to the Internet. Implicitly you've given all of its plug-ins permission to do the same thing.

So, unfortunately, your control for plug-ins is within the browser. And I didn't mean to really say "unfortunately." The fact is you can manage plug-ins easily under add-ons and plug-ins under Firefox. The same thing for IE. You can go through IE and look at all the paraphernalia that your Internet Explorer has collected over time and just say, I don't know what this is, I think I'm going to disable it until I think I need it again.

Leo: I'm sorry, I didn't mean to disappear on you. I was just looking.

Steve: Yeah. I was also catching my breath because we talked just last week or the week before about a critical problem in Safari regarding the way it handled - Safari for Windows, the way it handled pop-ups. And there was a means had come up that allowed pop-ups to be abused. Well, we've got another two problems in Safari which are so new that Apple has neither acknowledged nor responded to them. Yet there are technical details publicly available for the first of these.

So we are currently at 4.0.5 of Safari. And unfortunately that version, the most recent version from Apple, has two problems. The first is caused by - and we've talked about this before also - a use-after-free error in the way Apple's Safari for Windows handles references to Windows objects after releasing them, such that a specially crafted web page can be used to trigger this vulnerability, and a successful exploitation of it might lead to remote code execution. And again, technical details for this are available publicly. So Safari doesn't, as we know, doesn't have a large market share. But the danger...

Leo: On Windows, yeah.

Steve: On Windows. Yeah, exactly, on Windows. But the danger is that someone might know that a corporation or a group or a company or a user is a Safari user, and then do a so-called directed attack...

Leo: Spear phishing.

Steve: Yeah, I jumped over that because there was a different term that we used recently. Because it might not be a phishing attack.

Leo: Right, not an email necessarily, yeah.

Steve: I'm blanking now on what the term was. But we had a new term of art that we've started using that was, like...

Leo: Targeted hacks. Targeted exploits.

Steve: Targeted, yeah, there was a better word for it. Anyway, it'll occur to us, or someone in the chatroom will let you know what it was. So, and the second is...

Leo: Weaponized email?

Steve: That's it, weaponized email.

Leo: Thank you to WindowWasher, who was the first to get that one.

Steve: Yup. So someone could send you, if they knew you were using Safari, a piece of weaponized email...

Leo: That's nasty-sounding.

Steve: ...that could take you down. So the second issue is an information disclosure problem with the way Safari handles HTTP authentication credentials in an HTTP request that can cause some information to leak out of your computer. So not such a big deal to worry about as remote code execution, but still hopefully Apple will jump on this and bring us up to 4.0.6, which we'll let people know when that happens. In the meantime, there's no fixes available for these. So be careful if you're a Safari user.

Leo: On Windows.

Steve: Yes. And by the way, I just wanted to mention that I realized often we're talking about security updates, and other times we're talking about security news. I had previously been sort of merging them together. And I've decided I'm going to sort of break that out since they are separate issues. So that's all of our update stuff. Now in the news. There was a truly horrifying revelation which occurred recently. This is some security researchers at UC San Diego and also University of Washington have recently delivered a report. I mean, so recently I don't have it yet. But they did release some news about their results hacking car control systems. And the only way I can do this justice is to read this story, which was covered many places, and in this instance it's from the BBC. The News.BBC.com says:

"An investigation by security researchers found the systems to be 'fragile'" - that is, the control systems in cars - "to be fragile and easily subverted. The researchers showed how to kill a car engine remotely, turn off the brakes so the car would not stop, and make instruments give false readings. Despite their success, the team said it would be hard for malicious attackers to reproduce their work." So I want to make sure that everybody heard that. But this is - my concern is what this foretells. And we'll talk about that once I'm through reading this story:

"The team of researchers, led by Professor Stefan Savage from the University of California-San Diego and Tadayoshi Kohno from the University of Washington, set out to see what resilience cars had to an attack on their control systems. 'Our findings suggest that, unfortunately, the answer is "little,"' wrote the researchers from the Center for Automotive Embedded Systems Security. The researchers concentrated their attacks on the electronic control units (ECUs) scattered throughout modern vehicles which oversee the workings of many car components. It is thought that modern vehicles have about 100 megabytes of binary code spread across up to 70 ECUs." So 100 meg of code scattered across as many as 70 different electronic control units.

"Individual control units typically oversee one subsystem. But ECUs communicate, so that many different systems can be controlled as the situation demands. For instance, in a crash, seat belts may be pre-tensioned, doors unlocked, and air bags deployed." So there's a reason for these systems to intercommunicate is the point they're making. "The attackers created software called Car Shark to monitor communications between the ECUs and insert fake packets of data to carry out attacks. The team got at the ECUs via the communications ports fitted as standard on most cars that enable mechanics to gather data about a vehicle before they begin servicing or repair work. The researchers mounted a series of attacks against a stationary and moving vehicle to see how much of the car could fall under their control.

"'We are able to forcibly and completely disengage the brakes while driving, making it difficult for the driver to stop,' wrote the researchers. 'Conversely, we are able to forcibly activate the brakes, lurching the driver forward and causing the car to stop suddenly.' In one attack, the team transformed the instrument panel into a clock that counted down to zero from 60 seconds. In the final seconds the horn honks; and, as zero is reached, the car engine shuts off, and the doors are locked.

"They found that almost every system in the car, including engine, brakes, heating and cooling, lights, instrument panel, radio and locks was vulnerable. The team concluded that car control software was 'fragile' and easy to subvert. In some cases simply sending malformed packets of data, rather than specific control code, was enough to trigger a response.

"The team are presenting a paper on their results at the IEEE symposium on Security and Privacy in California on 19 May," which is the day before we're recording this, is yesterday, or Wednesday, because we're recording this episode on Thursday this week because, as you know, Paul and I swapped...

Leo: Yeah, which is very kind of you, thank you. And I should say very kind of Elaine. I apologize to Elaine, our transcriptionist, who has to work twice as fast today. Sorry.

Steve: Oh, yeah. So they said, "'Cars benefit from the fact that they are (hopefully)'" - and they put "hopefully" in their paragraph - "'not connected to the Internet (yet), and currently are not able to be remotely accessed,' said Rik Fergson, a security analyst at Trend Micro. 'So in order to carry out a successful attack you would already need to have physical access' - you would currently need to have physical access - 'to the vehicle, as a break-in or as a mechanic seem the two most likely scenarios today. As cars and everything else in life, up to and including even pacemakers or refrigerators, become steadily more connected and externally accessible, research such as this should be taken increasingly seriously by manufacturers,' he added.

"'This represents an opportunity to head off a problem before it starts, in the not-too-distant future, as it may result in a real risk to life.'" Which is why I felt it was really important to share this. I mean, our listeners already know how terrifying this news is because we are, I mean, there's this tremendous drive to add features to our technology. And you can, I mean, we know that there's XM radio now in autos that is sending data to - so that we're able to listen in our cars. There's beginning to be technology that lets you check on your car. I know there's some web-based stuff that allows you to have some sort of interface with your car in some situations.

So unfortunately, I mean, I just - I hope that the people who are building these systems are listening to Security Now!, and they're being insistent enough with their management about the kind of safeguards that need to be put in place. It's already dispiriting to learn that it's possible to have, I mean, we know the problems that Toyota has been having with their brakes. And presumably this is buggy code. But here we see that it's possible for, in a research environment, for just accessing through the access ports that mechanics use, that it's possible to deliberately cause a car's brakes to be disengaged so that the foot pedal no longer engages the calipers on the disks.

Leo: So that's a hack. But we should emphasize, you need physical access to that port. You have to get in the car and reprogram it.

Steve: Well, we should emphasize what we know, which is that the researchers did have physical access. So, yes. I don't want to scare anyone away from driving.

Leo: Nobody's going to be pointing something at you, a ray gun, a portable dog killer at you, brake killer at you as you drive by. You have to get in the car. That access port's usually right under the steering wheel on all modern cars. And they have to plug into it. At least that's the hack that they were doing.

Steve: Yes. And so the concern, again, I don't want to over-alarm anyone. But Leo, we know where these things go. I mean, it's funny because as I'm reading about them talking about a malformed packet, it's like, wow, that's what we had with routers 10 years ago.

Leo: It's software. Software is hackable, often.

Steve: Yes. And unfortunately, when we hear that there's a hundred megabytes of code, it's like, okay, I'm going to keep my current car running as long as I can. Just, you know, because I like the old, the nine-year-old technology I have in it.

Leo: You have a pre- what is the date that these things became common? It's been a while.

Steve: Well, yes, it is. I mean, I have a 2001 car. So it's nine years old.

Leo: And it doesn't have the port.

Steve: Oh, yeah, I think it does.

Leo: Yeah, I think '98 is when they started putting those ports in.

Steve: I think it does. I mean, I think that's what they check. But...

Leo: '96, yeah.

Steve: But again, what happens is, as we've seen before, it's like, oh, these ports are nice. These ECUs are handy.

Leo: Right.

Steve: Let's put them in the seat belt. Let's put them, like, 70 of them apparently scattered around now in many cars, all little nodes. I mean, it's like the Borg, little nodes communicating with each other. And it's like, oh. Again, I wish there weren't - okay. First of all, what would be the motivation? It's difficult to see the motivation. And motivation does matter because we know that people are hacking, that bad guys hidden through anonymity on the Internet are hacking people's computers to get their credit card numbers and identity and authentication information in order to, ultimately, somehow, to make money, to steal money, or send spam or something. So I hope there isn't motivation for this kind of auto hacking. Frankly, Leo, it wouldn't surprise me to learn that it's possible today because that's the way these things are. The level of complexity that these vehicles have obviously now achieved to me makes them seem, as these researchers said, extremely fragile. And that's just not good news.

Leo: Right.

Steve: So we'll hold our breath. As we've seen also, it takes motivation. And so we'll hope there isn't nefarious motivation.

Leo: This will be something on "Law and Order." I mean, it's a way to murder somebody.

Steve: Yes. Yes.

Leo: But, I mean, remember "North by Northwest"? They got Cary Grant drunk, and they disconnected his brakes, and they sent him down a road. So that was 50 years ago. I mean, they had to cut the brake or let the brake fluid out. But if you have physical access to a car you can make it dangerous.

Steve: Well, yes, you can do anything, yes.

Leo: And that's true in hacking, too, that a lot of times we hear about exploits that require physical access to the computer. My philosophy has always been, if somebody has physical access, you're screwed.

Steve: And my point is, physical today, nonphysical tomorrow.

Leo: Right. Because it's software.

Steve: Well, and because there's, I mean, there's a tremendous desire for connectivity. I mean...

Leo: That's the issue. Because right now you can't get into a car remotely because it's not online, it's not, I mean...

Steve: Well, we hear about OnStar and, you know, oh, sir, we know you've been in an accident. We're going to deploy the 9/11.

Leo: Well, I talked with Ford about this, as a matter of fact, CEO Alan Mulally of Ford about this, and they make very sure to separate the entertainment computer from the car computer, and that there is not merely a firewall, but they're not connected systems.

Steve: Good, good, good, good, good.

Leo: So because of that; right? You don't, if you're going to open connectivity, and boy, they really are increasing it, you cannot allow that connected computer to speak to the car computer. That would be dangerous.

Steve: Yeah. Well, and we heard, same good intention with the high-security government networks that were going to be not connected to the Internet. But they ended up somehow being connected to the Internet. And that's caused problems.

Leo: Of course, somebody's pointing out, yeah, pointing out that the OnStar system can disable the car. The OnStar operator can disable the car. That would seem to me...

Steve: That's what I'm saying, Leo.

Leo: ...kind of an issue. I don't know how, and I think there are rules about can they do it when it's running, et cetera, et cetera.

Steve: I know, I know. And where did they get their security certificates? Who signed them? And has that been spoofed? I mean, you put together a blended attack, and it's like, oh, boy, this stuff relies on infrastructure that the designers assume is robust. And then elsewhere the security community goes, oh, that's not quite as strong as we thought it was. And then somebody with the motivation - again, it takes motivation. And I just don't want to have any motivation. But, unfortunately, this is, I mean, I hate being right about this kind of thing. But, oh, it's really - this really deeply creeps me out.

Leo: Yeah. You've got a good point.

Steve: Yeah. Now, Google and WiFi.

Leo: Oh, yeah.

Steve: Yeah. First I learned of this was when a journalist with Reuters called and said, Steve, have you heard about the Google admission that they were promiscuously - actually that's my word, I explained to him what promiscuous mode was on a WiFi radio - that they were capturing publicly available data and storing it, recording it on hard drives. And I was quickly brought up to speed and talked to him about what I thought this meant.

So for those of our listeners who may not have heard the story, it's been making big news almost for a week now because it was last Friday that I talked to this reporter. Google, in their Street View technology, and we talked about this a couple weeks ago relative to my realization with my iPad that the Skyhook service, which Apple apparently uses, was able to capture SSIDs and MAC addresses from WiFi hotspots as they were driving around with GPS, mapping where all these things were. And I thought it was very cool that this was available even in encrypted networks because the SSID and the MAC address are available in the clear.

Leo: Has to be in the packet or it won't work.

Steve: Correct. So what we found out was that, I believe it was Germany that was pressing Google because the Germans were very upset, just sort of felt a little creepy from a privacy standpoint, were apparently really pressing Google for exactly what data it was that they were capturing. And that forced an admission from Google that, whoops, well, we didn't really intend to, we didn't mean to, we didn't want to, but it turns out that, despite all of those disclaimers, we were capturing the payloads of the WiFi data that our Street View cars encountered as they were roaming around Germany, and storing them on disk drives, and we have all that. So the reporter from Reuters said, "Steve, what does that mean?"

And I said, "Well, it's funny you should ask because we've talked about this issue a lot on Security Now!, the podcast I do with Leo Laporte." I explained that, well, that could be the websites people were visiting, the email that they were transacting. Very often, if they're using POP or IMAP protocols, that is, not web-based mail typically, but regular sort of earlier protocols, their username and password would be in the clear. Not supersensitive stuff, which is generally deliberately encrypted by their connection, if not - and in this case not by the WiFi network. I said, but, you know, radio is radio. It's being broadcast. This stuff is in the clear.

Now, I did hear in part of Google's explanation for how this happened, a plausible source of, like, code. Apparently some other researcher doing something else years before had written some code that did do promiscuous capture, that is to say, it simply sucked in everything that a WiFi radio could receive and stored it. And when, years later, a different group who were doing the Street View project said they kind of looked around Google's massive project and software repository, it was like, oh, look, over here is some code that we could use that somebody wrote before. So they just kind of grabbed it, in sort of typical open source mode, and dropped it into their technology for Street View, and saved themselves reinventing the wheel.

Now, what this code did was record all the payloads of all the packets, rather than only what they really needed. As we've discussed before, what they really needed was the beacon's SSID, the hotspot's SSID, and the MAC address tagged with the current GPS coordinates, and presumably the signal strength. Because if I were doing this, I would incorporate signal strength in so that, as the car was moving, you'd get a sense for - you could actually do very good triangulation over time to get a sense for the physical location of this node whose SSID and MAC address you've acquired by looking at the signal strength as the car drives around.

So I think the problem is that hard drives are huge. I mean, and lord knows Google must have some sort of serious quantity discount they get on buying hard drives, with indexing and caching the Internet and Gmail that apparently has endless storage and so forth. I mean, Google's probably got more storage than anything else on the planet. So hard drives don't cost much. They probably weren't worried about saving hard drive space. So they were probably recording packets and maybe tagging it with this extra metadata, SSID, MAC address, well, actually that would be part of the packet, and like the GPS information, and maybe just who knows, I haven't looked in detail at their Street View technology; but, sure, they could be doing all of this processing in the vehicle as it drives around. Or they could just massively...

Leo: Capture it all, yeah.

Steve: ...suck all this in, yes.

Leo: Ah. I bet you that's what they were doing. And then that would explain it.

Steve: That's my, yes, that's my guess is that they had a relatively brain-dead massive capture operation where they were just sucking this stuff in, tagging the packets with the GPS metadata. And then offline, or off the street, rather, like back at headquarters, then they would reprocess the data and do all the computations necessary to geolocate the specific MAC address and SSID node. I mean, that makes sense. So they didn't have to do it that way, but that was probably the path of least resistance, which makes sense they would do. So I guess my overarching feeling is, hey, the best thing about this is it serves as a wakeup call about unencrypted WiFi.

And in other news, and I can't remember whether I mentioned this, whether we had a Q&A, I think we actually do, somebody mentioning a recent judgment by the German government about unencrypted WiFi. But to me this helps raise awareness of the relative exposure that people have, not having their wireless networks encrypted. I mean, we've talked about it all the time.

Leo: Now, I mean, it's fragmentary data they got. They probably got nothing of value.

Steve: True. They would have - and, see, that's just it. Germany's freaking out over this. And my sense is, first of all, I really believe that this wasn't deliberate. I can see how they would have followed a path that would have allowed them to capture this due to what they explain. It's entirely plausible to me. I mean, and why would they care? I mean, they've got as much data as they could ever ask for just being Google.

Leo: Right.

Steve: They don't need to drive around and suck up random packets that are unencrypted as they're driving by. It's not like they set up permanent listening posts and were sucking this in.

Leo: Right.

Steve: So, yes. I think it's, on one hand, much to do about nothing. Their explanation makes sense. And but I do hope it serves as a bit of wakeup call.

Leo: Absolutely, yeah. I mean, if you still have an unencrypted - the point, in a way, is these people are broadcasting that stuff anyway. It's out there. All Google did was the same thing you do when you listen to a radio station. It's being broadcast.

Steve: Yes.

Leo: So you really ought to not broadcast is the message.

Steve: Exactly. It's radio. And if your keystrokes and your username and password, the sites you're visiting are available, well, again, I hope that the story serves to raise awareness of this, that Google doesn't get tainted because of what they did. I mean, yes, they could have arranged not to store this data. But...

Leo: They have now, by the way, and they've deleted it all.

Steve: Yes. And they've got a gazillion trillion terabytes of data, so what's a little more?

Leo: Right. Right.

Steve: Okay.

Leo: Ooh, this is a bad one here, this one.

Steve: Yes. Yes, yes, yes.

Leo: This one is really creepy.

Steve: Yes, it is. And it turns out this is not the only such site. There's a site called Paste-It.net. And so if you use Google to do a site-specific search...

Leo: Oh, I know where you're going with this one. I didn't see this in your notes. This is really interesting. Yes.

Steve: Yes. And if you - so, for example, if you put Google.com/search? and then q=site:paste-it.net, then +visa, or you can just go to the Google.com, click advanced search, specify the site...

Leo: You don't even have to do that. If you just do site: it'll work even in a standard Google search.

Steve: Oh, okay, yeah.

Leo: Site:paste-it.net+visa will give you this result.

Steve: Yes. Now...

Leo: I know where you're going with this, too.

Steve: Well, yes. And in fact, Leo, if you do that, just, I mean, click on the first link. I have. It's safe. You will see a page full of people's credit card information, real people, their names, their addresses, their...

Leo: What the heck is this?

Steve: Their CVV2. This is a site that the bad guys use for buying and selling credit card information. It is a site designed to allow people to paste information. Then they get a unique URL. And unfortunately Google indexes it.

Leo: And I love it that they have Google ads on the right for Capital One Visa cards. That's nice.

Steve: Yeah. I mean, this is literally, look at it, it's actual credit card information. There they are with their CVV2 code and the name and the street address.

Leo: Oh, this is terrible.

Steve: I mean, it's horrifying.

Leo: You know, if you do the same thing with SSH, people are publishing their SSH keys in this, too. This is a fun search to do. With a lot of things [laughing].

Steve: I know.

Leo: Wow. So the bad guys, yeah, because, I mean, it's not like an individual user put his card in there. This is a list of hundreds.

Steve: Oh, it's hundreds. And those links, I mean, when you do the Google search, you can sort of see what the dates are. There are some that are only - that are fresh.

Leo: Right.

Steve: So what happens is, from packet capture, just like we've been talking about, or from malware, some guy running a botnet collects all these.

Leo: And he's publishing it.

Steve: He gets a buyer, well, he gets a buyer for it. And he uses a site like Paste-It.net...

Leo: Right, because it's anonymous.

Steve: ...as his anonymous intermediary. So he drops all that stuff there, then gets a unique link which, you know, he gets payment from his buyer, sends the buyer the link. The buyer clicks on the link, brings it up, copies the page, and here's a whole ton of recently captured, fully, I mean, all the information you want in order to charge people's credit cards maliciously. So I just thought I would share that little bit of happy news with our listeners.

Leo: One way this could be solved is if the folks at Paste-It would use robots.txt to say don't index this stuff.

Steve: Yes. That would be a help. Now, see, or maybe do, I mean, if this is a problem, that we've got anonymous drop sites like this, I guess obviously there are...

Leo: Well, they're very useful. I mean, I use them all the time to share code or whatever. So that's fine. And then there's drop.io, which I use. But they should absolutely block Google indexing.

Steve: Yes. Yes. So VeriSign has sold its authentication services to Symantec.

Leo: Oh, how interesting.

Steve: I just thought I would - it's a, like, $1.82 or $1.28 billion dollar purchase.

Leo: Wow.

Steve: I'm sort of not happy about it, only because, I don't know, I've never been a big fan of Symantec. And, I mean, I've been buying - VeriSign's my SSL cert provider. And remember the VIP program that we've talked about extensively? That's part of it. And VeriSign's whole PKI infrastructure. So PKI, VIP, and SSL, I guess that's Authenticode also, and I'm an Authenticode user, have all been sold to Symantec for a huge, one point something, I think it was a $1.28 billion purchase. I received two pieces of email yesterday because I'm on VeriSign's various lists as a purchaser, as a customer of theirs. So, and it's been in the news. And I guess maybe I need to change my attitude toward Symantec. It's an old attitude, back from the days of Gordon Eubanks, who was the founder. And...

Leo: I like him. You didn't like Gordon?

Steve: I heard some stories about him.

Leo: Oh, interesting.

Steve: From people who really did know him, out shooting squirrels. It's like, okay, well, that's too bad.

Leo: Interesting. Yeah, because Gordon was originally the CEO of, it was not PFS, it was another great software company that I really liked. And I got to know Gordon then.

Steve: Was it personal software?

Leo: It wasn't. It was another, they did...

Steve: Because that was Fred Gibbons.

Leo: Yes, it was Fred Gibbons. It was a DOS text editor. Oh, what was the name of his, well, I'm sure I can Wikipedia Gordon Eubanks. But I knew him in that context. And after he went to Symantec we kind of lost touch. But he worked at Digital Research. He was a partner with Gary Kildall. And Naval Postgraduate School. He wrote CBASIC and BASIC-E. Let's see. Well, I guess it was Symantec. Q&A. That was it. Q&A was an integrated database and word processor with natural language queries, which I loved.

Steve: That's right, yup.

Leo: You remember that? It was really cool.

Steve: Yeah, yeah.

Leo: And then he became the CEO and president of Symantec. He was at Oblix...

Steve: So, VeriSign has sold that off. I just wanted to let our listeners know. So just as another little security news...

Leo: For what it's worth.

Steve: I did want to mention in errata that my handle on Twitter didn't last out the day last week.

Leo: I noticed you changed.

Steve: Yeah. There were, I mean, it's significant, when I learned about the problem with handle length for retweets, that that's a problem because they take up space in the text of the tweet. And there was enough comments about, well, yeah, this is a spelling test. And I thought, okay, we don't want to give people spelling tests.

Leo: AgileSynapse, right.

Steve: So AgileSynapse has been replaced just by SGgrc.

Leo: Now, there is somebody named SteveGibson on Twitter that's not you.

Steve: No, it's not. And in fact I just saw something from him, actually, and I meant to send a note to him this morning. But he's, I guess now that I've joined, he's just been flooded with people who think that I am him.

Leo: So you are not SteveGibson. You are SGgrc.

Steve: I'm not SteveGibson. He's been on Twitter for three years and no doubt likes his handle as it is. I'm just SG, as in Steve Gibson, and GRC, as in Gibson Research Corporation. So SGgrc, that's me.

Many people did seriously ask for the plans for the portable dog killer. You wouldn't believe all of the justifications that I heard for rats in the backyard...

Leo: Please, please, please, please. If you're not smart enough to invent it yourself, you're not smart enough to use it, my friend.

Steve: Well, that's precisely the lesson I was going - I mean, you've got me, Leo. You know me well enough. I was thinking about Oppenheimer, and the lesson being exactly that. If you're not able to design it yourself, then you don't pass the test of being responsible enough in its use. Because it really, you know, something like that, as I demonstrated, could cause some problems. So I think it's - first of all, I don't have the plans. And I went rummaging around in war surplus store bins in order to get them, like the specific pieces. So I wouldn't even know how to specify it these days. And I just - I wanted to share the anecdote. But unfortunately I did put on the map the fact that you could have a lot of fun with a sonic beam weapon.

Leo: Well, give us a clue. What frequency audio does it generate?

Steve: The problem is, I had a Heathkit scope that I built in Christmases before. And there was really no way - I didn't have a frequency counter. There was no way to really calibrate the sweep. My guess is that it was like in the 15 KHz range.

Leo: Very, very low. That's sub-audible. I mean, above audible.

Steve: No, most people can hear to 20,000.

Leo: Oh, okay. Not me.

Steve: So 20,000 is about the - well, as we get older we do - our high-frequency cutoff of our ears drops lower and lower. But when like you get an audible test where they give you those tones...

Leo: Right, you can hear it. It's almost a physical sensation.

Steve: Yeah, with 15,000 I think you can probably hear. It was high, but it wasn't supersonic, by any means.

Leo: Well, what is the frequency of those - I guess they don't do it anymore. But in the old days the motion detector alarms would send out a very, very high-pitched sound you could hear.

Steve: Well, that's technically ultrasonic. That's the...

Leo: When I was a young man I could tell. I could feel the sound.

Steve: No kidding.

Leo: Oh, yeah.

Steve: Well, I have - I do remember you telling me that you had extremely high-frequency hearing.

Leo: In the old days.

Steve: Well, in fact - well, no, not even so long ago. Remember when I wrote the speech compressor using the Speex codec.

Leo: Oh, yes, I did hear some differences there.

Steve: And we did the AB testing. I could absolutely not hear any difference. And then you said, oh, I can hear the difference. I said, oh, come on. And like we played a little game. And you got it every time. So...

Leo: There is a - my kids have it. You can download a ringtone, a sound for your iPhone. They call it the Teen Annoyer.

Steve: Yes, I heard. And, like, parents can't hear the phone ring.

Leo: So the kids use it so they can notify each other in class. The teacher won't hear it, but the kids will hear it.

Steve: I love it.

Leo: Yeah. See, this is kind of along the same lines.

Steve: Yes, very much so.

Leo: And Henry uses it. And we actually did it during a TWiT when we had a live audience some years ago. And Dvorak and I are sitting here, blithely playing the tone. And the younger people in the audience are going, "Agh, make him stop, make him stop." So they really...

Steve: No kidding.

Leo: ...can hear it, yeah.

Steve: Wow.

Leo: They say mostly people under 25.

Steve: I love it. That's great.

Leo: That's a kind of a sonic...

Steve: Audio, well, it's an audio filter. Yeah, it's sort of a sonic firewall because we can't hear it, the old folks can't hear it, but the young kids still can. That's neat.

Leo: Yeah. And I'm trying to find the spec for that tone. But I'm thinking it's around 15 KHz.

Steve: One of our listeners, who listened to Episode 248 last week, sent me the URL of San Mateo Electronics, sure as hell on 42nd Avenue. And I thought, oh, my god. So it's SMElectronics.com. And I clicked the link - well, actually I typed it in myself because you know about me and links - and put it into my browser. Up it came, 42nd Avenue Electronics. And on the home page it says "Since 1961." And I'm thinking, well, they had been there 10 years before the portable dog killer was built using the mini box that I...

Leo: That you found. Now, I have to say that you did not hear it, nor did I, but I played a 15 KHz tone moments ago, and the chatroom is going "Ow."

Steve: No kidding. Do it again.

Leo: And by the way, if you're listening to the MP3...

Steve: Oh, yeah, it might not be able to get through.

Leo: ...you probably won't hear it.

Steve: I'll bet it can't get through Skype.

Leo: It's probably Skyped, not - but I'm in the studio, and I can't hear it.

Steve: Wow.

Leo: On the other hand, it did - people in the stream can hear it, which is interesting. Flash - listen, listen. Nothing. They hate it. Now, I'm going to play - let's play something - let's play 10 KHz. You could hear that. Can you hear that?

Steve: No.

Leo: Skype's rolling it off, then.

Steve: Yeah, I'm sure they are.

Leo: Yeah, Skype's rolling that off. We're doing a little hearing test. All right, kids. 20 KHz. The chatroom is going crazy [laughing]. Now, and I did it as a - I have to say I did it as a blind test because I did not say I'm going to play the 15 KHz tone.

Steve: Right. They really did hear it.

Leo: People complained, they really heard it. It's very interesting, isn't it.

Steve: Where are you getting these different frequencies?

Leo: This is NoiseAddicts.com. And it's called - a blog that's called...

Steve: I'm sorry that I asked. I'm sorry that you said.

Leo: It's the online music and audio magazine. And there is a post, a blog post, you'll have to go back a year, "Can you hear this?" And he ranges from 8 to 22 KHz, little waves. And actually a very interesting idea.

Steve: So 15, so you could imagine - imagine if that 15 - oh, you can't hear it.

Leo: I can't hear it.

Steve: I think maybe the portable dog killer was down at 10, then. Because, I mean, it was, I mean, adults could hear it. Mr. Archibald had no problem hearing it from across the quad, so...

Leo: And, you know, probably - I don't know what MP3 rolls off. But MP3 is a very interesting codec that doesn't just - it doesn't necessarily roll off frequencies, but it does some interesting things. And...

Steve: It's using psychoacoustic science...

Leo: Exactly.

Steve: ...in order to understand what it is that we do hear and don't hear in complex waveforms. And it gets rid of complexity that we just - that doesn't matter.

Leo: And the chatroom is just going crazy. They are hearing it. They say their dogs are barking. Oh, my god, I have a headache. I don't how much of that is facetious and how much of that is real. But I will no longer play anything. And it may be that - we don't know what the Flash, I don't know what the Flash media encoder is doing to it. So it may be - because if you're watching at home, you're watching on it.

Steve: Well, now you've made me curious. So I'll have to go over there and listen to it myself, so...

Leo: We'll have to call this episode the "portable listener killer."

Steve: So I did get a neat note from a Security Now! listener, Mark Gottselig. His note was, "SpinRite Saved My Ubuntu," which is not something I've heard often. He's in Calgary, Canada. He said, "Hello, Steve. I've been a SpinRite owner for several years now, ever since I heard about it on Security Now!. I've used it in its maintenance mode the whole time, running it" - not full-time, but never other than maintenance mode - "running it monthly on my own PC, and running it on friends' and family machines when they had me do an upgrade or repair for them, and have gotten several of them to buy copies for themselves after much praise."

And, hey, Mark, I've got no problem with that, as our listeners know, thank you. That is to say I have no problem with him running it on other people's machines, encouraging them to buy their own copy. "However, I've never had a need to use it in recovery mode until yesterday. I've recently installed the newest Ubuntu release, 10.04, and have spent the last week off and on configuring and learning this new OS I've only had a fleeting need for previously. I ended up having a lockup and had to force a reboot with the power button on the laptop." And he said, "I didn't know about Alt-SysRq-R USB." I suppose you do, Leo, but I don't know what that is. But some keystroke invocation, apparently.

Leo: Yeah, no...

Steve: And he said, "When it rebooted I got an error message and a command prompt. Not knowing what happened, I tried a couple of different things to no avail. I figured before I'd try anything more serious I'd give SpinRite a whirl. I booted my SpinRite CD and started to watch some TV with the laptop propped on the couch beside me," he says. "I was very surprised to see the DynaStat screen appear after 30 minutes or so..." That's SpinRite's dynamic statistics technology, which it drops into when it needs to, to do sector-level repair. And so he said, "...and watched excitedly to see SpinRite do something I'd never seen before." Then he says, I love it, "This novelty wore off quickly. So I left the computer to work away at the drive overnight. I checked this morning, and

SpinRite had completed. I did a quick reboot before work and was shown the Ubuntu login screen that I had been trying to get for several hours the previous day. Thank you so much for such a great product. SpinRite saved my Ubuntu." So thank you, Mark.

Leo: And that's an important point, that SpinRite is not operating at the file system level. It doesn't know from operating systems or file systems.

Steve: So it runs on Linux just fine.

Leo: Yeah, because it's looking at the sectors on the hard drive.

Steve: Precisely.

Leo: It does have to run on a BIOS-based machine, though, because you use BIOS calls. Which means EFI-booting machines, like the Macintoshes, it does not work with.

Steve: Not today.

Leo: Is somebody - you mentioned at one point somebody was looking at a way to do that.

Steve: I've heard people anecdotally say they've succeeded. And I've even had them take screenshots. I mean, I've seen photos of SpinRite running on Macs.

Leo: Really.

Steve: Because apparently there is a way of getting Mac to support a BIOS as part of its Boot Camp procedure. But I've never pursued it myself.

Leo: Right, right. And somebody's asking in the chatroom if SpinRite can be put on a USB drive with a boot loader. Can you boot it, in other words. And that would be, if your BIOS boots from USB...

Steve: It will.

Leo: Absolutely.

Steve: And people, I mean, I have it on my own USB key that I carry around with me that I've mentioned before. Yeah, SpinRite often - there is a - in fact, I've read a story a couple times where someone had a USB flash drive that was too large for his BIOS to recognize, but he used a smaller one, which the BIOS did recognize, and it just booted and ran SpinRite fine.

Leo: We should point out, though, and this is a different thing, that checking a USB drive with SpinRite is going to give you limited results.

Steve: Oh, no no no, don't want to do that.

Leo: Because the USB hides the interface.

Steve: Well, and there's, I mean, SpinRite's really oriented toward physical magnetic media.

Leo: No, but, I mean, a spinning USB. You're right, obviously a flash drive don't do.

Steve: Oh, oh, a USB-connected hard drive.

Leo: A USB-connected hard drive.

Steve: Yes. Well, we do have success stories with it. I'm not as bullish about it because, as you say, Leo, the USB interface only does reads and writes. And SpinRite is able to do a much better job if it, like, has physical low-level access to the drive. That's far superior. But people say, I mean, as a last resort, SpinRite can still work.

Leo: Worst case you'd take out, I mean, if you really - if it didn't work, take it out of the USB enclosure. It's still an SATA or an IDE drive, and then put it in a PC, then SpinRite it.

Steve: Yes. And I have had people successfully do that with their Macintosh. They've taken the drive, as a last resort, out of their Mac over to a PC motherboard and run it there, and SpinRite will fix the drive that way.

Leo: And even iPod drives, I've been told.

Steve: Many, oh, remember, yes, we had a lot of people. There was one guy that we read who had a large collection of iPods because he became the iPod dumping ground for all of his friends. And then he realized, hey, SpinRite fixes these. And he fixed a whole bunch of them and gave them back their music that was...

Leo: Got to get the interface and all that. Steve, I have questions. We're already - we're an hour in.

Steve: We're an hour in.

Leo: Do you want to do a few, anyway?

Steve: Absolutely.

Leo: Why not? Let me pull up the questions here. We'll get to as many as we can before, well, we've got 40 minutes.

Steve: Yeah, we do, before your next...

Leo: But nobody's ever complained about the show being too long.

Steve: No.

Leo: You can always pause it. That's the beauty of it. Paul Stob in Nashville, Tennessee wrote about the portable dog killer, your episode last.

Steve: And let me just briefly interrupt and say I got a ton of email from our listeners. I really thank them. Clearly the episode was the most popular one we've ever done. I actually had people rating it. Someone thought the Vitamin D episode was number one, this one was number two.

Leo: Somebody who's not into security, obviously.

Steve: Somebody thought it was the other way around. So I just chose one of so many notes because, as our listeners are about to see, this achieved my dream for one of the main reasons I wanted to share that last week.

Leo: Excellent, excellent. So Paul writes: Just wanted to add my voice to the likely thousands of people emailing you about the latest Security Now!. It was absolutely wonderful. My wife, god bless her, usually hates Security Now!, for reasons I'll never understand. I might have an idea, but - but she absolutely loved this episode. I had it on in the background Saturday morning. It was a little Garrison Keillor-ish, wasn't it.

Steve: It was so good.

Leo: In fact, she thought so highly of it that she made our eight-year-old son listen to it. He's a typical eight year old who loves to play videogames. And like most kids his age, he plays them way too often. But after listening to the show, he went right into our garage and started disassembling some of his old, broken electronic toys. He wanted to see how the components fit together, and I'm pretty sure he had grand ideas about making his own sonic blaster. Anyway, thank you for the wonderful episode; and, please, give us more like it. Wow, that's great, Steve. That's really great.

Steve: So, yeah, I just - and the next question feeds into this, and something that I wanted you to contribute to, Leo. So, and again, I want to thank everybody for writing. I really - I appreciate that they appreciated the episode. I felt like I was taking a little bit of a risk going so far off topic. But I think it worked, so.

Leo: Yeah, you know, we do a lot of security information. We did even on that show. And I think it's good to talk about your life experience because you've got some. Mike York in Seattle, Washington writes about our mention of FIRST robotics in Episode 248. I mentioned the FIRST USA robotics competition. Thanks for the mention of FIRST. I've been involved for seven years serving as a team mentor, a judge, and a referee. Hey, it does make a difference. A significant difference. We've seen phenomenal growth of FIRST FRC teams here in Washington state the last few years, and expect it to continue as more businesses see the value of FIRST and provide resources for teams, scholarships, and competitions.

As a referee at the Seattle regional this year, I had the best seat in the house to see, hear, and smell the robotics competition. It's encouraging to see the excitement in these scientists and engineers of the future. Keep up the good work, and great podcast. While Security Now! may not be an appropriate podcast for a segment on FIRST, it may be a good subject for one of Leo's other endeavors.

Yeah, I think we're going to - I want to do a FIRST show. If you go to USFIRST.org, you can read about the FIRST Robotic Competition, FRC. And there are different programs for all ages. And the high school team, which is of course kind of the varsity league of FIRST, is what I think I'm going to do. And they say it's about $6,000. That includes all the gear and so forth. So I want to fund this FIRST team in my kids' high school. And then I want to do a show. I have to get waivers from all the parents and everything because I would like to do kind of a reality show, following them week after week as they design and build these robots.

Steve: Well, I just - I love the notion of a robot because I do think there's a little something lost in a software-only solution. I mean, we talked about programming, and programming's interesting. The reason videogames are compelling is that they have that - they have at least some real-world tie-in. But what's beautiful about a robot is that it merges the physical world, I mean, some sort of battery and motor and also the computerized controller and software world together. And so you can have relatively simple hardware. It doesn't have to be powerful and exotic, just, you know, a couple motors mounted on a platform with wheels. And then you give it the brains in software. So but you get something physical, something tangible.

Leo: Right.

Steve: It's like these lights that are blinking behind me.

Leo: People love those lights. You want to explain those briefly so people...

Steve: Well, I was just going to say, they're machines. They're PDP-8 computer emulations. But they exist physically. It would have been far easier to have just an emulator on the screen that looks like those. But those things actually exist. They've got switches and buttons and lights and knobs. And, I mean, the tangibility of it really makes a difference, the fact that it exists physically. And so I just - this idea of messing with a robot, because this just isn't expensive any longer, a couple motors and, I mean, there's lots of resources on the web. These little controllers, like the PIC chips, are only a few dollars now.

There's even Lego that, if you want to back away and not even get your hands very dirty, there's, like, Lego robotics kits where you get to stick this thing together and then, again, do the programming in a simplified programming language. But again, do something. I just, if you can break that inertia. Also I'm trying to think what it was in the past, there's something that I talked about that a lot of people wrote back, sort of like, oh, I'm going to go do that with my son. That'd be a perfect thing to do with my son. And I don't remember now what it was. But that's another idea.

Leo: Or daughter. Or daughter.

Steve: Or daughter, yes. The whole - there's a problem, I think, just cross-generational. The eight year old wants to play videogames. But, boy, if he wanted to, like, build something with his dad, or dad and daughter - and actually I did get some mail from dads who had daughters who were listening to 248.

Leo: Good, good.

Steve: And who wanted to engage them in this way. So again, I think that the robot connection, there's a hook there because you don't have to build something really fancy. Even some basic mechanics that you then give a brain to with programming. And then you get that real-world thing. It's like, oh, look, it's actually moving on the floor.

Leo: Lego Mindstorms are great. In fact, there is a Lego Mindstorms Robotics FIRST Competition. And I know that our high school participated, did very well. So that's a very easy one. And they're suggesting in the chatroom, and I absolutely should mention Arduino, which is a really interesting open source electronics prototyping platform that includes a processor, has its own programming language. People are building all sorts of interesting things from Arduino. We did an interview with the Arduino, one of the Arduino designers on FLOSS Weekly, and I encourage you to listen to that, or go to Arduino.cc. This is another great way you can get started with kits and robotics. And it's really, really cool.

So I think there's more stuff out there than ever before, really. I mean, we had Heathkit when we were kids. But there's a lot of cool, kind of hardware-software things out there right now.

Steve: Yes. And all I would ever, I mean, all I am suggesting is break the inertia. Do something.

Leo: Yes, do something.

Steve: It's all out there. But don't let it stay out there. Bring it inside.

Leo: Right. Question 3 from listener Matt. He says, "Please, Sir, can I have some more?" Episode 248 was fantastic. Oh, and you're right about ctrl-c, the copy-paste bug. It's been like that for a few years. So much so that it's second nature for me to always now press ctrl-cc. And that always works. But it's a pain in MS Office because it brings up a multiple-paste toolbar. So everybody's responding to this. I mean, apparently it's something people are really having happen.

Steve: Yes. I wanted to drop this in for Matt, mostly as a placeholder and reminder. Many of our listeners have responded that they were so happy to hear this brought up.

Leo: Not just you.

Steve: Because they've been thinking it was just them for a long time. And one person wrote a lengthy piece of email where he's convinced this happened at Service Pack 3.

Leo: Oh, interesting. Of XP.

Steve: Of XP, and it's in Vista and 7, that Microsoft did deliberately, in their security enhancements for Service Pack 3, they changed the way the keyboard hooking technology works in order to thwart some behavior of keystroke logging. And that it's his belief - and this is not confirmed, but I just wanted to share it - that that was the boundary; that Service Pack 2 works fine, reliably, and that it got broken somehow subtly when Microsoft went to Service Pack 3 and beyond, that that was the boundary. And it had something to do with the way Microsoft increased the security in order to thwart keystroke logging. So I don't know whether that's true or not, but I thought that was an interesting thought. And I know that Paul and Microsoft are pursuing this. So if you think of it when you talk to Paul again...

Leo: I will ask him, yeah.

Steve: Have them take a look at that.

Leo: They are, apparently. I don't know what's going on with that. Question 5, Jason in Winnipeg, Manitoba, Canada wonders and worries about magic packets: Hi, Steve. I was troubleshooting my network adapter, and I came across a network setting in Windows 7 I've never seen before. The checkbox said "Allow a Magic Packet to wake the computer." Should I be worried about the magic packet? Is it some strange secret Microsoft backdoor we don't know about? I did a bit of reading on the magic packet, and now I have a bit of understanding of it. But I'd appreciate a Gibson explanation, if possible. Thanks for the great netcast. You and Leo are doing a great service for the security community.

Steve: Something we've never talked about.

Leo: I never heard of.

Steve: The Wake-on-LAN.

Leo: Oh, I have seen that even in BIOS settings.

Steve: Yes. And in fact that's where it has to be. In some cases you can put it in your network adapter because it'll then write it into the firmware of the network adapter, typically as a BIOS setting. Wake-on-LAN is a really interesting technology that allows you to essentially have your computer turned off, completely off. On the other hand, many people have probably noticed that our computers are never completely off any longer. And it's a little distressing the first time you notice, for example, that with your computer off there's still, like, a little light on the motherboard that's on; and that, like, network adapter, if your network connector has little LED monitors, they'll still be flickering. It's like, wait a minute. If my computer is off, how is this stuff on?

Well, the LAN adapter is still powered up, specifically for the purpose of allowing your computer, if it's enabled, to be powered up upon receipt of a so-called "magic packet." What's magic about it is the payload that a LAN broadcast packet carries. We've talked about MAC addressing several times, even in this hour. The MAC address is the Ethernet, the 48-bit Ethernet address of the packet on the network. But in several instances it's necessary for an adapter on the network to be able to call to everyone.

For example, ARP protocol uses that. When a computer is coming on the LAN for the first time, it sends out a broadcast asking for the - it knows the IP, for example, of the gateway, but it needs to get the MAC address of the gateway. So it'll send an ARP broadcast that is addressed to the broadcast address on the LAN, which is heard by every NIC, every Network Interface Card on the LAN. What happens is, the so-called "magic packet" is any broadcast packet where somewhere in the payload, somewhere in the data payload of the packet, there are six bytes' worth of all ones. That is to say, FFFFFFFFFFFF, six bytes of all ones, followed by 16 repetitions of the MAC address of the NIC whose computer you want to wake up.

So the idea is, if some device on the network wanted to wake up a given computer - the computer is off. And remember that there isn't an IP protocol on an Ethernet. There's only the MAC address. So you have to, in advance, you need to know the MAC address of the computer you want to wake up because it's not until the whole TCP/IP stack exists that it has an IP address, and that's defined in the software. What we need is something which is defined in the hardware while the computer is, independent of operating system, completely turned off.

Well, that's the MAC address. So if you know that in advance, you just put a packet onto the LAN, sent to all of the adapters on the network, containing the special six bytes of ones, followed by 16 repetitions of the 48-bit MAC address. If it's been enabled in the BIOS and/or the adapter, and that machine has the ability to be awakened by such a packet, the adapter will see that, scan for that string, see that it matches its own MAC address, and wiggle a little line on its interface to the power supply, turning the computer on.

Leo: That's quite clever, actually.

Steve: It's very neat, yeah.

Leo: Yeah. And nothing - and completely harmless.

Steve: Completely harmless. Nothing to worry about. The magic isn't dark magic or black magic or evil magic. It's good magic.

Leo: It's clever.

Steve: But if you, I would say, if it's not something you're actively using, turn it off.

Leo: Oh, yeah. I turn it off, yeah.

Steve: I do, too, because that's not something I need. So if you see it, disable it unless you know you need it, as is standard computer security advice. Turn off what you don't know you actively need.

Leo: Sometimes people call me on the show, say my computer wakes up in the middle of the night. And there's a lot of reasons it could do that. But I wonder if sometimes - do you think it could be possible that a stray packet could somehow do this?

Steve: Well, this isn't transmittable unless your router is specifically configured. It's not something that...

Leo: It's not routable.

Steve: Right. Because it only works within your local LAN, not in a WAN, in a Wide Area Network setting. There are routers that can be configured to send out Wake-on-LAN packets. I would guess that their computer is in standby, and something like Microsoft Security has awakened it in order to update itself.

Leo: Or a cat moved the mouse.

Steve: Yeah, exactly.

Leo: That's usually what I tell them. First thing I ask them. You have a cat? Question 6 comes from Vegard in Norway. He asks about hosted versus self-hosted blog. So apparently, I didn't know this, but you've started a blog. You mentioned you were going to.

Steve: Well, I think I mentioned...

Leo: And he wants to know why you're using WordPress.com and not hosting it yourself with WordPress software.

Steve: Yeah, I think I did mention that I was going to, and I do plan to.

Leo: Oh, you haven't done it yet, though.

Steve: Correct. My concern is a couple things. I did look at what it would take to host my own WordPress blog. And unfortunately, the first thing you need to do is install SQL Server.

Leo: No, MySQL, MySQL.

Steve: I don't care. The last thing I would ever allow into GRC's network.

Leo: Yeah, because MySQL has all sorts of injection attacks.

Steve: Yeah. And the other thing is, I mean, I would love to host my own. But I thought about it. I mean, I have UNIX servers. That's where my newsgroups are hosted, for example. So FreeBSD is the UNIX that I had chosen years ago. It's been good to me. So I could do that. But it's like, okay, wait a minute. What kind of rat hole am I going to go down? And what value am I really going to add? And then it's backing it up, then it's blah blah blah and so forth.

And I just thought, you know, it makes much more sense, even though I'm going to have less control, it looks to me like WordPress.com is doing a good job. And so it was - I wanted to bring this up because it's an interesting question that I could imagine many people would ask themselves. Do I do this myself, or do I use a third-party hosting provider? And for me, much as I'm able to, it would take more time than I expect, because everything I do does, which would delay everything else that I really need to be getting on with, like finally getting going on CryptoLink, but I've got to get the other things finished first. And why? So I have it running on my own server, big deal. I mean, I did look at WordPress, and it does allow me to alias my own domain to theirs. So it will be...

Leo: Won't be obvious where it's hosted at all.

Steve: Well, correct. And for ease of use it'll be something at GRC.com. So I get the benefit of using my own domain as the anchor for it, yet all the other mess is theirs. And there just isn't enough value that I could provide by doing it myself. So that was the decision I made. And not an easy one. But I'm sure the right one.

Leo: I'm not sure I would recommend WordPress.com. Nothing against it, although they will put - you know they'll put ads into your stuff.

Steve: Oh.

Leo: And there's probably arbitrary JavaScript code also. You might want to look at Squarespace.com, which is a similar hosted solution, but it's a paid solution. WordPress.com is free.

Steve: Can you pay WordPress?

Leo: Oh, you might, yeah, I'm sure you can.

Steve: Okay.

Leo: And it may be that they then turn off the ads and so forth. But Squarespace is a great solution, too, and a sponsor of the network. So I just want to mention...

Steve: Oh, Squarespace.

Leo: ...Squarespace.com. I think they're very easy to set up. They run on a Java - they're basically Java based, and they run a VPS system. So it scales really well. It's cheap. And they do a good job. And they can do the same aliasing with a CNAME and all that stuff.

Steve: Okay, I'll check it out.

Leo: Yeah. I'm a fan. We do - I'm moving my blog over there, and we do our in-house TWiT blog there. Inside TWiT's on Squarespace.

Steve: Cool.

Leo: Question 7, Richard Doyle in Sydney, Australia stumbled upon your legacy project. I don't know what that is. Dear Steve, I'm 32, and I've only been listening to Security Now! for a few months, but I'm quickly catching up. Your explanation over the last several weeks of the fundamentals of computer architecture, organization, design and evolution over time has been accessible enough to inspire me to genuinely want to learn more and more about this entire area myself, from very first principles.

Have you ever thought of writing and publishing a book encompassing in greater detail everything you've explained - and would like to explain - in the current Security Now! series on the Fundamentals of Computing? Every other resource out there, mostly books, are dry, boring, and many assume a level of knowledge that most people just don't have. And for the most part every other resource out there is techie from the start. Not a bad thing, but we are badly in need of something that can begin to explain a thing in an extremely simple way, then scale up in plain language to the relevant level of detail. Other authors seem to enjoy an abundance of technical jargon for its own sake. And the people you've inspired through your current series in Security Now! are left with no entry point into this wonderful and amazing field.

Please consider, Steve - blah blah blah blah blah. How a computer works - the substance is there. Kindest regards, Richard. I don't have a Twitter account. You know, there is - my good friend wrote and has kept up to date a wonderful book called "How Computers Work." I think you probably know him. I'm trying to remember his name. He's been doing it for years. Let's see if I can find it on Amazon. And it's beautiful. It's a very - it's Ron, Ron White. He's been doing it for years.

Steve: Oh, my goodness, yeah.

Leo: You know Ron; right?

Steve: Yeah.

Leo: This is kind of the definitive book on this. And it's done with great illustrations. It may not be as tech- it's not as technical, in fact, as what you're talking about, Steve. But it is definitely aimed at the nontechnical, and it's a good start, if you're looking for that kind of thing. Incredible illustrations. But Ron is a smart guy. I mean, the detail in here is fantastic, and he has kept it up to date, which I really admire. So I think this would be a good start.

Steve: I agree. And it's available today.

Leo: Right. Would you like to do a book?

Steve: Well, no.

Leo: Yeah, I didn't think so.

Steve: But I've wondered what - this is going to sound strange, and I'm 55 and still have a lot of life left in me. But I've wondered what I'm going to do when I'm 75. That is, you know...

Leo: A good question.

Steve: And because I really do believe that, if you retire and sit on the patio in a rocking chair, you expire not long afterwards.

Leo: Seems to be the case in many cases.

Steve: And we know that I spent a chunk of time in the last year sort of looking at antique machines. I built the PDP-8s. And I also spent a lot of time researching instruction sets. I looked at field-programmable gate arrays, which I referred to last week as being these fantastic electronic building blocks which can be used for defining hardware out of software. And, for example, one of the things that people are doing is they're implementing processor instruction sets in field-programmable gate arrays, FPGAs, like taking classic instruction sets and creating computers out of these field-programmable gate arrays. There's something called OpenCores.com or .org, OpenCores.org, which has a lot of these. And so I thought about that.

And I also thought about these antique machines, the PDP-8 and the PDP-11s that I have. The problem is, they're not interfaced to any contemporary peripherals. I mean, you have to have a teletype, or maybe a serial interface. But what are you really going to do with them? And so I've sort of just - everything was sort of in a big mashup. And then I looked at instruction sets. And I sort of, like, surveyed the evolution of instruction sets over time. And all of this sort of ended up giving me the incentive to do this Fundamentals of Computer series that we've embarked on.

And for a while I was thinking, well, I was thinking the PDP-11 was the right instruction set, sort of like something that would be fun to program. And then I thought maybe the VAX. There was one instruction set from a company called National Semiconductor. They had the NS32000 series, which unfortunately never got off the ground. But in many ways I think it's the best instruction set ever designed. It was fun and nice to program, eight general purpose registers, a very regular instruction set, just beautiful. And then I thought, okay. I'd like to program the instruction set. But the chips don't exist anymore.

So then for a while I was thinking, okay, I could write the software to put this chip into an FPGA, a field-programmable gate array, basically create this chip that no longer exists out of contemporary silicon. The problem is that, when I thought about, okay, what kind of performance would I get, well, the things that have been done by, like, Intel and AMD to squeeze performance out of current processors are just over the top. I mean, it's unbelievable the technology that is in these things, with multiple parallel execution paths and multi-cores and pipelining, and even optimization of instruction sequences, and branch prediction, where it guesses based on local knowledge what path your code's going to take. And it's just daunting.

So I realized that, if I were to create this idealized processor that would start off probably as the instructions that National Semiconductor designed - five years after DEC's VAX, by the way, so they had five years of experience with the VAX instruction set and sort of tweaked it a little bit to make it better - I might do the same, you know, tweak it some more. The idea would be to create this ideal instruction set. And but if I put it into hardware, it would never perform like a contemporary machine because there's no way I'm going to invest the unbelievable resources that an Intel or an AMD has.

And then it hit me that, if I emulated that instruction set, this ideal instruction set, in a contemporary processor, like the current Intel, the current Intel architecture, then I'd be, in machine language, I would emulate another machine, sort of like PASCAL, and we've talked about p-code, like a pseudomachine. But because I was writing the emulator in machine language, and because I was writing it for hardware, the Intel architecture, which is already so unbelievably powerful, I would end up with an amazing amount of performance of this, like, the most beautiful instruction set I've ever seen, that I've, I mean, like, that there is, in my opinion. And so then I thought, okay. That's what I would want to write my operating system in. And that's what I would want to create an environment in.

And so my plan for retirement, my legacy, is to essentially create an entire open source free environment around the most ideal, beautiful instruction set that we've ever had, and write an entire world, a computing world in that - meaning assembler, editor, operating system, environment - with the goal of teaching a low-level operation of all of this stuff. Because it's hosted on contemporary hardware, everyone gets to play with it for free. And because it's a virtual machine, it gets to live forever. All anyone would have to do to port it into the future is write that little interpreter for the instruction set, and then everything else is available. So at this point, 20 years before I'm ready to start, that's what I think I'm going to do.

Leo: That's exciting. I look forward to it. I thought - so you've got a few projects, actually, for when you retire. I think you're going to be very busy in your 70s.

Steve: I want to be busy in my 70s.

Leo: Me, too.

Steve: I want to make sure I don't just sit around and decay. So you and I will be doing the podcast Episode 3,927.

Leo: Exactly.

Steve: Actually more than that. Well, anyway.

Leo: So one of the reasons I like my business is because there's a long tradition of people in their 70s and even older, look at Paul Harvey, still working. Especially radio because your body can fall apart, but you can still sound good.

Steve: How old is Jerry Pournelle, by the way?

Leo: Jerry's in his 70s. Jerry's going strong. He's writing his column, Chaos Manor. And, I mean, I love Jerry. So you're right. You can keep going.

Steve: Yup.

Leo: I plan to. And if you don't - you've got to keep it smart up here, but you've also got to - and I know you're doing both. You've got to keep the body in shape and...

Steve: Yup.

Leo: Larry King, how old is he? He's in his 70s; right?

Steve: Yeah. Bad example, though.

Leo: You know, if I'm in my 70s and on my eighth wife, I'm going to figure...

Steve: Good point. There's something still working there.

Leo: ...there's something going on. An anonymous listener suggests, let's design a network. I really enjoy your program and especially am impressed by how you take the time to explain the fundamental technology to give your listeners a deeper understanding of the weekly topics you cover in your Q&A and your regular episodes. If I may make an episode suggestion, it might be worthwhile completing the let's design a computer series with one or two episodes on networks and the Internet.

Oh, I agree. I agree. You talk about how networks work from time to time, but never as comprehensively as you do with computer systems. Considering the network-based nature of most exploits these days, I would greatly enjoy one or two episodes dedicated to a comprehensive explanation of how networks work and the Internet. I can think of no better person for this task. Thanks for the great program. I'd like to add my vote to that.

Steve: We're going to do that. Listeners who have been listening for almost five years will know that we did some of that in the beginning. But it was five years ago. And I know that there's been a churn of listeners. We've got listeners who have not been with us since the beginning. And this is such a fundamental core domain of technology, exactly as this anonymous listener suggests, that I think he's right. So we have a bunch of other things sort of already in the queue, things like my analysis of LastPass and so forth, some things I've already got planned, I want to get through. But probably around the time we start in on year six, I think that going back over and taking our time, starting at the fundamental basics of packets and routing and networks and the Internet would be another great series. So we're adding that to the queue.

Leo: Yeah. And really you can start at fundamentals on that, even. And I think that there's a lot to be understood. And frankly nowadays, as the old Sun Microsystems slogan, the network is the computer, nowadays really a computer without a network is not a significant device. It's the network that...

Steve: Fundamental.

Leo: Yeah, really is fundamental to the overall operation. Question 9, Ashley Black in Reading, Berkshire, England brings us the Wonderful Glitch of the Week. Subject: Skyhook. [In a British accent] Hello, Steve. I've just listened to the podcast where you described the Skyhook WiFi location finding system, and it totally explained a strange bug I've been having with my iPhone. When using the Maps application on my iPhone - I'll go in the normal mode here.

Steve: Although I really did like that.

Leo: Did you enjoy that?

Steve: It's amazing what a difference that...

Leo: [With accent] It sounds so much more cultured and intelligent. I have British friends, and they all sound so literate. When using the Maps application on my iPhone - the reason I stopped doing it is because the British people are howling in pain.

Steve: Yes. It's like, oh, my goodness.

Leo: Just like when we see a British actor, sometimes, not - Hugh Laurie is an example of somebody who can do it perfectly.

Steve: I was going to say, it's amazing when you see him interviewed separately, when he's not doing the accent.

Leo: A good actor can do a good accent. But I am not that. When using the Maps application on my iPhone at work, it kept showing my location as the old address of the company. Hmm, that's interesting.

Steve: Okay, now listen to this carefully. This is a wonderful glitch.

Leo: We moved to a new place three months ago, and this is very confusing because it was five miles from the new location, where I am now. Then the iPhone got GPS, the signal came in, and the location would be correct. But for the life of me I couldn't figure out why it would think I was at the old office. Obviously, that answer now is obvious. The wireless access points we used in the old office are now in the new office, but Skyhook's database still has the old location. I wonder how long they take to update their database, or does iPhone Skyhook API do it for them? Thanks. Long time-listener and SpinRite faithful, Ashley Black. That's really - of course that would happen.

Steve: So isn't that cool. We talked about how the SSID and the MAC address of the hotspot, the access point, was once located by Skyhook trucks driving around. And so the database knows the physical location of those access points. Well, the company moved months ago. Those access points are now five miles away from where they were. But Skyhook has not updated its database. So for anyone near those access points, any technology, probably Google's street maps, I would imagine, is going to be the same way, Street View technology. Anything that once roamed around the streets logging the locations of those access points, the device sees the access point, says where am I, and the database says, well, you're where you were five months ago. So anyway, I thought that was just very cool. And he asked a question that was the first thing that occurred to me, which is, if the phone got access to GPS, so it knew exactly where it was, wouldn't it make sense that it could...

Leo: Just send it back.

Steve: Yes, that it could say, whoops, wait a minute here, this is more than a small error. So I don't know, maybe you want to put it in pending somehow, or say, well, maybe it's time to roll around this area again. I don't know how they handle that. But it absolutely makes sense that GPS-enabled equipment, which is also WiFi-enabled, could be essentially doing the equivalent of what rolling trucks around are doing, but doing it on the fly, dynamically. Which, wow, that'd be really cool, too.

Leo: I think there's privacy considerations that keep the iPhone from updating the Skyhook database.

Steve: Ah.

Leo: I would guess. Right? That's the kind of thing that I can just see the headline: "iPhone tells company where you are."

Steve: Yup.

Leo: Without your permission. So, I mean, if they popped up a thing that said...

Steve: Good point. Very good point.

Leo: ...can we update Skyhook or something, maybe. But that's an inter- I'd be very surprised if it did, just for that reason alone.

Steve: You're right. So, for example, it could know that Skyhook and the GPS in the phone are disagreeing substantially, pop up a notice saying, hi, our geolocation data from WiFi no longer matches GPS. Would you mind if we update this, our database to the sky? Skynet wants to know about Skyhook. And so it's like, uh, no. I like you not knowing where I am.

Leo: Yeah, exactly. I mean, that's - people are very worried about that. And Google this week announced an API for its Latitude. If you use Google devices - I don't think it works on the, yeah, I guess it does work on the iPhone. It knows where you are and sends that information back to Latitude in real time. It's not even like a check-in. It's, like, as you move. And there's an API for that now. And I think that people - there are three million users. And I bet you many of them don't remember signing up and allowing it. So this has become - you're going to see a lot more about location stuff. This is just the beginning of real paranoia over that. Steve, we've come to the end of our great nine questions.

Steve: Yes, and a great episode. I hope everyone enjoyed it. Number 250 next week.

Leo: Holy cow.

Steve: Yeah.

Leo: You know, you scared me. You said "as we start our sixth year." But that's, like, three weeks off.

Steve: That's 260.

Leo: 260. It's not far.

Steve: Not far.

Leo: Amazing. Amazing. If you want to know more, GRC.com, the Gibson Research Corporation, GRC.com/securitynow. You can get 16KB versions of the show, the show notes, full audio quality, transcripts, all the information. Steve puts it up there as a pro bono. Really appreciate your doing that, Steve. He also has a lot of great other, you know, things at GRC.com: SpinRite, the world's finest hard drive and maintenance utility; a lot of free utilities. GRC.com. Steve's Twitter handle we must not hesitate to tell. And you've got to start following some people, Steve. You're still only following two people.

Steve: I'm just nervous about so much stuff coming in. I'm actually reading everything that comes in from my own listeners, from my own followers.

Leo: If you read the @ stuff, that's pretty good, too, yeah. And if you see somebody who is really consistently giving you good information, follow them. Anyway, follow Steve, Twitter.com/SGgrc. That's easy. Steve Gibson, GRC. SGgrc is his Twitter handle. Steve, we'll see you next week on Security Now!.

Steve: I did want to make a mention that since we're recording this a day late, this collides with Elaine's own transcribing schedule. The block that she normally allocates she's not able to fill. So transcripts may not appear, for example, until next Monday. But don't worry if you don't see them.

Leo: They'll be there.

Steve: I will get them up as soon as Elaine is able to do them for us.

Leo: Yeah. I should mention that we normally do this show, and you're invited to watch us do it live, every Wednesday at 2:00 p.m. Eastern time, 11:00 Pacific time, 1800 UTC. It's at live.twit.tv. We do make the video available on iTunes and everywhere else podcasts are available. Probably the easiest way to find it and subscribe is to go to the TWiT page, TWiT.tv/Security Now!. TWiT.tv/sn. You'll see "Subscribe" dropdowns for audio and video, makes it very easy to add us to Google, Zune, iTunes, and everywhere else. Thanks, Steve. We'll see you next time.

Steve: Thanks, Leo.