The Browser Extension Ecosystem

Description: This week we prepare for the first-ever Presidential Alert unblockable nationwide text message. We examine Chrome's temporary "www" removal reversal, check out Comodo's somewhat unsavory marketing, discuss a forthcoming solution to BGP hijacking, examine California's forthcoming IoT legislation, deal with the return of Cold Boot attacks, choose not to click on a link that promptly crashes any Safari OS, congratulate Twitter on adding some auditing, check in on the Mirai Botnet's steady evolution, look at the past year's explosion in DDoS number and size, and note another new annoyance brought to us by Windows 10. Then we take a look at the state of the quietly evolving web browser extension ecosystem.

High quality (64 kbps) mp3 audio file URL: http://media.GRC.com/sn/SN-681.mp3
Quarter size (16 kbps) mp3 audio file URL: http://media.GRC.com/sn/sn-681-lq.mp3

SHOW TEASE: It's time for Security Now! with Steve Gibson. This week we talk about a lot of things. I'm Jason Howell filling in for Leo. We talk about the Presidential Alert that's coming up. It's been postponed, but it's right around the corner. Also Chrome's temporary www removal reversal. There's a link that promptly crashes Safari OS, so you don't want to click that link. And Steve talks about the state of the web browser extension ecosystem and some of the bad habits that we've gotten into over the years. All that and more, coming up next on Security Now!.

JASON HOWELL: This is Security Now! with Steve Gibson, Episode 681, recorded Tuesday, September 18th, 2018: The Browser Extension Ecosystem.

It's time for Security Now!. This is the show where we talk about all the latest security news, the happenings, everything security related on this very show. I'm Jason Howell, filling in for Leo Laporte, joined by none other than Steve Gibson. How you doing, Steve?

Steve Gibson: And you are at this point, Jason, a fully trained-up co-host for Security Now!.

JASON: Oh, yeah. In fact, if you want to take the week off, I got this. I will just interpret your notes as best as I possibly can.

Steve: Well, so this is your third and final week co-hosting while Leo's on vacation.

JASON: That's right.

Steve: And I think we'll end up with a bang here.

JASON: Excellent.

Steve: So this is #681, which I titled "The Web Browser Extension Ecosystem" after some news of a Firefox change in two major releases from now sort of got me thinking about, like, we've talked about - you and I have in the last couple weeks talked about browser extension security issues where malicious extensions get into people's machines. So I did a little bit of poking around. And I was a little, I mean, even more than a little surprised by the number of extensions the majority of people have.

And in fact this is the chart that is this week's Picture of the Week. It shows a histogram by number of browser extensions. And oh, my god, even at the tail, it's like, you're kidding me. Somebody, like there's a non-zero number of people who have 69 browser extensions. I don't know how that thing even gets off the ground with 69 extensions. But believe it or not, the peak of this histogram looks like, what is it, that thick bar, it's like 13 to 14...

JASON: You have 15, somewhere around there.

Steve: Yeah, it looks like, yeah. So that is to say the most people have 13, 14, 15 individual browser extensions. And it doesn't drop by - it's not like it's a super sharp peak. And we're talking from I think they surveyed several hundred thousand. Anyway, we will get to it at the end of the podcast. But I wanted to sort of step back and take a look at what has happened to the whole idea of extending the browser and what it means because it's funny, too, because in the research I ran across references to that old Conduit extension that was so nasty and was infecting people's machines. And just, I mean, they've been a problem.

And in fact even now, what is it, I guess it's when I'm updating Java on a machine where I need to have Java because of other client stuff that I'm doing, they're still trying to slip an Ask toolbar or something in, you know, the little "install this because it'll improve your life." It's like, ugh, thank goodness I saw that it was still checked, so I was able to turn it off. And I've missed it a few times and had to go back and take it out.

JASON: Yeah, you've got to be really careful with those. It's super easy for those to just kind of slip right through.

Steve: Yes. And I'm sure that some of us who are in charge of keeping other people's machines alive have sometimes gone to a machine we haven't seen for a while, and half of the vertical height of the browser's page space is bars. It's like toolbars running. It's like, how do you even see the Internet through the little...

JASON: But it's useful, Steve. Just think of how productive you are when all of those features are always in front of you.

Steve: Oh. And always running, always having to get going whenever your browser loads and so forth. Anyway, we'll have fun talking about that at the end of the podcast, which is titled "The Browser Extension Ecosystem" because it has become that. And we'll talk about even extensions fighting each other for dominance and kicking each other out, which also happens. But anyway, we have a lot of other stuff to talk about. We need to prepare. It has been postponed. It was going to be day after tomorrow. But the recent hurricane caused FEMA to decide, let's bump this back. But we are still going to have the first-ever Presidential Alert, which is an unblockable, nationwide - if you have a phone, and it's on, you will get a Presidential Alert a couple of weeks from now, a text message.

It's funny, too, because the last time I was up there it was - I think it was Christmas before last. It was when Leo's plan at the time was to have a couple hosts join him for an end-of-the-year kind of a holiday, I don't know if it was TWiT. But so I think it was two years ago, so that we knew that President Trump was - he had been elected. He was

going to become our President. And somehow this topic, I mean, the idea that there was going to be the ability for the President to simultaneously send everybody in the U.S. a text message that they could not block was discussed during that holiday event. And so anyway, it's happening. It's two weeks and one day from now. But anyway, I think I pretty much already covered that topic, so we may skip that when we get to it.

But we've also got to talk about Chrome's temporary www double removal reversal that we talked about last week, that suddenly www was disappearing. And now it turns out some people said no, it isn't. Well, it turns out they had changed their mind very quickly, but only temporarily. We'll cover that. We've got Comodo's somewhat unsavory marketing, which was brought to my attention by one of our listeners who tweeted a note to me. We're going to discuss a forthcoming solution to the problem of BGP, the Border Gateway Protocol, hijacking, which we've discussed extensively previously, that is, the hijacking problem, not the solution.

California recently passed some legislation waiting for the governor's approval or signature for IoT security, which is kind of a mixed blessing because of course it's government. We'll deal with the return of cold boot attacks. Oh, and also take a look at this, click on it, and your iOS device crashes. I tried it, and it works.

JASON: Oh, another one of those.

Steve: Both in 11 and in my freshly updated iOS 12. So Apple didn't get it fixed. We also have - I want to congratulate Twitter on adding some very welcome auditing. It's sort of basically catching up with what other people have been doing. We're going to check in with the Mirai botnet's steady evolution, look at the past year's explosion in DDoS number and size. Also note a new annoyance brought to us by Windows 10. And then we actually have an interesting bit of errata and some miscellany. Then we're going to take a look at the state of the quietly evolving, but boy, it's really become something, web browser extension ecosystem. So I think you'll have a good sendoff, Jason.

JASON: Absolutely, and I've already counted the number of extensions, at least on my Chrome OS device here, my Pixelbook. And it's, like, right there at the peak of that graph. It's like 14.

Steve: Interesting.

JASON: So I'm right there with everybody else. And I even tried to trim that down a couple of weeks ago.

Steve: In fact, it's interesting that basically what you have is a battery-powered browser. I mean, essentially Chrome OS is, as we know, it's just basically a browser. Because, I mean, the browser has become, as we've often said on this podcast, our portal to the 'Net. And with web-based apps and the increasing efficiency of code running within the browser's control, that only looks to expand in the future. So, interesting. I'm down at, like, five, I think. But on the other hand, this system has recently been set up. So I haven't had a chance to accrue any barnacles yet.

JASON: Yeah, yeah. Barnacles, it's a good way to put it. Maybe we should change it to web barnacles. I think that's probably more appropriate. All right. So, yes, we've got the Presidential Alert that was supposed to happen, I think two days from now; right? But that's been pushed back.

Steve: Correct. Correct. It was going to be Thursday, but what was it, Hurricane Florence, I think that was her name, we're still recovering from that. FEMA's busy, didn't want to, like - well, and I think they probably worried that, if somebody got something

that says "Presidential Alert" on their phone while a lot of people were literally under water, that could be a problem. So anyway, so it's been pushed back to October 3rd.

Okay. So we have two things. There's the national EAS, which is the Emergency Alert System, which I don't know about you, Jason, but it just drives me nuts when I get this [mimicking loud alert sound], oh my god, on my cable, like switches over and everything stops, and it's just like, okay, thank you. And they say it's monthly, but it seems like it's a lot more often than that. So that's existed for a long time. What hasn't - oh, and it's been tested, I had it in the notes, three times previously. Oh, yeah, there it is, on November 2011, September 2016, and September 2017, an emergency alert system different from what we're getting on our cable boxes. What's new is called WEA, which is Wireless Emergency Alerts, which will be tested for the first time beginning at 2:18. How they chose that time I have no idea. Not 17, not 19. No, we're going to do this at 2:18 p.m. Eastern Daylight Time.

JASON: There must have been a reason.

Steve: On October 3rd. So who knows? So what does that make it? That's Pacific Time, that's 11:18 a.m. for us on the West Coast, 2:18 p.m. on the East Coast in the U.S. And so what happens is, starting at that time, for approximately 30 minutes, every cell tower in the United States that's hosted by participating wireless carriers, and more than 100 carriers are participating, I mean, they didn't say every one. But probably, I mean, it has to be all of them. And so what will happen is our phones will give us a text message with the header reading "Presidential Alert" and text that reads, first four words, all caps, "THIS IS A TEST," which is a good way to start, "of the National Wireless Emergency Alert System. No action is needed." And we are told that the phones, first of all, all of our phones have this capability built in, which has never been exercised.

So the assumption is that we'll get, during the 30-minute continuous broadcast, the phone will only give us one text message, let's hope, and that all phones within range will receive that one text message. So anyway, I'm sure we'll talk about it in two weeks because two weeks from now it will be the day following that podcast, being a Wednesday. So anyway, it was supposed to be in two days. I had it in my notes. I double-checked. They moved it. It's like, oh, well, okay, we'll talk about it anyway and then remind everybody in two weeks.

JASON: I know these things are important, and I know that it's important to set it up. I will point out that on - this is Android Pie. And there is a - I don't know if there's any sort of over-the-shoulder camera, so I'll just kind of show you from a distance. There is an Emergency Alerts section in Settings where you can toggle on or off the different types of settings, the different types of alerts. So Amber alerts, extreme threats, severe threats. I have an entry there for required monthly test, and I have that off, so I don't get the monthly test ones. So maybe check on your phone and see if there's a way to, like, switch that one off.

Steve: Wow.

JASON: But there is nothing in here where you could turn this particular warning off. The presidential one guaranteed comes through, from what I understand.

Steve: Correct. And, I mean, really, in our wired world today, if there were something that was, like...

JASON: Major.

Steve: ...of absolute national consequence, we want something like this. And given that everybody is walking around with cell phones, that's the way to get a message out to everyone. I mean, you have to be very, very, very careful with how you handle it. I mean, we know what happened in Hawaii.

JASON: Yeah, that's what I was going to mention.

Steve: When the guy pressed the wrong button, and Hawaii thought they were under attack. So it's like, okay, let's...

JASON: I mean, could you even imagine what that must have been like, to get that message and not get any sort of an update to it for, like, 30 minutes, an hour. I can't remember how long it took, but people for a very considerable amount of time assumed that it was real.

Steve: They thought missiles were in the air. And it's like, oh, goodness.

JASON: Yeah. Craziness.

Steve: Okay. So Chrome 69. To www or not to www? Well, so it's back, kinda. The Chromium blog wrote: "In Chrome M69 we rolled out a change to hide special-case subdomains" - okay. First of all, okay, special-case subdomains? So www is a special-case subdomain? Okay, notice they're no longer calling it "trivial." I think they got slapped pretty hard because they originally said, oh, these are trivial. Okay. So now they're special case. Many people argue that. But anyway, so what they said was "...to hide special case subdomains 'www' and 'm' in the Chrome Omnibox," which is what they call their multifunction URL field. "After receiving community feedback about these changes, we have decided to roll back these changes in M69 on Chrome for Desktop and Android."

Now, I'll just pause here to say that I absolutely verified, as I mentioned last week, that this was all working for me prior to the podcast. Several people subsequently said, "What are you talking about?" And now for me indeed it is no longer working. So they reached out and tweaked my Chrome. And I don't know if the other people had already been tweaked or never got tweaked or untweaked, I don't know. But anyway, so it's gone for now.

However, they continue: "In M70" - so that's the next major release - "we plan to reship an adjusted version. We will elide [as they put it] 'www' but not 'm.' We are not going to elide 'm' in M70 because we found large sites that have [what they call] a user-controlled 'm' subdomain." Okay, and I thought, what? Okay, so the point being that the user can decide if they want the mobile version or the desktop version by choosing or not choosing to prefix the domain with an "m dot." And so they learned the hard way - because of course they immediately just, first of all, they just got rid of it, it's like, ooh, ouch, okay - that that should be user controlled. Then they continue, saying: "There is more community consensus that sites should not allow the 'www' subdomain to be user controlled." And it's like, okay. They're on the peninsula. What are they smoking? I don't get this. But believe me, a lot of people don't, either.

They continue, saying: "We plan to initiate a public standardization discussion with the appropriate standards bodies to explicitly reserve 'www' or 'm' as special case subdomains under the hood. We do not plan to standardize how browsers should treat these special cases in their UI." Huh? Okay. "We plan to revisit the 'm' subdomain at a later date, after having an opportunity to discuss further with the community. Please leave feedback" - oh, did they get some - "or let us know about sites that have user-

controlled subdomains at 'www' or 'm' in this bug." Because, you know, this is their Chromium bug list is where this all lives.

So, okay. And on top of this, remember that I briefly noted last week that Chrome 69's initial implementation not only was controversial by doing this unilaterally, but also buggy. And it would match and remove multiple www's which were embedded within a single domain name, not just the leading www - this in a code patch they have since fixed. So, however, users who are voicing opinions remain unhappy with the Chromium team's stated intention to return to removing a leading www from the URL in Chrome 70. The Chromium team solicited comments, and they received many. I pulled three out. Actually, I got these - these were pulled by Lawrence over at Bleeping Computer, and I like his taste in comments. So I'll just - I'll quote those.

First one, someone posts: "What is the reasoning behind the decision not to wait for the standardization discussions? As the majority market shareholder, it's incumbent on Chrome and Chromium to be cautious when making arbitrary decisions about how connection information should be displayed or, in this case, deliberately transformed and obscured. The urgency with which this change is being pushed is baffling."

He continues: "I strongly disagree with the idea that we should be connecting to one host name while displaying another in the address bar. Absent legitimate discussions with the appropriate standards bodies, this decision feels myopic at best, and wrong-headed at worst."

Comment 2: "Standardization discussions aside, no changes to the look and function of subdomains should happen unless it is an opt-in setting. SaaS, ESP, and ISP providers take great care to manage their subdomains and use 301 redirects to bring users to the right location." It's interesting he should say that because that's exactly what I had mentioned GRC does deliberately.

He continues: "A browser should not determine this function unless it is completely a user choice. Perhaps Google/Chromium would like to respond as to the exact end game and reason for such a change." It does feel like they ran out of other stuff that was more important to fix and so, oh, let's screw up the domain that we're displaying. Anyway: "Dictating a change which is controlled through DNS, but shows up differently on a browser, is just plain confusing and wrong. I don't agree with what Safari and Windows have done, either."

And the third comment, a shorty, says: "At least make it a toggleable option. I don't mind if URLs are elided by default, but please have an option to disable this."

So anyway, I think these are appropriate responses. I mean, first of all, I have always and long said that it's unfortunate that the Internet and the web has evolved at all the way it has. I mean, http://? Really? You know, we're trying to explain that to our moms? And then, you know, should I use www or not? I mean, it's a mess.

But I completely agree that this is not something that a single browser, despite being the majority browser on the Internet, should decide. That is, this should be, I mean, the only reason this exists at all, the only reason there is a Google and that there is a Chrome browser is that there are standards which preexisted their existence. And it's the compliance to those standards and the interoperability of those standards which allowed Google to get created and allowed them to create a browser. They've been standing on the shoulders of others who set these standards and then abided by them. And now they're just unilaterally saying, "We're just going to change what the user sees."

So I'm in 100% agreement with the first two comments that say, okay, yes, let's talk about this. But if we're going to do this, this needs to be a global decision, and all the

browsers need to decide how they're going to handle this. And that allows then websites to similarly decide, first of all, to participate in this, which didn't happen. And as a consequence, Google realized, ooh, crap, this "m dot," that was important, and we took it away.

So again, I just don't think - I think they've overstepped. They've been having fun. They've been saying, oh, well, we're going to expire our recognition of certs early. We're going to do this. We're going to do that. We're going to stop supporting SHA-1. And they've tweaked some feathers by cutting their own path.

I think maybe they've stepped a little too far this time. They immediately backed away, said they're going to go forward again. Let's hope that doesn't happen. Again, I would love to see http://www. all go away. Just kill it. But we have to do that universally, by agreement, with lots of discussion and deadlines to make sure that things are ready for it. And then all the browsers have to be in sync with this change. It would be great to fix this, but I don't think simply blinding users of one browser to this arbitrary change, I mean, I don't think that makes any sense. It's just - it's wrong.

JASON: I think Google has many times before had big ambitions or big ideas kind of similar to this. And I think they believe that because Google is Google, and Google/Alphabet is so large and influencing and all that, that many times they can just decide to do a certain thing, and that's going to create critical mass that moves everyone there. It kind of falls into Google's kind of ethos and a lot of Silicon Valley ethos of move fast, break things; that whole mantra of just, like, do it, and then ask for forgiveness later and see what happens. Doesn't always work out, obviously.

Steve: Yeah, they definitely broke something this time. Okay. So I got an interesting tweet from Sean Nelson that I saw just before launching last week's podcast, and I shot back a note thanking him for the information, telling him that there wasn't time to get it into last week's podcast. But I told him that I thought it was worth sharing. Which was sort of - this evidences an interesting, I would think a little slimy and questionable marketing practice.

He tweeted to me: "I manage a couple hundred school district websites that use Let's Encrypt certs for HTTPS." And first of all, let me just stop and say that's a perfect example of an application where it makes sense to save school districts, a couple hundred school districts, each needing an expensive domain cert only, I mean, not because they have to have it, but because they want to have encryption. That's a beautiful application of Let's Encrypt. So probably before Let's Encrypt this wasn't being done. These things were - they weren't able to offer HTTPS protection; or, if so, only at substantial cost. So bravo for the change that we've seen over the last couple years. A couple hundred school districts, he says.

Let's Encrypt, as we know, "renews automatically," he writes, "using a script on my servers. I just got a breathless phone call from Comodo warning me that one of my hundreds of domains has a cert that will expire soon, like," he says, "70 days from now, and I'd better buy a three-year certificate or I would be sorry. When I told her that I know what I'm doing, and it renews automatically, she made it sound like I was being irresponsible by risking the certificate renewal every three months rather than every three years." He finishes, saying: "I wonder how many other LE users are getting these phone calls from Comodo desperately trying to scare them into buying a certificate instead of using a free LE cert."

Anyway, I got a kick out of that. I thought that was an - so it's an interesting marketing approach since, when you think about it, every website certificate clearly displays its expiration date. So desperate sales agents with nothing better to do could just attempt to

roust users whose certs are nearing retirement and renewal and say, hey, just wanted to make sure you knew, and why don't you buy ours?

And of course I'm sure that Sean was being a little bit rhetorical because, unless somebody receives the marketing call who is not the admin, who as Sean says knows what they are doing, anybody using Let's Encrypt is well aware that the system just takes of itself, and that it starts early so that it's sure that it's able to get a renewal in time and that it's a constantly rolling renewal system which actually also solves the problem of long-life certificates that need to be revoked because that's of course famously another thing that Chrome - you were talking about Chrome going off on their own direction.

And I of course brought a lot of attention to the fact that Chrome's certificate revocation system is completely broken. I mean, like the most broken of any browser there is. So constantly rolling short-life certificates is a solution to having revoked certificates otherwise living for a long time. So anyway, I just got a kick out of Sean's note. And I'm sure that anybody using Let's Encrypt knows not to pay attention to such a somewhat questionable marketing call.

Now, we've talked about Border Gateway Protocol a number of times. In fact, a couple months ago there was one large ISP, I think in Portugal, maybe Brazil, I don't remember where, who was finally shut down by the collective decision of all other ISPs on the Internet because this one group was causing their routers to advertise that they handled small networks of IP space, IPv4 space, that they had never been allocated. And this had been going on for years. So if nothing else, this is a testament to the fact that the people in charge take their responses seriously, don't make decisions on a whim, and you've got to really demonstrate that you're determined to break the rules for years before the hammer finally falls. But when it does, as for example happened with Symantec and their misissuance of certificates, you're out of the game. Sorry. Go find something else to do.

So in this particular case this sort of demonstrates the problem. This one large ISP kept claiming to adjacent routers that it was the owner of small blocks of IP space. Border Gateway Protocol is that communications protocol by which routers talk to each other and update their routing tables. And routing works by finding the most specific path to a network. So advertising small routes, that is, a fewer number of host IPs within a network, that's a very specific path. So those specific paths would get propagated automatically from one router to the next, out across the Internet.

And what ended up happening then was that essentially they were stealing, successfully stealing IPv4 networks, small subnets from their actual owners. The actual owners would stop receiving that traffic, and it would go to this hijacker instead. And BGP attacks or misconfigurations have also happened where a well-meaning ISP, someone just types the wrong /subnet limiter or specifier into a routing table and suddenly ends up receiving traffic they don't want because their connections can't handle it. But as a consequence of the mistake, they're saying, yeah, we have pretty much all of the IPv4 space. And local routers go, oh, we'll just send our stuff to you, and off they go.

So it can be inadvertent. It can be deliberate. But what's been clear for years is it is a problem. So what we have now is the emergence of a solution. The final piece of the standard to protect against Border Gateway Protocol hijack attacks and also misconfigurations, which can happen, they don't last for long because it pretty much comes to everybody's attention. But we have the first official draft of the final of three pieces. The effort is termed SIDR, standing for Secure Interdomain Routing, and that's what BGP is about. It's these border gateways are the routing between domains.

So we have three interrelated protocols. There are two that were ratified and finalized about a year ago. It was in September of 2017. RFC 8206, which is titled BGPsec, as in,

you know, we're all used to talking about DNSSEC. That's DNS Security. This is BGP Security, Considerations for Autonomous System Migration. And then RFC 8210, which is known as the Resource Public Key Infrastructure, RPKI, to Router Protocol Version 1. And it actually updates a previous RFC, which was Version 0.

And so, for example, there they said: "In order to verifiably validate the origin Autonomous Systems and Autonomous System Paths of BGP announcements, routers need a simple but reliable mechanism to receive Resource Public Key Infrastructure, prefix origin data, and router keys from a trusted cache. This document describes a protocol to deliver them." And so our savvy listeners can see that this is very much modeled on the DNSSEC approach, the idea being that routers will in some way arrange to sign and provide signatures which can be verified of the routes which they authoritatively control. So that's what the system is going to be providing.

This third document, a year later, which was recently finished - and I should mention that this is the NIST and DHS are the two U.S. governmental organizations behind this, although this has been a huge multi-participant effort because they're working to get it right the first time. They just released a mind-numbing 264-page document which is - do I have the title of it? Oh, yeah. They called it the BGP Route Origin Validation (ROV) standard which, when coupled with the other two existing protocols, promises to help ISPs and cloud providers protect against BGP hijack attacks. And of course, as we know, it'll do more than that. It'll also, by creating a means for validating what a router is claiming, the addresses that a router is claiming to be responsible for, it will solve this problem of both inadvertent mistakes and attacks.

So I have in my show notes here a bunch of bullet points from it. But we already understand what this is about. It is intended to strengthen BGP to provide using the public key infrastructure, much as DNS is planning to, or actually, I mean, has. We now have the root DNS server signed, and the effort is stalling because it just requires people doing things, and people don't want to do things unless they have to.

In this case, we at least have standards. And over time, routers will probably be updated to support them. And we will eventually move towards a world where it's no longer possible for someone like this small ISP to deliberately commandeer chunks of the Internet for the purpose - actually in this case, as we talked about before, they were reselling this IP space to spammers, stealing legitimate IPv4 space which had not been blacklisted as spam sources and essentially reselling it to these spammers. So it's still early days, as we've talked about. And you can tell because, when you look at the URL, it often says http://www for no obviously good reason.

JASON: Again and again and again. So California has this IoT legislation, and I'm wondering how much actual teeth it has, or whether it's even satisfactory to begin with, because it's a pretty light touch, I feel.

Steve: Oh, goodness. Well...

JASON: There's some good stuff in there, though.

Steve: Well, actually there's, yes, a little bit of goodness. So a little bit of background. The California State Legislature recently approved "SB-327 Information Privacy: Connected Devices," that was the bill, and handed it to our Governor, Jerry Brown, to sign into law. It purports to introduce security requirements for connected devices sold in the U.S., and defines these devices as any device which connects directly or indirectly to the Internet and has an IP or Bluetooth address. Okay, which is kind of everything. Unfortunately, legislators. This is really why anything like this needs to get done by technologists and then sort of just like, get out of our way, please.

JASON: Right, right.

Steve: Anyway, so quoting from this: "This bill, taking effect on January 1st, 2020, will require a manufacturer of a connected device, as those terms are defined, to equip the device with a reasonable security feature or features" - oh, so you can have more than one, how nice - "that are appropriate to the nature and function of the device; appropriate to the information it may collect, contain, or transmit; and designed to protect the device and any information contained therein from unauthorized access, destruction, use, modification, or disclosure, as specified." Okay. Which basically said nothing, but used a lot of words to say it. So this, I guess, we can say, well, this is evidence of intention. I don't know what this means.

But there was a little bit, there was a little tiny sliver of silver lining here because at one point the bill says: "If a connected device is equipped with a means for authentication outside a local area network, the authentication system must meet one of two criteria. First, if the device uses a default password, the password must be unique to each device; or, two, the device must prompt users to set up their own password whenever the user sets up the device for the first time." So that's a breakthrough in, like, we don't have anything like that now. And that's very clear. That first paragraph could not be less clear. This one says no default passwords if there's authentication outside a local area network.

Which is to say, all of these situations where there have been routers with a default password exposed to the WAN, that is, to the Internet, the Wide Area Network, those can't be sold. And I don't know, I mean, they're saying in the U.S. I guess they can't be sold in California. California is a huge market. So that gives this legislation some teeth. So it hasn't turned into law yet, but it certainly does represent a nice step forward.

Now, any legislation is going to have detractors. And last Monday, in reaction to this, Errata Security's Robert Graham, who is the - he was the original guy behind the Black Ice Firewall many moons ago, back when personal firewalls were a thing, and they were added to operating systems like Windows 98 that never knew what a firewall was. He generated a long posting from which I'll just share the first three paragraphs. He wrote: "California has passed an IoT security bill awaiting the governor's signature or veto. It's a typically bad bill based on a superficial understanding of cybersecurity/hacking that will do little to improve security, while doing a lot to impose costs and harm innovation."

He writes: "It's based on the misconception of adding security features." He says: "It's like dieting, where people insist you should eat more kale, which does little to address the problem if you're pigging out on potato chips. The key to dieting is not eating more but eating less. The same is true of cybersecurity, where the point is not to add 'security features,' but to remove 'insecure features.'" Okay. No one's arguing that.

"For IoT devices, that means removing listening ports and cross-site/injection issues in web management. Adding features is typical 'magic pill' or 'silver bullet' thinking that we spend much of our time in info security fighting against. We don't want arbitrary features like firewall and antivirus added to these products. It'll just increase the attack surface and make things worse. The one possible exception to this is 'patchability.'"

He says: "Some IoT devices can't be patched, and that is a problem. But even here, it's complicated. Even if IoT devices are patchable in theory, there's no guarantee vendors will supply such patches or, worse, that users will apply them. Users overwhelmingly forget about devices once they are installed. These devices aren't like phones and laptops which notify users about patching." And of course he's echoing many of the concepts and suggestions of this podcast. So anyway, it'll be interesting to see where this goes, whether it gets signed. It feels to me like Jerry Brown is probably going to sign it. I don't know, I mean, that first intent paragraph sort of says, I mean, you can reasonably...

JASON: What does "reasonably" mean? Yeah.

Steve: Yeah, yeah. Exactly. You could just say, oh, we have port closed, you know. I mean...

JASON: We put the icon of a lock on the packaging.

Steve: Yeah.

JASON: That's enough.

Steve: Exactly. Yeah, we said it's secure. So be happy. Okay, fine. Anyway, who knows what's going to happen. I mean, this is the beginning. We know we have a problem. When we talk a little here before long about what has happened with DDoS attacks, it is entirely a function and a cause of IoT devices. And, boy, is it sobering. So this has to get changed. And of course we talked last week, Jason, you and I, about what's gone on with routers and the number of routers which are now being hijacked and enlisted for DDoSing. Okay, so...

JASON: Yeah. I mean, I suppose it has to start somewhere. So I'm happy to see that. I like the password suggestions. Those are very useful, at least.

Steve: Yeah. And in fact it's, absent that, it has been possible for manufacturers who absolutely could care less to set, I mean, all of our routers have admin/123456 when you install them. We all know to change that. So it will, on the other hand, that's local access. And that doesn't have to change. It's the WAN side access that does have to change. But many routers you can turn on global access. This law says, if you do that, it cannot be with the same password.

So there will have to be some firmware updated. And I would argue against Rob's position that doing that is a bad thing. If you turn on WAN access and the law states you must then prompt for a non-default password, I don't see how that's anything but good. But I certainly, you know, all of the other points he makes are points we have long made on the podcast, which I naturally agree with.

Okay. So 10 years ago, in 2008, we talked about the problem of sleeping PCs and laptops which go into that ACPI S2, I think it is, it's either S2 or S3. That's where everything is shut down, even the processor, except RAM is kept powered up. That's the so-called "sleep" that we're used to using on our machines. And we all see it. Typically on laptops the power light throbs very slowly to tell us that, oh, it's asleep. But when you open the lid, it blinks back on. It doesn't take long for the device to come back. It's not rebooting, it's basically powering up all the systems that have been idled, and then it continues where it left off.

So that was 10 years ago. What we talked about then was so-called "cold boot attacks" because what was recognized a decade ago was that the RAM was still - RAM still had its contents. And if you could arrange to reboot the system without the RAM powering down, or even not powering down for long, if the RAM wasn't wiped proactively, and you could boot into a different OS, you had access to the state of RAM at the time that system slept. And if, for example, BitLocker was in use and had been unlocked, or for that matter TrueCrypt, which was a big deal 10 years ago, or any other important crypto keys, I mean, basically RAM is there. And you can rifle through it as a third-party OS that gets control.

And as we know, there's also literally freezing cold boot, where you spray Freon on the chips. And we've talked about how, as a consequence, the fact that DRAM are actually a large bank of tiny capacitors which are slowly losing their charge, thus the need to

refresh the contents before they lose so much charge that you can't tell whether they used to be charged or not. If you sprayed them with Freon, you could actually remove them from one machine and stick them in another and power them up in time to preserve their contents.

So what had happened since then is that the BIOSes began, like from 10 years ago, when the so-called "cold boot attacks," where you don't take the RAM out and physically freeze it, you simply keep the RAM alive and reboot into a hostile OS environment, which then looks at the RAM and says, hey hey hey, we've got some goodies here. What BIOSes had started doing is, first of all, being less willing to boot from, for example, an external drive, a little more resistant to that, and also proactively wiping the contents of RAM.

So last Thursday two researchers from F-Secure gave a presentation at the Sec-T Conference titled: "An ice-cold boot to break BitLocker." And the teaser for their presentation wrote: "A decade ago, academic researchers demonstrated how computer memory remanence could be used to defeat popular disk encryption schemes. Not much has happened since, and most seem to believe that these attacks are too impractical for real world use. Even Microsoft has started to play down the threat of memory remanence attacks against BitLocker, using words such as: 'They are not possible using published techniques.' Well, we will publish techniques that allow recovery of BitLocker encryption keys from RAM on most, if not all, currently available devices. While BitLocker is called out in the title, the same attacks are also valid and fully effective against other platforms and operating systems."

Okay. So what did these guys do? It was pretty simple. Keeping the target machine - in this case a laptop, but it could be a desktop - powered up, they opened the back and hooked onto the readily available and exposed flash memory with a programmer and tweaked the machine's BIOS setting storage chip where it does the nonvolatile BIOS settings in order to cause it to [do] what they wanted. They tweaked the settings to disable the RAM memory overwrite, which it turns out is not exposed to the UI, but does exist; and they enabled booting from external devices. And that's it.

They then carried out a traditional cold boot attack by rebooting that powered up system into a special program on the USB stick. It's special because it needs to minimize its RAM footprint so as not to overwrite anything valuable in main memory. But that's easily done. And they were then able to obtain the BitLocker or other disk encryption keys in order to have access to the drive.

And remember that, since this defeats BitLocker or whatever other whole disk encryption might be there, they're not only getting the contents of RAM, but are also able then to decrypt the entire contents of everything stored on disk, which is then accessible externally without the layers of protection provided by the OS, which it is assumed is the only entity able to perform that decryption. Which means you know you have root, I mean, absolute full root unrestricted access to the file system on that system on which you have performed this attack.

So what do we do? We can expect that, in reaction to this, machines will get further hardened, as they should be, against this. Microsoft maybe will be a little less glib about the fact that these memory remanence attacks are no problem for BitLocker when in fact these guys last week demonstrated that they in fact are today by pulling off exactly this attack. What it probably means is that, until then, or for existing systems, if you're going to have systems which sleep, that is, that do this, you need to maintain physical security. This is a physical security attack. Not possible to perform this without digging into the machine itself. But you can imagine, if law enforcement were to grab a machine that were sleeping, they would now keep it powered up and take it back to the lab and

probably be able to gain full access to it. I mean, sleeping is dangerous, even if you're prompted for a password when you open the screen.

So as with the recent TPM bypass that we talked about, the culprit is this sleep state. Remember that the TPM specification itself had an error in the way sleep is managed with the Trusted Platform Module. So if it's possible, as I suggested for anyone concerned about this, disable the sleep option in the BIOS. Most BIOSes today still let you disable sleep so that it's just not available. But the other thing you can do, at least in Windows, is you can use the so-called "Group Policy Editor." And I have on the page in the show notes here, I've detailed it completely. You type at the Run bar gpedit.msc, which brings up the Group Policy Editor. And I won't go into details because it is here in the show notes for anyone who's interested.

Using Group Policy Editor you're able to disallow Windows to use the standby states S1 through S3 when sleeping, both on battery or when plugged in. You're able to disable that. And it simply removes the option from the menus. It will no longer sleep. And you are then able, for example, to enable hibernation if you would prefer to do that. Hibernation still makes me a little uncomfortable, but at least it's not going to be - you're not going to have the data in memory, and presumably Windows is a lot safer coming out of hibernation than it is waking up from sleep because sleep has a running instance of everything sitting there in RAM.

So if anyone is really concerned, if their system is sleeping, and they believe there's some possibility that a system that is sleeping could be actively physically attacked, then disabling sleeping certainly makes sense. And of course you are able to more easily do that in a UI just by disabling the automatic move to sleep and instead have the system hibernate or just shut itself down.

JASON: Sometimes I come on this show, and it makes me realize I have bad habits because I do put my computer to sleep. But I do that actively. So it's like I'm making that choice, and that doesn't make it any better.

Steve: Yeah. And it certainly is a convenience to have it just wake up out of sleep. So, yeah.

JASON: Sure.

Steve: And I would argue, too, that it increases the system's longevity because you're turning off, you're powering down the processor and all of the peripherals and only keeping the RAM alive. So it minimizes the heat in the system.

JASON: Sure.

Steve: Okay. So as they say, don't try this at home. I couldn't resist, mostly because I had a laptop that had not yet been updated to v12. And I thought, okay, if they're going to fix this after - I don't mean a laptop, an iPad. If they're going to fix this in v12, which just became available, I want to know that it wasn't fixed in 11 point whatever I had before. So this is known as Safari Ripper. It is an Apple Safari vulnerability that quickly crashes iOS and Mac devices, where links are handled by Safari. And it works.

Sure enough, I clicked the link, pow. It just completely collapsed the iPad. Then I updated to v12, did it again, whoops, screen went black, and it rebooted and had to come back to life. So Apple has a problem that they did not fix in the most recent update to iOS. It exists before, and it exists today. It leverages a relatively new and processor-intensive feature of CSS, you know, Cascading Style Sheets, which is known as the "backdrop filter." That uses 3D acceleration to process the image background underlying CSS elements.

The backdrop filter is able to blur or color shift regions behind an element. And you can imagine fancy, I mean, we see it on web pages often, fancy pages where you move over something, or you scroll up, and something that you're sort of not supposed to be paying attention to blurs. Or you have a full-color picture, and it goes to black and white. Or it goes to, sort of like a chroma keyed effect. You can do all kinds of things with it. But this is actually happening, not with code, but, well, with code driven by static CSS elements and, in this case, this implementation of a backdrop filter. So the point is it's image processing happening merely by a declaration of a line of CSS.

So security researcher Sabri Haddouche was poking around while looking for reliable denial of service bugs against various browsers when he discovered this, a simple way to essentially bring any Safari-based system to its knees. All he used was nested divs. The "div" is short for "division." And so you can have a division in a division in a division in a division. And in fact I looked at his code, and he's got, I didn't count them, but he's, like, hundreds.

And the problem is, if you were to apply the backdrop filter to the CSS for the division, then you're going to cascade the need to perform this rendering, that is, essentially you're layering hundreds of these filters on top of each other. And so the result that you're going to see on the top layer is the result of its filtering through the CSS for it, which it doesn't get until it knows what the CSS behind it is going to resolve to, which it doesn't know until the CSS behind it is going to resolve to, which you're able to force with this crazy stacked set of divs in CSS, in the HTML.

So it turns out, boy, is it potent. For anyone who's interested, I have the link in the show notes. You might be able just to google "Safari Ripper" right now and find it. It was not easy for me to discover it. I mean, not difficult for me to discover it. It was easy. And, I mean, it doesn't do any harm, but it does immediately crash your Safari-based rendering, whether on iOS or Mac. And I wouldn't be surprised if the link ends up getting retweeted and so forth, people playing games with this. As I mentioned, v12 doesn't fix this. So I would imagine that it won't be long before we get a 12.0.0.1 that does somehow arrange to resolve this.

JASON: And that only works in the Safari browser? Is that right? We've got some folks in the chatroom who are saying 403 Forbidden on that link. I know I tried it earlier, but I wasn't on a Safari browser, and I got that 403 Forbidden link. But does that mean that they've bottled it up? Or is that [crosstalk]?

Steve: Okay. I just tried it, and it came up. Although it came up on, let's see, it came up on Firefox, which is my default renderer on Windows. And it looks just fine. And it says "Triggered," which is I know what the image is supposed to look like.

JASON: Yeah, I get a Forbidden, yeah. Strange.

Steve: Interesting.

JASON: Get a 403, yeah.

Steve: Anyway, so maybe it's been taken down. It's certainly on the 'Net and around. And in fact I've got it on my browser. So who knows?

JASON: Interesting.

Steve: Okay. So as I mentioned at the top of the show, Twitter has permitted apps and devices management, which makes them late to the game, but better late than never. In Twitter, if you go to Settings and Privacy, under there you'll see Apps and Devices. You

can also just go to the URL Twitter.com/settings/sessions, which will take you to the same page. First, at the top of the page, I did it yesterday, they're promoting connecting to Facebook. No, thank you. But then there's a section of "Apps connected to your Twitter account," followed by, and this is what's new, "Recently used devices to access Twitter," which is nice to see.

And we've seen this with other, you know, Google has been doing this for a while. We've talked about Facebook and the Facebook API. Broadly, these various services have APIs that allows us to give applications or other services access under our permission. Very much like the browser extensions - or the browser barnacles - that we'll be talking about at the end, these sort of tend to just accrue over time. You're doing something. Something says, hey, I need access to Facebook, or I need access to your Twitter account. And you go, okay, fine. And so you use this new widget for maybe five minutes, and then it's like, okay, and you get distracted. You see something else that's shiny, and off you go. But that thing still has access because they don't ever give it up as a consequence of disuse.

It's also very useful in the case of something where a lot of you is present, like on Google. It's worth looking, auditing where the devices have been that have logged into your Google account. So that we didn't have before now. Now we do with Twitter. So on the first hand, in the case of apps, it's always useful from time to time to just scrape the barnacles. Just go through the list. And I did yesterday. And it's like, oh. There were, like, maybe half of the things that I had given permission to previously I was no longer using. And so they no longer have permission. They're gone. And then the second half is where have logins occurred recently? And maybe it's a little buggy, or maybe it's just because it's new. But I had several devices showing that I had last logged onto Twitter on New Year's Eve 1969.

JASON: At 4:00 p.m. I've got one on my list.

Steve: No kidding.

JASON: Who knows what it is? It's an Android [crosstalk].

Steve: Were you alive, Jason, in 1969? I guess you were.

JASON: Not that I know of. I don't believe that I was. But I can neither confirm nor deny.

Steve: Okay. In that case, I can tell you that in '69 I was, what, I was a freshman in high school, and I had not yet encountered the PDP-8 from DEC which would change my life as I learned to program in its machine language and its assembly language. But there was definitely no Internet, and there was definitely no Twitter. And so this was definitely a mistake on when did your app last use Twitter. Okay. But anyway, definitely worth, again, auditing these things, whether they be, in these barnacles, whether they be permissions that we've given to services or extensions that we've installed in our browsers over time. Definitely worth doing for the sake of privacy and security from time to time.

JASON: For sure.

Steve: Just a quickie note that the Mirai botnet is on track to become historic. Things like Code Red and Nimda are worms that we could argue will go down in history. WannaCry and, you know, various things have made a big impact over time and have ended up carving out a little bit of a niche for themselves. I think the Mirai botnet is probably on track. In its latest evolution, its repertoire of exploits has grown to 16. And while the

majority remain aimed at compromising routers, networks, video recorders, and DVRs, Mirai is also more recently and now targeting unpatched Apache Struts vulnerabilities.

So, I mean, the people behind this are demonstrating an intent to jump on anything that they can go after. We talked last week or the week before about the fact that there were - we were now seeing botnets which were using a version of Linux which had multiple processor hardware targets in order to significantly broaden the hardware platforms they were able to infect and occupy. And so when you combine that with a growing vocabulary of exploits, I mean, known exploits which are patched from their manufacturers, but not patched on the endpoint devices, we're ending up looking at a botnet which is growing in size and is becoming incredibly powerful. And we will talk about, when we talk about the explosion of DDoS next, just how much these things are becoming a threat.

Okay. So explosion in DDoS network sizes. There's a company, Nexusguard, N-E-X-U-S-G-U-A-R-D dotcom, that's in a business I want no part of, which is DDoS mitigation. It is just - it's not a fun job to be in the DDoS mitigation business. If you're doing it by trying to filter bandwidth, then you're tackling staggering sizes of flooding attacks these days. And it just seems to me it's sort of a thankless job. Anyway, they're in the business, and so they have the ability to generate statistics that we don't see about what's going on in the DDoS world. In their most recent report, they have compared the first two quarters of last year versus the first two quarters of this year.

So during these identical periods, they have seen a 29% upwards jump in the number of attacks and, get this, a staggering 543% increase in the average size, that is, the bandwidth flood size of the attack, where the new average DDoS attack is at 26.37 Gbps of attacking bandwidth - 26 Gbps. Which says, you know, if you have a gig connection to the Internet, it will be flooded. The point is that it will be saturated with attack traffic such that no valid traffic has the opportunity to get into your servers. You will be knocked off the 'Net.

So this Nexusguard report shows the average size of attacks in 2017 during this period was 4.1 Gb, with a maximum observed of 63.7 Gb. One year later, the average size has grown by more than five times to 26 Gb. That's average; right? That's 26.37 Gb. The maximum size attack they have seen, 359 Gb. So again, as I said, being the company trying to keep an entity online that is being hit with 359 Gb, good luck.

And of course the attacking sources will come as no surprise to followers of Security Now!. Nexusguard in their report wrote: "The increase in attacks and their sizes is attributed to attackers amassing giant botnets using insecure IoT devices. Attackers are using vulnerabilities in these devices to rapidly build large botnets that can then be used to perform targeted attacks that are increasingly difficult to stop. For example, at one point the Mirai Satori botnet was seen from over 280,000 IP addresses over a 12-hour period, and the newer Anarchy botnet was able to amass over 18,000 routers in a single day. These botnets were created by attackers exploiting vulnerabilities in routers such as ones made by Huawei and D-Link."

And of course we've talked about these before. They are sitting on ever larger bandwidth WAN links, thus allowing them to attack targets with ever larger traffic. And 280,000 machines in a single botnet is just staggering.

Nexusguard wrote: "In addition, severe botnet epidemics like last year's Satori continued to threaten cyberspace by exploiting zero-day vulnerabilities. Since its high-profile attack on Huawei home routers in December 2017, Satori has wreaked havoc over the past few months on various IoT devices, including GPON-capable routers" - those are the ones that we've talked about often in Brazil - "manufactured by South Korea's Dasan, D-Link's DIR-620 routers" - that we've talked about - "and the uc-httpd IoT devices. Additionally,

the quarter saw the emergence of the Anarchy botnet, which exploited zero-day vulnerabilities in a similar fashion as Satori."

The report also said that slightly more than half of attacks, 55.28%, had a duration under an hour and a half. So slightly more than half of the attacks had a duration of less than 90 minutes, but the average attack lasted 318 minutes long. And that average was pushed up because some attacks lasted for days, with the longest being six days, five hours, and 22 minutes. And then they also finally said, although nearly two-thirds of attacks, 64.13%, were under 10 Gbps, the average size, as we mentioned, was 26.37 Gb. Oh, and lastly, the United States was the largest source of attacks at 20%, followed by China, France, Germany, and Russia successively, sourcing smaller levels of attack. So we are in a world where DDoS happens with increasing frequency and with literally unstoppable force.

JASON: I kind of feel like that should be a T-shirt: DDoS Happens. Are we at the errata?

Steve: Not quite.

JASON: The strange Windows 10 popup, the screenshot.

Steve: Well, okay. So Windows 10 is starting to warn - actually, yes, warn - users who attempt to install another browser. Sean Hoffman tweeted this. His tweet was - he tweeted it to @MicrosoftEdge. "What kind of slimy marketing cesspool crap is this, Microsoft? I proceed to launch the Firefox installer, and Windows 10 pops this up. If I wanted to use your browser, I would."

And he quotes, and I found this elsewhere also, so his tweet includes a screenshot of a happy person with the Edge "e" behind him in a window. And this dialog from Windows 10 reads: "You already have Microsoft Edge - the safer, faster browser for Windows 10." Okay. And then there's two options: Open Microsoft Edge or Install Anyway. And then underneath it says: "Don't want to be warned in the future? Open settings." And I didn't look, and I haven't had this pop up. This is on the Nightly build, or the Windows 10 Insider at this point. But what is the settings going to say? Don't warn when I attempt to install a non-Microsoft browser? Is that what it's going to say? Anyway...

JASON: Just give me a checkbox, a checkbox on that thing that says don't do this again. It reminds me of when you get subscribed to something like an email thing, and then you go down to the bottom, and instead of the Click Here to Unsubscribe it's Click Here, and then that takes you to another page where you have to, like, work in order to unsubscribe. It's like, just do the thing.

Steve: Well, and that's it. It says: "Don't want to be warned in the future, open settings." So what is the setting going to read that you turn off to disable being warned when you're trying to install a browser that Microsoft didn't provide? I mean, I guess I get it that maybe somebody who would have Windows 10 would not click on a link where Edge immediately comes up and would instead think, oh, first thing I need to do is install somebody else's browser. But come on, really? We know that's not the case. They are deliberately creating some pushback, some back pressure against anybody who wants to install a different browser.

It turns out that this also applies to Firefox, Chrome, Vivaldi, and Opera; that it's in Windows 10 Insider Builds only. And, yeah, I know I'm annoyed when I'm going to Google's search page, and I get the little box in the upper right. Google's doing the same thing, you know, promoting their own Chrome browser when I'm using Google Webs without a Google Chrome browser. But that's what I prefer to do. At least Google lets me ignore it, and I can just sort of go about my business. Here you've got to say, yes, I'm

sure, I really do want to install the browser that I clicked on to try to install a minute ago. So anyway, grumble.

JASON: Grumble, grumble.

Steve: I have a bit of errata from Alexey, I'm not kidding, Alexey in Russia. And if Leo were here, he would probably give us his Russian accent reading. I will spare our listeners that in this case.

JASON: I will not follow in his footsteps on the Russian.

Steve: Thank you.

JASON: You're welcome.

Steve: The subject was "Couple of thoughts in defense of MikroTik," he says, parens, "(and they *can* auto-update)." He says: "Hi, Steve and Leo. Long-time listener, SpinRite licensee with all the customary blah-blah-blahs to you both. I've been using MikroTik (MT for short) routers since about 2012. I have a couple at home and one at work. I'm a sysadmin in a small company. I'd like to mention that my oldest router still receives all the updates, generally one or two a month. It's a very strong point for MikroTik in my book. I've heard about Ubiquiti long after becoming acquainted with MikroTik, and never had reason to switch nor spare cash to experiment, despite your fascination with the infamous Ubiquiti Edge.

"First of all, I'd like to mention that MT routers can be easily configured to auto-update. Details are below, if you're interested. Secondly, I do agree with you MT have to change their policy and default settings. And they have the capabilities to do so without additional hassle for anyone. They have several templates for users to choose from on initial setup. They just have to harden their home router template and make it the default for new devices. They don't. But if they did that, users would be reasonably protected, and advanced ones will configure from scratch or choose another template.

"However, in their defense, I have to say that, A, it's absolutely clear from even a cursory look that their routers are not consumer-oriented plug-and-pray or plug-and-forget devices; and, B, that they're targeted, not at casual home users, but at geeks at the very least." He says: "I'll leave out the professionals and such here. In my experience," he says, "configuring MT router is a bit more like configuring Linux server than configuring a traditional consumer router. Common good practice to use non-standard ports for services, including SSH, Web UI and Winbox if you use them, fully applies here, and to disable ones you don't use, as well. And you absolutely have to configure your firewall. Default configuration is far too basic for deployment.

"There's no checkbox to allow/disallow WAN access. We're grownups here. You have to configure firewall rules - with netfilter/iptables-like CLI commands or in a GUI - to allow or disallow such connections. And, yes, with great power comes great responsibility. MikroTik routers can easily be misconfigured. I have to say all recently reported vulnerabilities you told us about were mitigated by properly configured firewall, i.e., allow inputs only from trusted IPs in LAN or maybe WAN if you really need it. Drop all other inputs. And changing default service ports helps a lot, too."

And then finally he says: "Regarding auto-update. As you've said in your coverage, RouterOS, the OS of MT's routers, have extensive scripting capabilities and cron-like scheduler. Script to check for update, install it automatically, and reboot is about four or five lines long. Couple more if you want to be notified by email before or instead of updating. Schedule it to run daily at 3:00 or 4:00 o'clock in the morning, and you're good to go. Judging from MT forum posts, sysadmins and ISP guys don't think their routers

should auto-update. They prefer to test new releases beforehand and only deploy security fixes, let's say. But for home users who don't want to do it manually, it's there. And users don't need to buy a new router. Their old ones are perfectly capable of doing so. Me personally, I prefer to do my updates manually. And now I have an automatic update available notification."

So that's what Alexey has to say. I will note that, first of all, I first learned of MikroTik when people were comparing it to the $49 Ubiquiti EdgeRouter X. And so I went over to MikroTik.com, and I found, for example, the hAP ac, which they say is our most universal home or wireless device. Dual band, 3x3 MIMO with gigabit ports that opens the full advantages of 802.11ac speed while maintaining compatibility with legacy devices in 2 and 5 GHz. So there's the hEX Lite at $39.95, the hEX at $59.95, hEX PoE Lite at $59.95, the hEX S at $69.00, and so forth. Now, yes, they absolutely have enterprise-oriented products because, for example, the CCR1072 1G 8S comes in at $3,050, a 1U rack-mount device with lots of features. And they clearly have a sense of humor because sort of in the middle is the RB1100AH Dude Edition, which comes in at $349. So a range of routers.

I understand and respect what Alexey said. But I disagree. I mean, I understand what he said. But, for example, if there is a template, and a home user can say give me the home router template, and that doesn't protect them from incoming service access from the WAN, which is what Alexey just said, that is, the template, the default home router template does not provide protection from a user who does not want to set up iptables cron scripts, then MikroTik is doing it wrong. Given the numbers that we have seen, those are not enterprise - those are not $3,000 routers. Those are $39.95 routers, and they've been set up as home devices by people who assume that means they're done. And what Alexey has said is the default template for the home user does not protect them the way they would expect to be protected. So I get it that it is a cool, feature-packed, powerful router. But they've got their defaults set wrong, and they should fix that.

And I'm on the fence about updates. I get it that there are at the high-end level the enterprise guys are saying, we don't want our routers auto-updating. We want to take responsibility. But if you're, I mean, especially if you're going to default to having service ports for sensitive services by default exposed to the WAN by design, as I said, that's a policy problem. That's not a bug. I just think it's unconscionable in this day and age to do that. And so there's no way to let them out of the hot seat. I get it that they have a lot of power in their devices. People like them for that reason. But if you select "I'm a dummy, please protect me," and then it doesn't, that's wrong.

Also, in Miscellany, Jeff Root in San Diego, California, he said of our last podcast: "OpenVPN Running with Privilege?" He said: "All the reporting I saw on this bug said it was exploitable on Windows only." And I stand corrected. I should have mentioned that. He said: "Checking my Linux system, I see that OpenVPN is installed," he says, "by default on Debian, at least, so that it runs as user 'nobody' and group 'nogroup.' The config file for OpenVPN clearly states that it will drop privileges after initialization except on Windows. As far as I'm concerned, the question that should be asked is: Why can't it drop privileges on Windows? Is that not supported by the Windows API? Which should, actually, be the fix for this anyway. This violates the fundamental security principle of 'least privilege,' which has been known and implemented since the '60s."

And I will say that having built a custom installer for my SQRL client, I learned exactly what Windows can and cannot do. And in fact it cannot, there is no privilege de-escalation availability or capability in Windows. So in fact there is no way, I mean, I'm sure you can come up with some rigamarole where you then run with some sort of limited special user. Unfortunately, they're running with system privileges, rather than creating a special user that does exactly what it is they want.

So I guess more could be done. But it is the case that you are unable to give up privileges once you're running. And it makes for some interesting gyrations when you're running an installer that needs to install as a limited user, but install in places where you need to be an admin in order to do so. But those problems can be solved.

An interesting instance of SpinRite repairing an SSD. As we've said, I don't know what I'm going to in the future with the name because I really like SpinRite, but it fixes things that don't spin. And somehow PlugRite doesn't really work for me. I've not figured out what I want to do. But Justin in Olympia...

JASON: AllRite.

Steve: AllRite. Ooh.

JASON: AllRite. See, it's kind of...

Steve: Oh, Jason. That's pretty good, yeah. Justin in Olympia, Washington, his subject was "SpinRite Catches the Bad Guy." He sent this on September 16th. He said: "Steve, long-time Security Now! listener, blah blah blah. Wanted to let you know how SpinRite helped to catch a criminal. Granted, it was in a roundabout way. But still it probably would not have happened without SpinRite. My security camera server was having issues. The normal rock-solid software started crashing on a frequent basis. Every time it happened, until I happened to notice it and was able to reboot the server, my system was down. It was really getting annoying."

So the point he's making is it was crashing frequently, and therefore being offline until he happened to note that it was crashed, then he would reboot, and it would be up for a while until it crashed again. So it's spending the bulk of its time down, apparently.

He said: "I'd tried a lot of things to fix it, but I was getting the sneaking suspicion that the SSD startup drive might be the problem. Earlier this week, I ran SpinRite on Level 2. While it did not report any bad sectors, it did the trick anyway. The camera software has not crashed since. This is a very good thing, as today someone tried to steal some things I had in front of my house. I was *praying,*" he has in asterisks, "that the camera software had not crashed and was thus able to record the attempted theft. And thanks to SpinRite, the entire incident was recorded.

"I was able to provide the clips of the guy to our Sheriff's Office, and within an hour he was in custody. Oh, yeah, he fought with a deputy, and when he was being arrested got introduced to the business end of the K-9 unit. He also had several felony warrants outstanding - a real upstanding member of society. Thanks to SpinRite, a really bad dude is behind bars. Thanks again." And, yikes. Thank you, Justin.

JASON: That's awesome.

Steve: For sharing that. And, yes, a very cool story.

JASON: Yeah.

Steve: Okay. So we wrap up by talking about the web browser extension ecosystem. And, boy, you know, I'm going to stick with barnacles. I think that's, you know, that would really - we had a problem because the text, the web browser extension ecosystem wouldn't fit all on the lower third line for the podcast.

JASON: Yup.

Steve: And so it could have been just Web Barnacles. That would have been good, actually.

JASON: That works for me, Web Browser Barnacles.

Steve: Web browser, yeah. So as I mentioned at the top of the show, I was taking a closer look at the current state of our web browser extension ecosystem, driven by the news that two major versions from now - right now we're at Firefox 62. In Firefox 64 it will be adding a welcome new feature to the UI. Right now in the popup menu, the dropdown menu, from its web browser extensions icons, there is a list of commands such as Manage the Extension, Remove the Icon from the Toolbar, Pin to the Overflow Menu. And then there are also, lower than that, some non-extension-related display options to Hide or Display the Menu Bar and Hide or Display the Bookmarks Bar, and also to further customize the browser's UI.

Firefox 64 will add a prominent, actually at the top of the list, Remove Extension - or scrape barnacle - at the top of the popup menu list. The team at Mozilla wants to make the process of removing unwanted extensions easier and very, very clear. And at some point in the future, although apparently not yet with Firefox 64, they further plan to build in an option to submit an abuse report, on the spot, to solicit feedback about the reason for an extension's removal.

So this news, along with our recent coverage of malicious extensions creeping into Chrome and the many times I have encountered weird and apparently unwanted browser toolbars installed in other people's machines led me to look a little bit into the current browser ecosystem for web browser extensions. And that led me to this stunning graphic. Oh, my god. I'm just, well, and Jason, you're at the top of the peak, baby.

JASON: I'm right there in the middle at the very, very top, absolutely.

Steve: You are representative of the majority of users. So to describe the shape of the curve, for those who can't see it, it's a histogram where the bar height is the number of people with that many extensions installed. So it looks like people with one extension is just a little less than halfway between 5,000 and 10,000 browsers in the sample size. And I'm trying to - I had it here in the show notes somewhere. Oh, okay.

So they examined 900,000 PCs, okay, so .9 million. They looked at a big cross-section of PCs. So it looks like about a little shy of 7,500 users had one extension. Looks like it goes by single units. So then it looks like maybe 8,000 had two extensions. Almost 10,000 had three extensions. Maybe about 14,000 had five extensions. 18,000 had six. Maybe 22,000 had seven. More than 25,000 users had eight. And we keep going up till we get to about 14 or 15. Anyway, so 12, 13, 14, 15, 16, 17 numbers of extensions were all above 35,000 users.

So essentially what we're saying is that many, I mean, okay, and, okay, so that's the shape on the leading edge of the histogram. The trailing edge is more slopey, meaning that the number of extensions keeps growing, although the number of users with that many extensions is falling off more slowly than on the leading edge such that more than 5,000 machines had 41 extensions. More than about 15,000 systems had 30. And believe it or not, there's a non-zero blip out here at looks like 75 extensions installed in some non-zero number of browsers. And actually there's tiny little blips all the way out to 90. Out to a hundred. Again, so, okay, holy crap.

JASON: You know, but they use every single one of them very regularly. Every single one gets used on a daily - no, probably not.

Steve: Just like a good barnacle.

JASON: Yes.

Steve: One of the reasons I love the analogy is that they do slow the boat down. The reason you have to bring the boat into dry dock every so often and scrape the barnacles off the hull is they represent substantial resistance to the water flowing in what's supposed to be a laminar flow across the hull. They mess that up. And in the same way barnacles, browser barnacles, slow down your browser. Because they need to be loaded. They need to be initialized. And they're doing something with your traffic.

So what are they doing with your traffic? I found, okay, this chart came from a study that was conducted titled "Quantifying the Web Browser Ecosystem." It's a research paper from the summer of last year, 2017. And I'll just share the abstract and a bit of the introduction.

They said in the abstract: "Contrary to the assumption that web browsers are designed to support the user, an examination of 900,000 distinct PCs shows that web browsers comprise a complex ecosystem with millions of add-ons collaborating and competing with each other. It is possible for add-ons to 'sneak in' through third-party installations" - as we've all experienced where you're installing something that wants to bring along some toolbar, thank you anyway - "or to get 'kicked out' by their competitors without the user's involvement. This study examines that ecosystem quantitatively by constructing a large-scale graph with nodes corresponding to users, add-ons, and words, the terms that self-describe add-on functionality.

"Analyzing add-on interactions at user level using the Personalized PageRank random walk measure shows that the graph demonstrates ecological resilience. Adapting the PPR [Personalized PageRank] model to analyzing the browser ecosystem at the level of add-on manufacturer, the study shows that some add-on companies are in symbiosis, whereas others clash with each other as shown by analyzing the behavior of 18 prominent add-on manufacturers. Results may herald insight on how other evolving Internet ecosystems may behave, and suggest a methodology for measuring this behavior. Specifically, applying such methodology could transform the add-on market."

Okay. So they're getting into the world of sort of statistical research. But they did have some interesting things to say in their introduction. They said: "Web browsers have become a major component of the routine human-computer interaction" - yeah, no kidding - "with some browser operating systems" - and Jason, you have it in front of you - "some browser operating systems based entirely on browsers, e.g., ChromeOS by Google." And there it is.

"Browser extensions, also known as add-ons" - and now also known as barnacles, okay, they didn't say that, I did, just to be clear - "are computer programs that, as the name suggests, extend, improve, and personalize browser capabilities." Okay. Hold your breath. "More than 750 million add-ons were downloaded and installed by Google Chrome browser users as of June [not recently] 2012." So that, lord knows, in the last six years that's only gone up.

"Some examples of add-ons include an extension that allows visually impaired users to access the content of bar charts on the web." Hey, that's very cool. "An extension that addresses user security concerns by seamlessly producing a unique password for each website the user accesses." And of course we've talked about Ghostery. NoScript was one I was promoting for a long time until you just couldn't use the web without having scripting enabled. And then more recently Gorhill's uBlock Origin is one that I promote. And we've talked about iOS allowing the content filtering, and that immediate upsurge in adblocking that then became available to Safari on iOS.

"Internet software companies," they write, "are very interested in installing their add-ons, and particularly toolbars, on users' machines." And of course that pressure is why end users end up with these stacked toolbars that provide hardly any room to actually see the web page. They write: "Toolbars are GUI widgets that typically reside in the upper part of the browser's window, extending the browser's functionality. Toolbars can collect information about the browsing history of the user, for example Yahoo Toolbar, and can redirect user search activity to a specific search portal, for example, MyWebSearch.com. Crucially, the company that owns the search portal, and typically also the toolbar, receives payments from ad providers per user click on the ads it displays." And then they say, "Primary ad providers are Google and Yahoo. This revenue generation model is used extensively by software companies that distribute free products."

Now, get a load of this. There's a stat I did not know. "For example, 45% of AVG's Antivirus Technologies sales in 2012 were generated by its browser toolbar." Okay. Nearly half, 45%, of AVG's AV sales came from the installation of its toolbar. "It was estimated that Google, the biggest web advertising firm, might have lost $1.3 billion in revenue in 2013 because of changes to its policy with respect to toolbars and a resulting shift of some add-on distributors to Google's competitors." In other words, there's gold in them thar barnacles.

"Consequently, add-ons compete with each other over resources such as battery, memory, disk space, and computing power." And of course we know that now browser coin mining is a thing. And so, yes, competing for computing power and user attention. "Regardless of how intelligent they are, they may be aware of each other and may piggyback on each other, or uninstall each other. Add-on behavior within the web browser is characterized by add-ons making their own decisions independently and often unbeknown to the user, which comprises a complex ecosystem with the user being just one of the participants." In other words, we're not in control. "A key issue in understanding that ecosystem, responding to it, regulating it, and transforming it into a mature market is the current inability to show that it is inherently stable and measurable. This study addresses that issue."

Okay. So I think our takeaway from this is the recognition that the end user and also apparently the browser's vendor are less in control over this browser extension world that has evolved than we and they may think, and that those browser extensions we may have installed some time ago and then promptly forgotten are still alive and installed and having access to everything we do on the web - which, by the way, is unencrypted to them, and also slowing down every launch of the browser.

So I welcome tools, which I mentioned in Firefox 64 we're going to get. We're going to make it easier for people to just right-click and remove something from Firefox. And this is another of those things that is absolutely, I would argue, you know, sailors, as I said, pull the boat out of the water and scrape the barnacles. We as browser users should take a check from time to time into our browser's add-on list and scrape those barnacles off the browser because they're slowing it down.

So, as I said, I welcome tools which profile our extensions and alert us when an extension is slowing down the browser's launch on page loads. And I'll note that I've seen Edge do that. It'll show me when a certain extension is representing a problem. And it's like, ooh, is it worth it to me to have it slowing me down? And anything that our browsers can do to provide us with additional decision-making tools and empowerment, I think we should all welcome.

The problem, of course, is that we're savvy system users, and web browsers are being used predominantly by people who don't know what they need and don't need. These are the people who for years have been downloading updated versions of malicious Flash

players because the web page they visited told them that they needed to do so. By comparison, our operating systems have pretty much finally locked down the management of what gets installed into our machines and when. But the browser extension ecosystem really hasn't yet matured to that level.

So anyway, I just think that in the same way that it's good to occasionally curate the apps that Google, Facebook, and Twitter now have been given permission to access our accounts against, it's worth taking an occasional survey of what extensions our browsers have accumulated and deciding whether we're still needing them or using them or not. So anyway, I was stunned by the graphic of the number of extensions in browsers. And when hearing, for example, about one to aid a visually impaired person to see something like a pie chart that they wouldn't otherwise be able to appreciate, I think that's very cool, and clearly super useful.

The other problem is the whole thing is about reputation. Mozilla and Chrome and Microsoft have earned, have hard won reputation. We believe that the browsers themselves are operating in our best interests because it is in their best interests for them to do so. The problem with extension makers is we don't, for the most part, we don't know who they are. I mean, I know LastPass, and I've got the LastPass extension installed in all my browsers because I want the services it offers.

But, boy, you look at some of these extensions that just come from an unknown author offering something that seems at the moment like something we want, but this person has no reputation. How do you make a decision to install this in your computer? I mean, the browser is in a position to see a huge percentage of what we do. And what I think users don't appreciate is this is global. Extensions have global visibility into what we're doing, not only when we call them up or only when we use them, but they're sitting there in the catbird seat, loving the fact that they're able to see everything going on everywhere in the browser. And I really think we just do have at this point a very immature system for dealing with this whole issue.

JASON: Well, it ends up being out of sight, out of mind, too. Like in order to actually see these things running, like sure you can look up in the corner of your browser, and you might see those icons. After a while that almost becomes something that you're so used to seeing that you don't really scrutinize it to any degree. In order to actually see your entire list, you have to go into the settings. And how often do you go into the settings and check your extensions? I've been scraping the barnacles while we're been having this conversation here. What I'm realizing when I look at it is about half of those extensions are actually Google extensions - Google Hangouts, or Google has this password alert extension that protects you against noticing phishing attacks, Google Docs offline. So some of these are Google.

Steve: Right.

JASON: And when you're talking about trust...

Steve: I agree.

JASON: You know, Google already knows everything about me. So I would leave that on there.

Steve: Well, it is their browser that we're adding the extension to. And I like that. There I would consider Chrome to be modular in that you're able to add Google modules to Google. They certainly know how to make a module that won't upset the browser and hurt the user's experience and so forth.

JASON: Hope so.

Steve: And so I like the idea of, for example, I don't have to have those Google things added if I don't want them. You're able to. And so you've extended Chrome. And there's the perfect example of the fact that Google has a reputation. I wouldn't hesitate, if I wanted those features, to add them to Chrome because I know where they're coming from. But, boy, you look at some of the shady characters that are, I mean, we just don't know who they are.

JASON: Right.

Steve: And it's like, yeah, come on in and watch everything I do with my browser. No.

JASON: Well, and I think what you said earlier about impulse, too. Like so often over the years we've become accustomed to the fact that there is literally an app or an extension or a website or anything that will appeal to that momentary impulse that we have to need something very specific. And at that moment in time it seems like it's so important. And you find the extension, like, oh, yeah, sure, I'll add that. And then you forget about it because it was useful then. And then months or years down the line, like you just don't even know that it's there anymore. And it's still, like you say, surveying the landscape. So it's important to go in there and clean it up.

Steve: I have so much crap on my iPhone, I look at it, and I just, it's like, okay, someday I'll need to deal with this. But, I mean, I can't think of a better example than, you know, the barnacles that I have on my iPhone and my iPad. But it's like, okay, fine, I still have lots of room for more barnacles.

JASON: I mean, this experience matches up to going into, like you were saying, social media accounts, going into Twitter and taking a look at all the accounts that you've connected through Twitter. I did this with Facebook. Like I hardly ever use Facebook anymore, but probably like a half year ago I went there and checked the apps connected to Facebook list. And, I mean, it was huge. It was like a hundred different things dating all the way back to like 2008 - apps, you know, services that don't even exist anymore, yet they're still connected somehow. And so I just - I basically just crossed my fingers and said disconnect everything, even the things that I think I might need now. Like, whatever, I'll figure it out. Just get rid of it all.

Steve: Right. If something breaks, then you can put it back.

JASON: Exactly. And you know what? I've been fine. I haven't even run into a single speed bump, so I'm happy I did it.

Steve: Yup.

JASON: Steve Gibson, this was a lot of fun. We've reached the end of Security Now! for this week.

Steve: And for you.

JASON: And for me, until the next time Leo gallivants, as I like to say, around the world, as he is wont to do and as he deserves to do, take vacations from here from time to time. And I would be happy to sit in in his place when he does so next time.

Steve: I know I can speak for our listeners when I say we will be happy to have you back, Jason.

JASON: Thank you, Steve. GRC.com if you want to check out everything that Steve's been up to. Of course he talked a lot about SpinRite today. It is the best hard drive recovery and maintenance tool, so you can find your copy there. Also information about SQRL. Audio and video of this show. The only place you can go to find transcripts of this show can be found there, as well, GRC.com.

And then of course our website is TWiT.tv/sn for Security Now!. You can find our audio and video there. Podcast links if you want to subscribe, if you haven't already. I'm sure you're already subscribed. But if you haven't, go there and subscribe. Everything you need to know is there. And the show records live every Tuesday, 1:30 p.m. Pacific, 4:30 p.m. Eastern, 20:30 UTC. And if you want to watch it live, you can: TWiT.tv/live. Like I said, if you're subscribed, you'll get it automatically each and every week. And maybe just do both. Then you're covered. Thank you, Steve. Really appreciate it.

Steve: A pleasure, Jason. Until we meet again.

JASON: That's right. Until next time. And we'll see you all next week, Leo will see you next week with Steve on another episode of Security Now!. Bye, everybody.

Steve: Bye.