SECURITY NOW!

Transcript of Episode #299

Going Random, Part 1

Description: This week's security news and events took up so much time that we didn't have time to cover the entire topic of "Randomness" in security and cryptography. So we split the topic into two parts. This week we open the topic and explain the background, problem and need. Week after next we'll plow into the solutions.

High quality (64 kbps) mp3 audio file URL: http://media.GRC.com/sn/SN-299.mp3
Quarter size (16 kbps) mp3 audio file URL: http://media.GRC.com/sn/sn-299-lq.mp3

Leo Laporte: It's time for Security Now! with Steve Gibson, Episode 299 recorded May 4, 2011: Going Random.

It's time for Security Now!, the show that covers your security and privacy online with this cat here, this man, Steve Gibson of GRC.com, the Gibson Research Corporation. He is our hero, our knight in shining security.

Steve Gibson: I'm a cat this week.

Leo: A cat. Yeah, I'm hep now. I've turned hep.

Steve: Yeah. Very cool.

Leo: That was a little Sammy Davis, Jr. channeling there. Hey, Steve, how are you today?

Steve: Hey, Leo, great to be with you again, as always.

Leo: Thank you. Today we're going to cover a subject that's kind of fundamental to, not only security, but computers in general.

Steve: It was something we have - we've touched on many times in the last five and a half years, but never really focused on it. And I have been focused on it personally because I've had some problems to solve in this regard. And I thought, you know, this would be something really great to talk about. And that is the whole issue of randomness - what is randomness, why does crypto depend upon it, and where do we get it? Now, it's a big enough topic, and we've got so much news this week, as we have been in the last few weeks, that I thought, you know, I'm not going to be able to do this all in a single podcast. So this will be part one of what we're calling "Going Random."

Leo: Going random, I love it.

Steve: Where we'll set up the problem and get into it, but stop short of going into the solutions which have been devised, which are really interesting. I mean, I think - I'm sure I've spoken on the podcast, for example, that somewhere at Sun Computer there was at one time cameras looking at lava lamps.

Leo: Yes.

Steve: Because lava, the flow of the wax in a lava lamp is a chaotic, unpredictable process. And so they were literally digitizing lava lamp images as a source of chaos to feed into their need for random numbers. So anyway, we've got a great podcast this week - lots of interesting updates and news, and the first half of the issue of random numbers in cryptography and the need for them and the problems of, like, that we have of generating them.

Leo: It's actually a fascinating subject in computer science. And as you say, germane.

Steve: Well, and it's a problem because computers aren't.

Leo: They're not. Random, they're not.

Steve: And that's the problem is, you know, every time you add two and two, you really want to get four. But sometimes you really wish, you need something random. And computers can't produce randomness. They absolutely can't.

Leo: So first I guess we should say happy anniversary, Steve. I can't believe it's been a year.

Steve: I know. I mean, it just shocked me. I looked at my profile on Twitter about three weeks ago, and it said "You've been a member for 49 weeks." And I thought...

Leo: Wow.

Steve: ...wow, okay. And so I'd had it on my own radar to mention to all of our podcast listeners. And then this morning I got a birthday tweet from Twitter. So they knew also.

Leo: They birthday tweet? I never got a birthday tweet.

Steve: So one of the things they're doing now is to say you've been a member for a year.

Leo: Happy birthday.

Steve: And so it was exactly, it was May 4th, the day that we're recording this is May 4th, 2011. It was May 4th, 2010 that I created the @SGgrc account on Twitter. Technically I guess I joined a little, a few days before that because remember my first account was AgileSynapse. And then I realized...

Leo: Yeah. That was good.

Steve: I liked that a lot, actually. Elaine moaned, I'm trying to think of - mourned, mourned the passing of AgileSynapse. She really liked that name a lot. But I realized that in retweeting you really want a shorter handle. It's also easier for people to type and so forth. But if you're going to retweet, then, like, I'm deliberately, often, if I'm posting something that I think may be retweetable, I'll leave room at the end so that there's room for an RT@SGgrc without people having to go in and shorten what I've produced. So it's much nicer if it's short.

But I wanted to take this moment just to say, I don't know if people have sensed an improvement in the quality of our - the top of the show stuff. I feel like it's gotten better in the last year, and maybe even more recently than that, because I've been a lot more active. And our listeners have been, those who are following me and who are in Twitter, have been really effective in sending me tweets for stuff they run across. And so, I don't know, I feel like there's more interest in our top of the show stuff. And it's really a consequence of me being so well informed. While I'm doing other things there's a constant Twitter feed that I keep an eye on of people saying, hey, did you see this, did you see this? Now, yes, sometimes I get 300 of the same things. But that's better than getting none of them. So I just want to say thank you to our listeners for taking the time to keep me up to speed. I really do think it improves the podcast, which was the whole reason that I wanted to do this. And so it's effective.

Leo: You can't go wrong putting more content in. I think our, you know, we know our audience. They're smart. They want more meat. And so more meat's always good.

Steve: Yup. So everybody is involved in our updates. We haven't had any updates for a couple of weeks actually. That category has just been empty. So everyone's there today. We have Mozilla, who has released since we last spoke to our listeners a new version of Firefox, where they fixed a bunch of things across all three of their currently maintained upgrade trains, both Firefox 3, 3.5, and 4. They fixed 53 flaws in the browsers, 12 of which were rated critical by them. The flaws addressed in the new version of Firefox 4 include a pair of issues in the graphics libraries that could be exploited to bypass certain security protections in Windows. So those are important to do. You want to keep Firefox, of course, up to speed.

Chrome, meanwhile, in its sort of background stealthful updating, has brought itself current, addressing 27 vulnerabilities in the Chrome browser. And so, let's see, so I said bringing the stable build of Chrome to v11 for Windows, Mac OS X, and Linux. And interestingly, we know how Google pays security researchers for reporting problems they've found. Well, these 27 vulnerabilities cost Google $16,000 in bounties to 11 different researchers who reported 17 of those 27 flaws. So they are paying out money for people who find problems and report them. Which I think is great. None of these were critical, but 18 of them were given high severity ratings. So Google, you don't really have to do anything to update Google. There's no restarting necessary. Normally when you just start it up, it silently updates itself. I know every time I look it's already got the latest and greatest from them. So it's doing it continuously.

We did have an update over on the Adobe Acrobat Reader, Acrobat and Reader side. A "CoolType.dll" had a pretty serious memory corruption remote code execution vulnerability which affected their versions of Reader 10.0.1 and earlier for Windows, 10.0.2 and earlier for Mac, and Acrobat 10.0.2 for both Windows and Macintosh. And this is about as bad as it can get because they fixed a number of problems. But this one particular memory corruption vulnerability that existed in the CoolType.dll library, which is used by the Reader and Acrobat, was actively being exploited in the wild. And all that was required for exploitation was that a user opened, just opened and viewed a PDF. The action of viewing a PDF would allow a remote attacker to execute arbitrary malicious code on the target machine, both in Mac and Windows. And interestingly, we do have some Mac news. Mac has been having some problem in the last week.

Leo: Yeah. A little worrisome. We've been problem-free for a while.

Steve: Have been, not a target. And just in time for the podcast, Apple has updated iOS. It is, depending upon which type of phone you have, it's 4.2.8 or 4.3.3. This is the fix to what they call their "crowd-sourced location database cache." In the window that you get, it says "This update contains changes to the iOS crowd-sourced location database cache, including reduces the size of the cache." Remember they were keeping it apparently forever, and now they said they were going to - in what we read before, last week they were saying they were going to bring it down to just seven days because that seemed to be reasonable. And they were saying, oh, yeah, it was a bug that it was keeping it forever. It will no longer back up the cache to iTunes, which is actually news. And it will delete the cache entirely when Location Services is turned off. So those fixes that we've been awaiting have now just this May 4th been released from Apple, in the morning before we're recording this.

Leo: Actually it's been pretty quick. That's less than two weeks since it was discovered. So that's - it's a pretty good response, I think.

Steve: Yeah. I think so, too. In security news, this doesn't directly bear on cybersecurity, but I did find it interesting to note that, in the successful - what word do I use? In successfully apprehending and killing Osama bin Laden, which the U.S. Special Forces Navy Seals did a few days ago, they spent a substantial length of time going through the compound and then acquiring 10 hard drives...

Leo: Oh.

Steve: ...five PCs, more than 100 storage devices including CDs, DVDs and thumb drives.

Leo: So that's what those couriers were bringing, were taking.

Steve: Yes.

Leo: Now, they had no phone service. They had no Internet access.

Steve: Well, and I think that was really smart. From a security standpoint, that represented some cleverness. And they were also burning all of their refuse, all of their garbage, instead of having any of it taken out of that compound.

Leo: Although, as we know, that was also a red flag for security services, that there was no connectivity in this mansion, and that periodically they were burning stuff. That was also a little, you know, suspicious.

Steve: There was a lot that was screwy because none of the other homes that were there had 18-foot walls.

Leo: Yeah, yeah.

Steve: And the third-floor balcony had its own seven-foot privacy shield, another wall, so that people could be out there on the third-floor balcony and not be seen. So, I mean, it would have been clear to anybody that whoever was inside this thing was intending to maintain their privacy and their identity to be kept a secret and regarded as a high value. Of course, no one, well, certainly we didn't, and lots of people did not know who was there. So it was a win for intelligence. And, but, yes, they certainly were taking measures to be secure, and having no contact with the Internet and just using couriers in and out. And of course it was, we were told, it was tracking couriers was the way they ended up deciding that Osama must be located in there.

And relative to that, I did tweet a reminder to everyone that hot topics which are flashing across the Internet through Twitter and through email still are getting people to click on links that they might not otherwise. That is, even trained people, when email comes in and it says "Post-mortem photo of bin Laden," or a photo of one of his wives supposedly used as a human shield or something, that in that moment of, ooh, wow, that's really newsy, or I really want to see that, that can cause people to forget their training and click. And that has been happening ever since Sunday when this news broke is there have been a, I mean, immediately the bad guys jumped on this and used this to exploit just our interest in more information.

Leo: Happens every time.

Steve: Yeah, it does. So I just wanted to take the opportunity to remind our listeners, once again, just even if you, I mean, and that's the problem. The more you want what the come-on is offering, the more likely you are to say, oh, well, maybe it's true. Maybe this is a photo. So just...

Leo: And there are no photos. In fact, I see now the news story that the President has decided not to release any photos. So if you see that somebody's offering photos, unh-unh, they don't exist.

Steve: Yeah. And the argument has been going back and forth, there have been some - the news agencies, of course, would like it. But the point has been that there hasn't been a huge level of push and skepticism from outside the United States for confirmation. Everyone is believing what we have. And frankly, Leo, you know that probably high, high up, other heads of state have had access to, if nothing else, a diplomat would have opened the folder and said, see, here it is, and then closed it and put it back in his diplomatic satchel. So you can imagine that people who absolutely have to know have had whatever level of convincing data that they've needed. That's certainly going on behind the scenes. But that's the kind of thing you just don't want loose on the Internet or it will never be pulled back. So, yeah.

Had an interesting and disturbing story that Ars Technica reported that I wanted to share with our listeners in its entirety because Ars did a very good job. And it's another reminder. The story was "PC rental store accused of using webcams and keyloggers on customers." This was just in Ars Technica yesterday. They wrote, "Built-in webcams are becoming more and more common in computers these days; and, in turn, they are becoming more and more of a liability. A Wyoming couple is now accusing national rent-to-own chain Aaron's, Inc. of spying on them at home using their rented computer's webcam without their knowledge."

Leo: Oh, my god.

Steve: Uh-huh. "Aaron's also allegedly used a keylogger and took regular screenshots of the couple's activities on the machine, leading the couple to file a class-action lawsuit in the U.S. District Court for the Western District of Pennsylvania. According to the complaint filed on Tuesday, Aaron's has been using a product called 'PC Rental Agent' on its rent-to-own machines since at least 2007 in order to 'surreptitiously access, monitor, intercept, and/or transmit electronic communications' made by Aaron's customers. Created by a company called DesignerWare, PC Rental Agent is advertised as a way to keep track of rent-to-own computers and lock out customers who fail to pay." Okay, well, obviously it does way more than that. I mean, this is full-on spyware. "According to the lawsuit, the product was sold to Aaron's under the guise that it was undetectable by users, and Aaron's apparently conceals the fact that it has the ability to monitor customers' activity when marketing its services.

"Crystal and Brian Byrd found this out the hard way in 2010 when they rented a Dell Inspiron laptop from Aaron's, which they paid off in full in October of 2010 - one month ahead of schedule. Aaron's didn't record the last payment correctly, however, leading an

Aaron's store manager to show up at the Byrd home in December in order to repossess the computer. The store manager then produced a photo of Brian Byrd using the machine, taken with the Inspiron's webcam, as apparent 'proof' that the Byrds were still using the computer.

"The Byrds ended up calling the police, and an investigation later concluded that Aaron's 'routinely installed the PC Rental Agent' on all of Aaron's rent-to-own computers. Law enforcement confirmed that the product indeed permitted the company to routinely take webcam photos, screenshots, and log the keystrokes of its customers without their knowledge or consent.

"It's unclear how many other photos Aaron's might have collected on the family, but Brian Byrd told the Associated Press that he was concerned about the content of photos that were potentially taken of his wife and child." He said, "'Chrystal gets online before she gets a shower and checks her grades. Who knows? They could print that stuff off there and take it home with them,' Byrd told the Associated Press. 'I've got a five-year-old boy who runs around all day and sometimes he gets out of the tub running around for 20, 30 seconds while we're on the computer. What if they took a picture of that? I wouldn't want that kind of garbage floating around out there,'" he said.

"The Byrds' situation is eerily reminiscent of one that occurred last year in" - get ready for it - the Lower Merion School District in Pennsylvania."

Leo: Uh-huh. That's right.

Steve: Which we talked about extensively. "At that time, some parents discovered that the school district had used remote software to activate the built-in webcams on the students' school-issued computers in order to check up on them at home, while the district insisted that its 'spying' policy only applied to laptops that were reported stolen." Which of course we know was not the case. "The district ended up settling two lawsuits for a total of $610,000, despite apparent e-mail evidence that the IT staff responsible for monitoring the laptops regularly viewed the students' photos for entertainment.

"According to Brian Byrd, the computer in question is still in police custody as evidence, and no one from Aaron's has yet commented publicly on the case. However, the Byrds are hoping to get the suit certified as a class action so that other customers who might have been affected can get in on the lawsuit. After all, Aaron's claims to have more than 1,500 stores in the U.S. and Canada alone, and there are bound to be others who are only now discovering that they don't have as much privacy at home as they thought."

Leo: As they thought?

Steve: Yeah.

Leo: As they should have.

Steve: As they - yeah.

Leo: As they deserve.

Steve: As they deserve. So and that's another reminder. I'm hoping that we will see physical shutters added to all webcams in the future. But we're not seeing them still. I mean, for example, all of the iPhones and iPads are shutterless. And I smile every time I go to Starbucks and I see a friend of mine, who I told about these things, who just has a post-it note, a little sticky chunk of a post-it note stuck over the webcam on his laptop because he doesn't want to worry about his laptop staring at him and looking at him and getting infected with something that would do that. So it's simple, physical security.

I have a very high-power laser, which is dangerously powerful, and some of the criteria for having that is that you need a key switch and a time delay from the time you turn it on and it actually energizes, and a physical shutter. So we have - there's a law in place that requires lasers of a certain power to be physically shutterable so that you have multiple fail safes. And it really is the case that our laptops should be the same way. They ought to just have something where you can slide it, and this thing is blanked out. It's a trivial thing to add. And we're still not seeing it.

Leo: We predicted this after the Lower Merion thing, that this would be in every laptop going forward, and it is not. And I'm kind of surprised, to be honest.

Steve: Yeah. It really does, I mean, we just need some pressure on the manufacturers to make that happen.

Leo: It's an easy thing to do.

Steve: And in weird news, but important, I thought, or just I wanted to mention it because we talk about Kaspersky so often, Eugene Kaspersky's son Vanya was kidnapped...

Leo: Saw that.

Steve: ...and safely returned.

Leo: Didn't see that. That's good.

Steve: And the bad guys were caught.

Leo: Yes.

Steve: On 4/22, on April 22nd, Kaspersky Lab posted a statement: "Kaspersky Lab respectfully asks members of the media to refrain from speculating and distributing unconfirmed information about alleged events relating to the Kaspersky family. Eugene

Kaspersky continues his day-to-day work at the company and has stated that the unconfirmed information being spread at the moment is harmful for the company."

Then two days later they posted the statement: "Kaspersky Lab confirms that an operation to free Ivan Kaspersky" - I ought to mention that Ivan is his pseudonym, his son's pseudonym that he uses for posting and doing things online and in the company, where his real name is Vanya. So it says: "...operation to free Ivan Kaspersky was carried out successfully by the Federal Security Services (FSB), the Criminal Investigation Department of the Moscow Police, and Kaspersky Lab's own security personnel. Ivan is alive and well and is currently at a safe location. No ransom was paid during the operation to free him. Eugene Kaspersky and Natalya Kaspersky express their profound gratitude to those who participated in Ivan's release and to all those who supported them throughout the last few days. Both are currently unavailable for comment."

And then he did also, four days later, on the 28th, Eugene has a Facebook page where he posted, among other things, that "Vanya is back home safe and sound. Thanks for your support." So that all came out well for them, which is wonderful.

Leo: What a relief, yeah.

Steve: Yeah. We mentioned last week that the MySQL site itself, that is, the people who maintain and develop and move the MySQL server project forward, had been victims of a blind SQL injection attack. I ran across an extremely good treatment, a step-by-step coverage of exactly how this was done, specifically on their site, at a company called Acunetix.com. If you go to Acunetix.com, up at the top there's a link to their blog. And if you scroll down to their blog posting, "MySQL.com Victim of SQL Injection Attack," what you'll find is a very nice, detailed treatment, step by step showing how this was done. We've talked about it in the past, so I didn't want to go through it again. But I know that we've got a broad audience, and some might be really interested in the details. So you can find them there at Acunetix.com on their blog.

A number of people, in the wake of the recent concern about Dropbox's lack of security, talked to me - or actually tweeted, that's how I learned about it, through tweet or through Twitter tweets.

Leo: Tweeter. Tweeter Twitter.

Steve: Yeah.

Leo: Just don't call it TWiT.

Steve: Something that's in beta called "SecretSync." The site is GetSecretSync.com. I have not done a full analysis and appraisal. I did spend some time looking at the site, literally every page that they've got that's available. It's still rather sparse. It is in beta at this point. And I think I remember seeing that you would have to have Java installed. So I think it's Windows only, but it's going to be multiplatform. Thus they're using Java to get the multiplatform support, which makes their job easier.

And everything I saw leads me to believe that they're doing everything right. That is, they're doing client-side encryption and decryption so that everything that is being stored in Dropbox - this is sort of a front end for Dropbox. So you still use, you get to keep your existing Dropbox account, your existing usage of Dropbox. But then installing this SecretSync service creates an encrypted folder in addition to your regular Dropbox folder. And what it means is, essentially, that it is encrypting what you drop there before it leaves your machine. So it's exactly the kind of security you want. They're saying that all of the responsibility for not losing your key that you create as part of setting this up is on you. That is, they don't have the key. They're storing nothing but pseudorandom noise which you send them, which comes out of the encryption process. All of that is exactly the right-sounding thing.

So again, I have not personally vetted this. I haven't installed it because I'm not a Dropbox user. So I can't vouch for it. But I wanted to tell people who said, "Hey, Steve, look at SecretSync, what do you think," what I think is, from everything I've seen, this looks like they're done the right thing. I don't know what plans they'll have in the future when it comes out of beta, whether they're going to charge people or what their deal is. But from everything I saw, it looks really good.

Leo: Does solve that issue, that's for sure.

Steve: Yes, yes. Were I a Dropbox user, I would use this in a heartbeat.

Leo: Well, we are, as you know. We use it heavily.

Steve: Yeah. Although you're using it more for...

Leo: Yeah, I don't care if people find your file.

Steve: ...public, exactly, for public stuff, exactly. And we have seen the first fake AV software targeting Mac users.

Leo: Terrible.

Steve: It's called - there is a legitimate product called MacDefender. And so this one is a - this is malware masquerading as MAC Defender, which claims - it's the bogus AV stuff. It claims that Mac OS X has been infected and asks users who get themselves infected with this for anywhere between $80 and $99 to purchase antivirus software to fix the problem. It is apparently spreading through Google Images somehow. I don't know if it's - there's been no claim that the Google Images site itself is the problem. It may be infected images. I saw something, someone said that he was looking for images of piranha, and I guess he got bitten.

So the one tip that I have for our listeners is that some Mac users in the past will have allowed Safari to open files which it deems safe after downloading them. And apparently it is people with this enabled which are getting caught. So under Safari, go to Preferences > General, and then uncheck the option called "Open 'safe' files after downloading." Turn that off so that Safari doesn't do this automatically behind your back.

Leo: Well, no matter what you have checked, you still will be asked for a password.

Steve: Really.

Leo: It does not - yeah. That's my understanding.

Steve: To install software. So people must be doing that.

Leo: Yeah, oh, yeah, but that's the problem. At least this was my understanding. We discussed this yesterday on MacBreak Weekly. But that's the problem, I mean, it's the same problem with UAC on Windows, which is you get kind of fatigued, and you go yeah, yeah, yeah, yeah, yeah.

Steve: Yup, exactly.

Leo: Apple at least, unlike UAC, even if you're running as an administrator, you still must enter the password.

Steve: Right. What was the - there was a joke Saturday night at the White House Correspondents Dinner. Shoot. I remember the punch line now, but I can't remember what the joke was.

Leo: Was it Seth Meyers or...

Steve: I don't remember whether it was Seth or Barack. I think it was Seth. And the punch line was that - oh, shoot. It was that I'm sure that most of us treat this the way we do updating terms and conditions on iTunes.

Leo: Oh, yeah. There was a little slam against that, yeah. Yeah, yeah, I think that was Seth Meyers, yeah. That was pretty funny.

Steve: Yeah, it was really great.

Leo: Which shows you how...

Steve: And everyone knew what he meant. Like, yeah, yeah, yeah, we all know. Updated terms and conditions, yeah, click, you know. And so on we move.

Leo: So to be clear, it will auto download if you have that turned on. The issue is it will not, even if you run it manually or automatically, you will have to give a password. This is true in any case, even if you're running administrator. You will have to give a password before it installs. So the problem is it looks pretty credible. I mean, it's a pretty credible-looking, you know, no obvious misspellings, no grammatical errors. But we just have to drill into people's heads you don't just randomly - the problem is, I guess, if you click saying yes, I want this thing, of course you're going to give it permission to install. You've already been fooled.

Steve: Yeah.

Leo: It's terrible. Don't be fooled by pop-ups.

Steve: Exactly. So we have a lot more on our Attacks & Breaches section, mostly about Sony. The news, the real news we have is that apparently the breach went further than just the PlayStation Network, such that on Sunday night the Sony Online Entertainment System, which is their online PC games network, was also shut down, and they sent email to another set of users. And I remember seeing the number 12.5 million. So there were, I mean, there are still problems happening.

Brian Krebs, our security follower and watcher, captured some screenshots which I saw, which then other people - which sort of annoyed him - other people were taking his screenshots and redisplaying them as their own without giving him any credit, which were of some forums where the people posting appeared to be saying that there were 2.2 million credit cards up for sale. And also someone was claiming that Sony had been given the opportunity to buy the database back, but had not responded. And there have been continuing reports surfacing of credit card details being sold on "carder forums," as they're called. And at least in one case there was a report that a Sony administrative password had been compromised. Which may have given people access to more.

So, I mean, it really is a mess. Now, Sony has said in an interview that was conducted in Japanese and translated, that the passwords were hashed, that is, they were not encrypted. There was some miscommunication initially. They're saying that they were not encrypted, but they were hashed, which is essentially the same thing. We hope that it was a salted hash.

Leo: That's the issue, isn't it. We've talked about that before.

Steve: Yeah. Otherwise rainbow tables could be used in order to reverse the hashes. And then I'm wondering, I mean, I guess somebody really messed this up from a communications standpoint because they really scared people by telling people that, if you use the same username and password anywhere else, that you had to go change it. But if they had properly cryptographically hashed the passwords, and if that's the only thing that the bad guys got, then that is an irreversible process. If you have a salted hash, and the salt isn't known - and hopefully the salt would have been stored separate from the database, I mean, you wouldn't expect it to be in the algorithm and not in the customer records - then it's as good, I mean, that is encrypted. It is, I mean, it's even better than encrypted because in typical encryption you can reverse, and a salted hash you cannot reverse. It is inherently a one-way process. So they really scared people, probably unnecessarily, if in fact they had hashed it well. We just really - we really don't know.

Leo: And if you're puzzled about salt, I think this show on randomness will help understand a little bit of that; right?

Steve: Yes, absolutely. And I got a tweet from @LeonZandman, who sent me an interesting link to a new posting at CareerBuilder. Now, on April 20th, on 4/20, the PlayStation Network outage began. On 4/21, Sony posted on CareerBuilder for looking to hire a senior application security analyst in San Diego.

Leo: Well, it's about time.

Steve: Which is where they're located. Now, the good news is it's a full-time position.

Leo: Yes.

Steve: And under required travel it says "none," which is good. We want that person to stay put, stay in San Diego, and do their full-time job of...

Leo: No traveling allowed, yeah.

Steve: ...making us more secure. And as I predicted last week, lawsuits have been filed against Sony for this. I mean, I feel badly for them. And again, it's unfortunate this happened. And I guess there'll be some court battles to decide what culpability was on Sony's side of this.

There was an Xbox Live podcast where they took the opportunity to mention that the Xbox Live people take security very seriously, was what they said. And so we're all glad to hear that.

Leo: Thank you.

Steve: Yes.

Leo: Just in case you were puzzled, or worse.

Steve: In case you people who are on Xbox and not on Sony were concerned. And then, in an example of something done really well, that is, of the kind of confession that you really want to see, although you never want to see the problem, DSLR, DSL Reports, was also breached, through I believe it was a botnet. In fact, I'm sure it was, or they believe it was. They said that 8 percent of their historical users - now, this goes way back in time, too - 8 percent of their users' usernames and passwords were stolen. They made a very responsible posting. They said what happened, and I regarded this as the proper way to report and take responsibility for the breach.

They said, in brief: "An SQL injection attack by a botnet" - oh, so it was both. It was SQL injection. That's what I thought I remembered. But it was conducted by "...a botnet on Wednesday afternoon," so that would have been one week ago today, "obtained a large number of email and password pairs. The ones they obtained were basically random, so they covered the entire 10-year history of the membership, but were sprinkled randomly throughout that. Some were very old accounts; some are new accounts; some inactive and deleted."

And so I'm reading from this posting, where the poster said, "I identified the newest accounts, those that were obtained and have logged in over the last 12 months, and have alerted those by email. This amounts to some 9,000 accounts. If your email/password was revealed, you received the alert email or discovered your login password has been changed by us already. You also need to think of what other sites you use allowing logins using the same registered email address and password." So in this case at DSLR they were not hashing the passwords, and the email and associated login did get loose.

Leo: [Muttering].

Steve: "Some sites" - I know. "Some sites, especially email services like Gmail and PayPal and Facebook, allow login by email and password. If you are in the habit of sharing the same password among many sites, then the people who obtained this data from us can log in as you. So you should secure your access to those sites by changing your password there immediately. Your first priority would be your primary email account, if the password was shared with it.

"It is unclear how much data the logged intrusion requested which actually reached them. The site was quite unresponsive during the attack. And whether that data is being used yet, we don't know. I'm going on a worst-case scenario here. It is also unclear whether the emails obtained will be spammed or just searched for high-value targets such as PayPal, Gmail, Google Docs, et cetera. Older inactive accounts involved are also being notified by email now, although the older the account, the less likely the email is still current or the password they use is still useful.

"Obviously having both an SQL injection attack hole, now closed, and also storing plaintext passwords, as we were, is a big black eye for us. And I'll be addressing these problems as fast and carefully as I can. My apology for any stress this causes. If you are like me, you've also got the PSN network issue hanging over your head, as well. Judging from the replies to the initial email, the impact is varied. Some people used a unique email or unique password for the site, and others used the same password everywhere and will have to be much more careful."

So that was what was posted in DSLR's blog. And I have to salute them and this person for being as, I mean, really taking responsibility and being as forthright as he was. I mean, certainly no one wants to ever have to generate a blog posting like that. They also had a short Q&A.

They said in the Q&A: "A large network (botnet) of compromised windows machines circumvented individual IP access limits on unusual activity. The attack was blocked before it had completed more than 8 percent of its work." And then under a question, they asked themselves, "What is the likely use for the data gained?" The answer is:

"Gaining access to email accounts, gaining access to accounts at PayPal/Google/Amazon/Facebook/Twitter or other big sites where login is via email and password."

Question: "What kind of shoddy operation are you running here?" And again, I salute them for, I mean, they posted this question. Answer: "Not making excuses, but it is sobering to read that just recently MySQL.com was hacked with an identical approach to the one used here (blind SQL injection). More MySQL-based sites will suffer the same issue this year, so users should take care to reduce their password re-use on multiple sites to at most high/medium/low value passwords. A common low value password for forums, unique ones for banking, et cetera."

And that's actually a good strategy. If you don't want to go through the trouble of maintaining a truly unique password for every site, or, for example, if you're not using LastPass to automatically do that for you, then you are able, you could absolutely regard some passwords as low value and allow yourself to reuse them, for example, for posting in forums, but absolutely use known unique ones for banking purposes. So anyway, it's unfortunate that DSLR got hacked. But they couldn't have handled it any more responsibly than they did, so I take my hat off to them.

I tweeted earlier this week that I had finally had a chance to sit down and spend some time with and start using Certificate Patrol, which is an add-on I mentioned blindly last week that I knew of, I had found out about, also through Twitter, and hadn't had a chance to look at it. I called it in my tweet "a terrific Firefox add-on for the security-aware power user, five-star Security Now rating, a total win." And I continue to feel that way. So I wanted to follow up on last week's mention of it, for those people who are not following me on Twitter, and say I'm really happy with this thing.

It's popping up, as it's designed to, whenever I go to a site that I have not - a secure HTTPS SSL site that I have not yet visited since I had installed it. And so when it's caching those sites, that's when I am being shown information about the certificate being used, how long it's been there, how much longer it has till it expires, and who the issuer of the certificate is. So so far I haven't run across an instance where one that it already has has expired. But it will notify me of that explicitly. Nor certainly have I run across one where I'm going to a site where everything seems changed about the certificate, which would be a real red flag indicating that potentially you're being spoofed somehow.

So again, I call it an "add-on for the security-aware power user" because initially, as you're going to SSL-enabled sites, you're going to be seeing these pop-ups a lot. It pops up, and it's a modal dialogue, meaning in Windows terms that you have to say okay and close it before you're able to proceed. Which is actually what you want in the case of something coming up and being bad. I'm also, it's interesting, I'm using it to learn about the sites that - the certificate authorities that other of the major popular sites that I go to are using. And this is not something I ever really, like, took the trouble to research before. Now it's just being given to me.

And one of the things that I'm seeing is that major people are using DigiCert. Facebook and Sony - I went to Sony's site, naturally, in the last week, doing some research for this podcast - they're both using the DigiCert high-assurance certificate authority. Thawte is being used by Google, and Equifax is being used by Level 3, some various sites that I happen to have been visiting in over SSL recently. But DigiCert's pricing is very good. So there's a chance that VeriSign is finally going to lose me after all these years because their pricing is the worst, that is to say the highest there is in the industry.

Leo: Thousands; right?

Steve: Yeah, it's insanely expensive. And for example, I don't have a fancy, make-your-URL-turn-green cert, an extended validation, an EV cert, because they're too expensive from VeriSign. And it's not like I only have to pay that once. I've got to pay it every time. So as I'm seeing people like Facebook, I mean, any CA that Facebook is using is obviously in every browser on the planet. So that's all I really need is a certificate authority that's going to be issuing certificates that I'm sure that everyone's browser will be able to receive. And I'm seeing now that a lot of high-profile sites are using DigiCert. So that's good enough for me.

So VeriSign's got to get with the program here. Maybe they'll just stay ultra high-end, you know, Level 3 uses them. But GRC, unless they fix their pricing, I think they're going to lose me. So it's been an interesting education. And the Certificate Patrol is a handy little add-on for Firefox.

Leo: Cool, yeah.

Steve: I received a tweet, or I saw a tweet, from Gabe Ormsby, who tweeted something that I pointed out, which was that password length limits are a strong hint that they're not hashed passwords on the server end. And when I saw that go by, I thought, yeah, that is a good point, and I wanted just to remind our users of it. We're all now becoming aware of the problem of people we rely on to protect our data - for example, Sony and others where we're trusting them. Well, if you are presented with a length limit, and I've seen them, for example, passwords must be between eight and 16 characters long, seems typical, I mean, that's a very good clue.

Leo: A minimum's okay. It's the maximum that's scary.

Steve: Right, because what that implies is that in their SQL database, which the hackers are currently trying to get into, they've got a 16-character field into which you can put a password of up to 16 characters. What we want is them to have a 256-bit field, which is, what, 16 characters, yeah, 16 bytes. And we want them to hash however long a password you give them and not worry about it, so that is to say where they're not imposing a length limit on the user. When they do, it implies they're storing the password itself. So that's just a little tipoff. When you see that, you might write a note to their support people or their security people or their privacy people, whatever, and say, hey, how are you storing this password of mine? Are you hashing it or not?

Leo: And salted hash is what you want; right?

Steve: Yes. Yes.

Leo: Even a hash by itself is not as good as a salted hash.

Steve: True, because a hash is going to be using a known algorithm. And it's, for example, if the bad guys created an account on that service and then stole their own hash, they would know both the plaintext password and the hash. And they could then run that through a bunch of all of the various hashing algorithms and quickly determine which one was being used by that site. And then that would allow them, if it weren't salted, that would allow them to apply the - to know what the hash was, and then grab the rainbow table that's appropriate and see about reversing the hashes. So, still, hashing is way better than not. Salting is what everyone who's hashing certainly should be doing. But storing in plaintext is just a bad idea. And if they're giving you a limit on the length you're able to store, that really does raise a red flag. It's like, okay.

Leo: And that's because hashing kind of changes the length anyway.

Steve: Well, actually, hashing, no matter what you - you could put in a one-character password.

Leo: And you'd get the same length.

Steve: And it would give you a 256-bit hash. The hash...

Leo: No matter what.

Steve: Yes, the output is always the same fixed length. It's 128 for a 128-bit hash, 256 bits for a 256-bit hash, and so forth. And then lastly, Jared in Redmond, whose Twitter handle is @jshoq, sent me a note just mentioning that Bank of America offers one-time-use credit cards, and even, as you mentioned, Leo, last week, recurring payments to a single vendor.

Leo: That's what we want.

Steve: So I wish Chase did that. But no such luck so far.

Leo: It may also depend on the card you've got, I guess.

Steve: Yeah. Under miscellaneous gizmos, something is going on with Bitcoin. I don't know what it is. But I mentioned that, last week I think it was, or no further back than the week before, I think it was last week, that they were at a dollar something, a little over, like, $1.10 or $1.20 or something. It's now at $3.50. So a single bitcoin is now trading for $3.50 in U.S. dollars. I checked this morning to see what it was because I got a tweet either from Bitcoin or from someone notifying me, I don't remember which. But it's like, whoa. So those bitcoins are becoming valuable.

Leo: They add up.

Steve: I'm glad I got 50 of them. And Leo, I just meant to ask you, I've seen commercials for the BlackBerry PlayBook, and I know that on the show we mentioned you had just received one, I guess maybe toward the end of a podcast a couple weeks ago you were just unboxing it? What do you think?

Leo: Well, I'm playing right now, I'm playing Need for Speed Undercover, which was designed, it says when you launch it, for the PlayBook. And you can see the power of the Tegra 2. I mean, this isn't - hardware-wise, this is nice. Now, watch. Remember, we're running QNX. I can slide up, and the multitasking continues. And you see I have other things going, including, oh, there's Steve. Playing back, by the way, in Flash. Now, it's fake multitasking, as you can see, because it's paused the Flash. But there we go, there's the live thing. And it plays for a little bit, but the game is stopped over there. But it kind of feels like real multitasking.

We also have the - Mediafly has made the TWiT application available, so that's kind of nice. You can watch them after the fact. But having Flash in a browser is cool. It's a nice browser. It's very much like the WebKit-based browsers on the iOS devices. In fact, I'm sure it's WebKit-based. It seems to work very well. You can zoom in and zoom out and all of that stuff. I do like the multitasking.

I think really the only thing wrong with it is the dearth of software. It's not - it's a nice form factor. It's a little thick, you know, compared to the iPad 2. But it has a lot more in terms of capabilities. It uses USB, micro-USB charging. It's got a camera front and back that's I think better than the iPad's camera. I like it a lot. I mean, I think it's a nice device hardware-wise. We just have to wait until - and, by the way, the OS really is well done. We just have to wait until there's more applications. Remember they're going to make it possible to run Android apps in emulation.

Steve: Right.

Leo: We haven't seen that yet, so we don't know how well it does with Android apps.

Steve: Ah, so it didn't have that at all out of the box.

Leo: No, it's not out yet. And then they also have a developer kit that makes it supposedly easy for developers to port to Android, from Android to the PlayBook. So that could help the problem. I mean, it's another one of those chicken-and-egg things. You've got a new operating system, a new platform, and you maybe haven't sold a whole lot of them yet. You've got to convince developers that it's worth the investment to write software for it. And you've got to convince people to buy the hardware, even if there's no software out there for it. I mean, this doesn't even come with an email app or a calendar app.

Steve: [Laughing] And you can't get one yet?

Leo: No.

Steve: Oh, my goodness.

Leo: Now, I'm sure there'll be third-party versions. If you have a BlackBerry now, you can use their bridge technology to put email on it. And if you look, it says, oh, email. But what it really is is it's using the browser. So if you have browser-based email you can do it.

Steve: Right.

Leo: But, I mean, it's not what I would call true email. So they've got a little ways to go. It's, you know, it's nice hardware, and the OS is very snappy, feels really good. I think they've done a nice job that way. But still and all I wouldn't say get it instead of an iPad 2. It's the same price. Oh, and no 3G option. It's WiFi only. So it's just too limited. And I think the problem is - Paul Thurrott talks about Windows Phone 7 in the same context. It'd be one thing if there weren't an iPad.

Steve: Right.

Leo: But there is.

Steve: In the shadow of the iPad, good luck.

Leo: Yeah. It's just very tough to compete against an existing platform with hundreds of thousands of applications available. By the way, while you've been talking, I've been downloading the iPhone 4 update. I don't know if this is intentional. It's a little weird. 666 megabytes, 666. It's a satanic download.

Steve: Wow. Wow.

Leo: Well, I think that - and they say in the notes that all they're fixing is this location database. But I don't think they have the capability of doing vector-based OS updates. I think you have to download the entire firmware each time.

Steve: They just can't do a delta.

Leo: No delta. Not, yeah, not vector, delta.

Steve: Wow. Wow.

Leo: Let's take a break. Did you want to do a SpinRite or anything, or...

Steve: Yeah, I've got a few more little gizmos I want to talk about.

Leo: Oh, there's more, I see. Good Morning America. Did that happen?

Steve: No. And I did want to follow up because a number of people, I get questions pretty much every day, whatever happened with Good Morning America. And I have no idea. My guess is that they may use the footage, it may just be in storage until a high-profile breach of some sort actually happens, and then they'll sort of dust me off because that's what they originally had me come up to talk about. They may have decided that the Firesheep issue was just too scary, or a little too geeky and not mainstream enough for their mothers doing the ironing in the morning crowd, who knows. But as far as I know, I mean, it's been so long now that I'm guessing it's just not going to happen.

I've also had a lot of people say, Steve, would really love to have the TechTalk columns from the old days of InfoWorld, which came up, I guess it must have been when you and John and Jerry and I were doing TWiT on Sunday a few weeks ago, Leo, that we were talking about the InfoWorld column. I wanted to mention that...

Leo: Are you going to do what he suggested?

Steve: Well, I've been struggling towards it. I just found the DAT tapes which...

Leo: Oh, my god. They're on DATs.

Steve: Yes. All of the backups, I mean, I see one here that is labeled "C: and D: August 1, '96." So I remembered at the time that the machine I was using, I think I was under Windows 3.1. I know that I published the books, the Passion for Technology books, in Ventura, which was the premier sort of geeky publishing tool at the time. I think Adobe had theirs, but I was over on Ventura. And it started off on the GEM platform actually.

Leo: Yeah, yeah. I was going to say, because if you just took all the text of the columns, it'd probably fit on a floppy.

Steve: Yeah. Although I did, I did a diagram. I went back, the column typically did not have diagrams. And so, I mean, I really put a lot of effort into republishing the columns.

Leo: That's great.

Steve: And when people were asking for them, it's like, uh, yeah. They're around here somewhere. And literally, I knew that they were on a hard drive that I probably have. But I also know that I was backing them up on DAT tape constantly. That was my backup medium.

Leo: Now, here's the big question. Do you have the DAT player?

Steve: I do. I've got two drives and the software. So I kept - in fact, what I don't have is a parallel port because it was a parallel port interface. So I purchased a PCIE, I think it is, to parallel port adapter card, so that I could stick a parallel port in one of my current machines in order - and then I'm going to have to set up a DOS partition because this thing ran under DOS.

Leo: Oh, my goodness.

Steve: It looked a little bit like XTree, as I remember the UI, so...

Leo: And to be honest, it's '96. This is 15 years ago.

Steve: Yeah.

Leo: This is something we've got to keep in mind, that these data formats have the shelf life of a fruit fly. And you've really got to either keep up or say goodbye. Because, I mean, how many people really, I mean, nobody has ZIP drives, DAT drives, the software. This stuff is - it gets antiquated so fast.

Steve: It's where me being a packrat comes in handy. I actually had all this stuff.

Leo: So that's good. So we can at some point expect an eBook with the columns of Steve Gibson.

Steve: And it'll be free. I would never charge...

Leo: Oh, Steve, why not? A buck fifty.

Steve: No. It's just - I'd just rather make it available. And I'd like it to be in PDF and...

Leo: Unprotected open standards.

Steve: Absolutely. It's just for people to, I mean, as you say, Leo, it's 15 years old. It's like...

Leo: Oh, yeah. It's great stuff. I bet you it's more - have you read it yet? I bet you...

Steve: There's a lot of good stuff there.

Leo: ...more of it's germane than you might think.

Steve: It's surprising, actually. As you look through this, it's like, wow, nothing's changed.

Leo: Yeah. Yeah.

Steve: Okay, now, I have, I think, a reading assignment for our listeners.

Leo: Ooh, I love that. I fire up my Kindle. You know I bought that new Kindle, the ad-supported Kindle, just to see what the ads are. It's not bad. Instead of those woodcuts, they've got the ads.

Steve: Because they are adding, I mean, they're advertising it on TV now at $114, and I always kind of grumble when I see that, thinking, well, is that really false advertising? Because it is ad supported. But it's not in your face, huh?

Leo: No. Once you're reading, it's gone. It's just instead of the beautiful woodcuts, which I do miss, there's the screensaver. It puts an ad there. Those are rotated quite a bit. They're different all the time. And then when you're on the menu page there's a little bar at the bottom that's an ad. That's all. It's I think very unobtrusive. I wish you saved more. You only save $25. But still, 114 bucks for a Kindle, not bad. And that's my, I think, my fourth or fifth Kindle now.

Steve: Okay. So our friend Mark Russinovich, whom we have spoken of many times, he's famously known as one of the two main, well, one of the two guys behind the Sysinternals website. And Sysinternals tools have been used by all of us security people, I mean, they've got, like, great process explorer and all kinds of low-level tools, I mean, I've used them for years. And of course we know, with a bit of a tear, Microsoft bought Sysinternals, which I'm sure was good for Mark, and I'm glad for that.

About a month ago he sent me a tweet saying, hey, Steve, I'd like to send you a copy of my new book. Where can I send it? And I got my address back to him, and Federal Express arrived the next day. Now, the good news is, it is available in e-format for the Kindle. So I bought it because I wanted to read it on the Kindle, but I really am tickled to have a physical copy of it from him. The book is fiction. And I had to stop myself from reading it last night because - and it'll be finished way before you hear from me again. It's called "Zero Day."

Leo: Yeah, we saw that he did that, that it's fiction, a novel.

Steve: It is good.

Leo: Is it good? See, that was my concern is like, well, Mark's a great programmer. He's a brilliant systems guy. But...

Steve: Yes. It is - the reason I call it a reading assignment is, I mean, it is factually accurate and chilling. He basically - and I'm at the 50 percent point. I'm exactly at 50 percent. It's when I finally said, okay, stop reading, Steve, you've got to go to sleep so you can do the podcast in the morning. And I can't wait to finish it. I have a sneaking suspicion of what's going to happen and where he's going to go with it. But it is really good. He does a very good job of painting the picture of the way we've become so dependent upon computers, and what would happen if this continues as we expect, and how it could happen that something really bad would happen.

Anyway, it's fiction. I wouldn't quite call it science fiction. I mean, it's really grounded in fact. And it's not a long read. It's just very pleasant. So I haven't finished it yet. As I said, I'm 50 percent of the way. But I wanted to give our listeners a heads-up that someone we've spoken of who really knows his way around the technology has written a book based on this technology. And, I mean, it's like a fictionalized version of this podcast because all the things that we've talked about, rootkits and viruses and propagation and nuclear reactors, I mean, it's weird that this thing came out when it did relative to things that have been going on. So I just, so far, I can really recommend it. I think it's a nice piece of work.

Leo: I can't wait to read it. It is on Amazon. As you said, it's available for Kindle. There is no audio book, unfortunately. "Zero Day."

Steve: It would be too soon, I would think, for an audio. But don't audios tend to lag a little bit because you've got to have time for someone to read them.

Leo: Depends on the publisher. Lately, if it's - and I think this is because it's not a well-known author. But lately a lot of stuff is coming out day and date with the print version. Very much more common.

Steve: Okay. So a short note from James, he called himself "Jay," Truesdale, who is a podcast listener. The subject is "SpinRite's temperature sensor." And he said, "Dear Steve, I finally have a SpinRite story to share, but not like any other I've heard so far. I went to look at a friend's computer that was really running slow. I pointed out the whine that was coming from the computer as a bad sign, and got to work on the usual Windows cleanup, including running the defragger. It was still slow afterwards, and I was out of time. So I took the computer home so I could continue working on it there.

"I let SpinRite loose on the SATA hard drive and came back later to check on its progress. I found a message from SpinRite stating that the drive was overheating. I shut down the computer and checked for ventilation problems and found none. I swapped out the friend's hard disk for one of my own hard disks, putting it into his machine, and ran SpinRite again. SpinRite ran just fine with my hard disk. So I knew there was not a ventilation problem, and that the original hard disk was suspect.

"I ordered a new hard disk, and when it arrived I ran SpinRite on the new hard disk. No problems were reported. I hooked both hard drives to the computer and used a small fan to keep the drives cool, and then used the open source tool Clonezilla to copy the old hard disk to the new hard disk. The new hard disk booted right up, and Windows XP seems to be running just fine now. I don't know why the original hard disk was running hot. But due to SpinRite's temperature warning, we were able to replace the hard disk before any data was lost. Thanks for the great podcast. James Truesdale."

Leo: You don't have a thermometer in SpinRite.

Steve: One of the things that SpinRite does is it continually polls the drive's SMART data. And we talked about SMART data being sometimes turned off. SpinRite checks and turns it on and notifies the user that it's doing so, so that it's able to keep an eye on the drive on the fly while it's running. And actually this is one of the coolest things SpinRite does is that SMART, the Self Monitoring And Reporting Technology, is it's useful, sort of, but where you really want it is to be monitoring the SMART data while the drive is under stress, that is, while it's actually doing work. That's when the parameters will demonstrate that the drive is having a problem.

But that's not normally the way SMART is used. It's the way SpinRite uses it. And as far as I know, SpinRite is unique in the industry in doing so. And in fact, when I was developing it, a lot of people in the forum were saying, oh, Steve, I don't know if you're able to poll SMART data on the fly while you're using the drive. And I said, well, we're going to find out. And it turns out we've never had a problem with that. Of course I wrote it carefully.

But one of the things it does is it polls the drive's temperature constantly. And if it exceeds the manufacturer's upper safe temperature limit, SpinRite will stop because it turns out the act of running SpinRite will generate more heat from the drive because it's seeking constantly, it's tick tick tick tick tick tick tick tick tick from one cylinder to the next. And each of that generates some mechanical energy which ends up actually increasing the drive's temperature. And we find often, or more often than on desktops, laptop drives will overheat when they're, like, not in a place where they're able to get enough ventilation because laptops notoriously have a problem being so small and needing to have small fans and just small air openings. They have a hard time moving enough air across their little drives to keep them cool. And oftentimes, unfortunately, ventilation is an afterthought. So that's just one more little goody that SpinRite brings.

Leo: 'Tis a goody. Coming up we're going to talk about how random numbers work on a deterministic thing like a computer. How would you make something random? I mean, truly random?

Steve: How can you get randomness when there's nothing random there?

Leo: Steve, let's talk random numbers. What do you say?

Steve: Well, so, okay. One of the reasons that I've always been a little bit annoyed by the people who talk about security through obscurity is no security is that, eh, it's not quite correct. I mean, and I've hedged that often because there are - the fact is there is always a secret of some kind in security. It may be the jaggedy, the exact pattern of jaggedy teeth on a key that you insert in the keyhole. Or your fingerprint that you want to keep secret, and you don't want to spread around. And we've talked about people being scanned at Disneyland.

But in all of the crypto that we've talked about, anytime we're establishing an SSL connection to a remote location, or we're encrypting a file just in place, or encrypting a drive, we're providing some sort of authentication and oftentimes a password. But separate from that there is a secret key. In some cases, in the case of asymmetric encryption, public key encryption, you've got two different keys, one able to encrypt and the other able to decrypt, that is, to reverse what the first one does. But again, at least one of them, in typical use, is kept secret.

And remember that even asymmetric encryption is so slow that it doesn't actually encrypt the payload. You don't actually, even if you're using public key encryption, you're not actually using the public key to encrypt the bulk data. Instead, you come up with a random key, and that's what you encrypt with the public, the asymmetric public key. And then you use that random key to actually do the data encryption, to do symmetric encryption, which is vastly faster. But in all of these scenarios where you're using SSL to connect to remote location, you're encrypting a file, you're encrypting a drive, you always have some sort of key. And that key is typically generated by some sort of random number generator.

Well, the problem, and this is a problem that goes way back in time, is that, as we know, computers are completely deterministic. That is, given a known starting state with known programs and known inputs, they will always follow a path of instructions and jumps and so forth, I mean, it may be, and typically is, incredibly complicated. But it's deterministic. The program is always going to do the same thing, given the same program, the same instructions, and the same computer that's executing them.

So the question is, okay, that's unfortunately not what we want in the case of getting something random. So let's step back a little bit and say, well, okay, what's the goal here? The goal is that, say that we were encrypting a communication. And we're not going to password protect it. We just want - Point A wants to talk to Point B, such as, for example, over an SSL connection. So we need to come up with a symmetric encryption key that cannot be guessed, that is unknown and practically unknowable by an attacker.

And so if we have - when we bring an attacker in, we need to define what the threat model is for sort of like what it is we're trying to achieve. And by that I mean, if the computer we're using were compromised, that is, if there was already malware in the machine which is able to see what the computer is doing, well, you can - anything you used to generate a key, no matter how good, for example, it was as a source of randomness, once this source of randomness had arrived at a final key, if the machine you're trying to use which you're assuming is secure is not, then it doesn't matter how good your source of randomness is. You've got something bad in your computer, malware which is able to grab your key.

So obviously that's not the threat model that we're trying to guard against in establishing a secure communication between two points because we can't guard against it. There is no practical means to guarantee that with the way our computers are designed today. In that sort of scenario, in a communications scenario, we're dealing with a situation where the bad guy has no access to the key. They may have access to the result of the key, and typically do. They would have access to the encrypted data, the results from applying this secret encryption key. But they don't have access to the key itself.

And so the idea is that we need to generate a secret of some length, either in one end or in both ends, such that knowing - and this is where this concept of randomness comes in.

Knowing even like prior keys or even maybe future keys, there's no way to predict the key that we're generating, no way to determine what the next key is going to be from the standpoint of the attacker who can see the consequences and may even have information about recent history of keys, but not the one we're trying to get directly.

Now, an early algorithm that I've referred to a couple times in the past is very weak. It's called a "linear congruential pseudorandom number generator," which is a fancy term for a very sort of simple-minded random number generator. It's the kind that was in, for example, the BASIC language when we were all cutting our teeth on programming in BASIC in high school, where you'd give it the Rnd() function, and it would spit out something that looked random. Turns out that those were very, very poor random numbers. But they were typically random enough for the little Star Trek simulations we were running, or rolling the dice, or guess the number between 1 and 10, the kind of things that the computers were being used for at the time. A linear congruential pseudorandom number generator takes a number and multiplies it by one constant and adds another constant in order to get the next number, the next pseudorandom number.

Well, it's very fast because all you have is a multiply and an add. But it's also very weak. It works within a register of a certain size, which might be 16 bits or 32 bits. You're multiplying a constant and then adding a constant. And that's getting a different, for example, 16 or 32-bit number. And then you do it again, and you get a different 32-bit number. Essentially, if you pick the two constants well, it will march the number all over the territory. If you imagine, like, a big - if you had a 16-bit number, you could imagine that is like in a grid, where the pseudorandom number it generates is jumping around within this grid in what looks to you an obviously not visually detectable pattern. But any kind of cryptanalysis would immediately, if you gave it a few numbers, it would break the algorithm and determine what the constants were and be able to go back in history or forward in time in either direction in the sequence of very poor pseudorandom numbers.

So that's an example of an early, very bad pseudorandom number generator. What's happened as we've moved forward, and as we've had to get cryptographically strong pseudorandom numbers, is a huge amount of industry and creativity has come to bear, to the point now where we have a good understanding for the strength of random numbers. Cryptographic algorithms that we've talked about often like AES-128 or 256 can be used, as can cryptographic hashes. And we always, however, have a problem of this determinism because in that simple example I gave with this linear congruential pseudorandom number generator, it obviously itself is a weak random number generator. But it's still being - it's being hosted by a computer. Well, our cryptographic approaches and hashes are also being hosted by a computer. So we have this problem of how do we break free of operating in a deterministic environment? How do we get something which is good enough? How do we know it's good enough? How do we prove it's good enough? And what is good enough? And that's our subject for two weeks from now, solving this problem.

Leo: Can't wait. That's fascinating stuff. Next week, of course, we're taking questions. And if you've got a question for Steve, you go to GRC.com/feedback and leave a question there. And we'll do 10 or so good questions about this or any other topic. Actually, I bet if we got some good questions about randomness, this would be a good, so to speak, seed for the following week's conversation. I think this is fascinating. And anybody who learns programming learns about Rnd, you know, the random number functions, and then learns that they repeat.

Steve: Well, if you...

Leo: It's totally predictable.

Steve: Yes. It's really bad. And in fact there was a randomize function, you'll remember, that sort of was supposed to break us out of that repetition. Now, sometimes having numbers repeatable is very useful. For example, if you've got a process where you're doing some modeling, and driving a model with random numbers, sometimes you do want to be able to repeat exactly the same string of random numbers. Or say, for example, that - remember when eEye, the security firm down in Aliso Viejo, was pounding on Windows, throwing random stuff at the Windows API until it broke, well, they may want to be able to set up an identical system and give the same sequence of not really random but pseudorandom data to Windows and watch it fail. So there are times when you actually do want repetition in your random numbers.

Leo: Interesting.

Steve: Although not always. There's a page that is public on GRC that's GRC.com/ - and you can put it in right now, Leo, if you're curious - GRC.com/r&d/js.htm, as in JavaScript. It is actually - that's a platform that I developed, it's the first JavaScript code that I wrote after I taught myself JavaScript recently for the project I'm working towards finishing now, which is the Passcode Designer that I've spoken briefly of before. Anyway, that page, GRC.com/r&d...

Leo: I love how you comment even your learning JavaScript stuff. I love it, Steve. This is a pro. This is a pro. He diagrams the stuff that's not even public.

Steve: Yeah, it was just [laughing].

Leo: Such a pro, Steve. You're unbelievable.

Steve: Well, that is my first JavaScript. And I had to solve the problem of running something in the user's browser which didn't necessarily have access to a lot of entropy because JavaScript's own random number generator is known, many of them are known not to be good. And we really can't count on the user to give us lots of randomness. For example, when you're establishing an encrypted drive in TrueCrypt, you know, they have you move the mouse around. And it's like, that just always annoyed me because that's - you could demonstrate that's not very random compared to what's available. So anyway, so I solved the problem, as that page demonstrates and documents, for anyone who's curious. And it's the heart of the random number generator technology which is then going to surface in the Passcode Designer that I hope to be showing our listeners very soon.

Leo: I have to say, I have no idea what you're talking about here, but it's very - it's very cool. I just love it.

Steve: That's my first JavaScript.

Leo: Steve Gibson is at GRC.com. That's where you go to get SpinRite. And you've got to get SpinRite. It's the world's best hard drive maintenance and recovery utility.

Steve: It'll even take your temperature.

Leo: It'll even take your temperature. GRC.com. While you're there, lots of free stuff, including ShieldsUP!, Wizmo, the passwords, all of the stuff that he does is so great. The Perfect Passwords. You can get a good 64-character password, totally random, from Steve.

Steve: 3,700 people do every day.

Leo: Is that - wow.

Steve: 3,700 a day.

Leo: That's almost encouraging. It's like, wow, there's somebody who cares, there are 3,700 people a day who care about security.

Steve: Yeah.

Leo: I'll take it. GRC.com. Steve, it's always a pleasure. Next week we answer questions. Two weeks from now we continue our discussion of how random numbers, real random numbers can be generated from pseudorandom situations.

Steve: Where we get them and how we get them from a computer that really doesn't want to give them to us at all.

Leo: It's a fight.

Steve: Yeah, it is.

Leo: It's a battle. Thanks, Steve. Have a great week, and we'll see you next time on Security Now!.

Steve: Thanks, Leo.