Gem Hack & Ghost Protocol

Description: This week we stumble over a number of instances where technology appears to be colliding with the status quo. In any complex social system, individual and group interests are often complex and may be in opposition. So when new technology comes along to offer new capabilities, not everyone is going to be pleased. So this week we discuss some of the mounting tensions being created by connectivity, storage, and computation which are being combined to create many new capabilities. We look at the surprising backlash to Mozilla's privacy-enhancing DNS-over-HTTPS support, concerns over the use of facial recognition and automobile license plate scanners, and the future of satellite-based Internet services. We present some SQRL news and share a bunch of closing-the-loop feedback from our listeners. We then examine how a Ruby code repository was hacked and look at the U.K. GCHQ's proposal for adding "ghost" participants into private conversations.

High quality (64 kbps) mp3 audio file URL: http://media.GRC.com/sn/SN-722.mp3
Quarter size (16 kbps) mp3 audio file URL: http://media.GRC.com/sn/sn-722-lq.mp3

SHOW TEASE: It's time for Security Now!. Steve Gibson is here. We're going to talk about DoH. It's not what you think. It's DNS-over-HTTPS. And it's why you might want to look at your router and see if you could do it. We'll also talk about satellites, satellites all over the place, maybe too many satellites. And a Gem hack that any Ruby user is going to want to know about. This is all coming up next on Security Now!.

Leo Laporte: This is Security Now! with Steve Gibson, Episode 722, recorded Tuesday, July 9th, 2019: The Gem Hack and Ghost Protocol.

It's time for Security Now!, the show where we cover the security practices of a wide-ranging spectrum of people including city governments and people like you and me with this guy right here, Mr. Steve Gibson, our security in chief enforcer. Hi, Steve.

Steve Gibson: Leo. Great to be with you again. As a matter of fact, we have a listener who is in one of the townships that we have been discussing. And so we have some interesting insider feedback from that person, who was a little chagrined, he wrote, to hear his town named in Security Now!.

I said last week that I wanted to talk about this GCHQ Ghost Protocol. But I didn't feel like that was going to be enough to talk about. Now I'm kind of thinking, well, maybe, because it keeps bringing me back to my annoyance with the fact that the arguments being made are semi-specious. And we'll get to that because I don't - our listeners get

confused when they hear me suggest that maybe encryption is not always going to be absolute. They think that I'm promoting the idea that it's not always going to be binary. And that's not it at all. I'm annoyed at the argument that it can't be done. It can be done. But the proper argument is, is that what we want? I mean, so that's the proper place to frame the argument. Anyway, we're going to get to that.

Also there was an interesting attack on a Ruby Gem repository. And what was so fascinating was that the person who discovered it explains the process that he went through. So I thought that would be sort of a cool anatomy of like how repositories get infected and exploited. But overall, there were like sort of a number of things sort of, you know, sometimes the podcast acquires a theme sort of by itself. And in this case there were a number of instances where our technology appears to be colliding with the status quo. So I want to discuss a number of instances where there are mounting tensions between the way things have been and the new capabilities that are being created by connectivity, huge storage, amazing computation, and sort of the problems that those are creating.

For example, we're going to look at a surprising backlash to Mozilla's privacy-enhancing DNS-over-HTTPS support. We've got concerns over the use of facial recognition and automobile license plate scanners, and also the future of satellite-based Internet services that will create censorship problems potentially for repressive governments. We also have some SQRL news, a bunch of closing-the-loop feedback from our listeners - I teased one of them before, but there's more - and then we'll finish, taking a look at this Ruby code repository attack and also sort of, again, taking a look at this Ghost Protocol, the idea that the GCHQ has proposed, and what I think is the unsupportable argument against it and sort of wishing that we just used the argument of what do we want? Technology will give us anything we want, so we just have to ask.

Leo: As long as you consider the unintended consequences, I guess, because there's always that, yeah.

Steve: So xkcd has done it again. This cartoon, our Picture of the Week, made the rounds a few weeks ago, and I just had to push it out on our stack in favor of things that were a little more apropos. But I wanted to get to it. So it's xkcd.com/2166. And we've talked about a number of different places where software architecture forms a stack. There's the famous networking stack where you have the physical layer, then the logical layer, then the transport layer, application layer, and so forth, the idea being that there's a stack of layers where each layer has its functions constrained to do just one specific well-described and well-defined job. It relies on services provided from the layer below, and it offers services up to the layer above.

And so this notion of a software stack takes many different forms in our industry. You can also have like a development stack, or you have a server on the top and then a language that the server calls in order to render pages for the website, and it then relies on underlying libraries. So again, there are different types of stacks.

Anyway, this one is great. It's titled "The Modern Tech Stack." And at the very bottom of the stack we have "Massive Undiscovered Hardware Vulnerability." Then on top of that we have "Compromised by a Foreign Government" or "Compromised by Our Own Government," you know, so basically that's the next layer. And then up above that is "Compromised by Unknown Hackers," "Compromised by Bitcoin Miners," or "Compromised by a Current Employee." And then above that is "Compromised by a Former Employee" or, on the very top, "Compromised by a Customer." In other words, pretty much every possibility. We've got customer, former employee, current employee, bitcoin miner, unknown hackers, our government, some other government, or the

hardware doesn't work. So, yes, welcome to the 21st Century of security software. Anyway, just another great cartoon by xkcd.

So you would never expect something as apparently clean, simple, and beneficial as enabling DNS lookups over HTTPS to stir up any controversy. Seems like a good thing. We all know what a problem it is that, despite every website we visit now being HTTPS, with few exceptions, where its identity is authenticated because we trust the entity that signed its certificate asserting its identity, and all of the communications back and forth are encrypted, relying on this TLS technology.

But unless special measures are taken, even though we've got all the sites we talk to are HTTPS, unless we do something special, all of our browsers' DNS queries to those websites and all other domains, each of the pages we receive from those websites, and as we know sometimes that could be hundreds of subsidiary domain queries because of all of the junk that is coming in from all corners, every one of those pages spews forth a storm of unencrypted UDP packets carrying a DNS query formatted payload in order to - so outbound is the domain name that our browser needs to get the IP address for, and the response is an IP or a collection of IPs where we can access that resource. They're all unencrypted. They're all in the clear.

So the concern has been that anybody passively monitoring our traffic, like our ISP, who's like first step off of our connection as our data goes out to the Internet, or, well, anyone passively monitoring our connection is able to see everything that we're doing, not into our connection, but they know where we're going based on the queries that we're making. But then also it means that, because it's unencrypted and unauthenticated by default, any active interception can manipulate the DNS replies to send our browser to some other server. And if that other server has arranged to obtain a certificate that we trust, and unfortunately with so many certificate authorities globally now and our browsers trusting all of them that's not a stretch, especially if the entity doing the interception is highly placed, then we don't really have any security.

So we've talked in the past about how Mozilla and Firefox and Google with Chrome are moving toward this idea of doing DNS in a different way, of tunneling DNS queries over TLS to a remote DNS server that establishes a single connection. So in this mode, which is described in RFC 8484, the browser no longer hands the job of DNS lookups over to its host operating system. That's what all of our browsers do now is the OSes do that job for the browser. Our OS has a DNS cache. It has a DNS resolver. So the browser just makes a query through an OS API saying, hey, look this IP up for me. That goes to the operating system, and it's our network adapters then that know the IP to query for DNS.

They typically get that by making a DHCP, Dynamic Host Configuration Protocol query, to typically our router that we have for our local environment. It may be a DNS server itself. But more often it's just passing through the IPs that it in turn received when it made a DHCP query to our ISP, using our ISP's DNS, which makes sense because we want the answer to come back quickly, and you want a DNS server to be close to you so that you're not - because all of your connections wait on getting the IP to connect to after sending out DNS queries. So DNS response time is very important.

So about a year ago, more, about a year and a half ago actually, Mozilla began experimenting with this. We talked about it at the time. And it was, by every measure, a complete success. That is, the idea of establishing a connection, an HTTPS connection to what's known as a DoH, DNS-over-HTTP, a DoH provider, and then tunneling the same lookups through that connection. So what that means is the browser no longer asks the OS. It itself establishes the connection. So what this provides, of course, is complete protection from DNS UDP passive monitoring or active interception. Cloudflare offers a service. Google offers a service. Quad9 offers a service. So there are multiple places now, providers that are well connected, global, and offering DNS-over-HTTPS.

And it turns out, for example, Firefox. You go to, in Firefox, go to Options in the browser's menu, scroll to the bottom of the first general page that is displayed, click on Settings there under Network Settings, and at the bottom of the dialog that pops up you'll find a checkbox: Enable DNS Over HTTPS. So you put a check in the box. And unless you have some reason for choosing some other DNS provider, you leave it set to Cloudflare, which Mozilla has established a relationship with for this purpose.

And now all of your DNS queries are - so at that point, when you turn that checkbox on, Firefox opens up a single persistent connection to a local Cloudflare endpoint using their global geolocation-aware technology so that you're connecting to someone relatively close to you. And all DNS goes through this HTTPS tunnel and is replied to. So from someone passively monitoring your bandwidth, whether they're on the Intranet or out close to you like your ISP on the Internet, suddenly, like, oh, what happened to DNS? This person's not making DNS queries like they used to before.

Okay. So that seems all good. What's the problem? Believe it or not, entities in the U.K. are all up in arms.

Leo: They were pissed off.

Steve: Yes.

Leo: They called them the "Internet enemy."

Steve: I know, Leo.

Leo: By the way, as soon as I read that article, I started using Firefox. I said, that's it. If they're the enemy of the Internet, I'm using them. Holy cow.

Steve: I know. The ISPA, the Internet Service Providers Association, has named Mozilla one of the three finalists for their Internet Villain of the Year.

Leo: Yeah, they're villains.

Steve: For, quote, I'm quoting them, "their proposed approach," that is, "their" as in Mozilla's, "proposed approach to introduce DNS-over-HTTPS in such a way as to bypass U.K. filtering obligations and parental controls, undermining Internet safety standards in the U.K."

Leo: Shocking. Shocking.

Steve: And Leo, okay, so I'm at this - I go to this ISPA page, the Internet Service Providers Association, as I'm pulling the pieces of this thing together. And I just had to shake my head when, at the bottom right-hand corner, I was presented with this popup that I have in the show notes here. It says, down in the lower right-hand corner for me - oh, and you've got it, too.

Leo: Yeah, because I'm using Firefox.

Steve: Uh-huh. It says - so this is the ISPA; right? Who is upset that Mozilla is going to be tunneling DNS-over-HTTPS for all the benefits that we just outlined. The popup reads: "It looks like your cookies are switched off."

Leo: Oh, yeah.

Steve: "To ensure the best experience whilst visiting our website..."

Leo: So British.

Steve: "...please consider allowing cookies."

Leo: Yes.

Steve: "You can find out how to change your settings or more about cookies we use at the bottom of this page." Unfortunately, I think it was covered up by the notice.

Leo: Yeah, I don't see it, yeah. I'm looking.

Steve: Yes, the ISPA has our best interests at heart. That's right.

Leo: Well, part of this is, remember, that the U.K. wants to do this licensing system that, if you don't get a license, you can't see porn on the Internet, because they're trying to keep porn out of kids' hands. So the people who are going to - one of the groups, you can go into a pub to get a license.

Steve: To monetize? Ah.

Leo: No, no. You can go to a pub to get a license.

Steve: Oh, you go in.

Leo: You have to prove you're - so you have to go with your driver's license or some age proof, and then you get a license that allows you to surf the Internet freely. One of the people who is doing this is a company called MindGeek, which runs YouPorn and pretty much every porn site you ever heard of is run by MindGeek. So you have to - what could possibly go wrong? - go to a MindGeek site, give them your driver's license, your passport, proof of age, and they will give you a license to use the Internet freely. This is such a bad idea.

Steve: I do think I remember seeing somewhere, as I was just scanning this, that they want your phone number, as well.

Leo: Yeah, oh, yeah, all sorts of stuff. The funny thing is they passed this bill and were about to implement it when they realized they hadn't told the EU because they thought, well, by now we'll be out of the EU. So they had to stop. They're not. So among all the other Brexit problems, this is another one. They actually had to put this change off until either they tell the EU about it or they're not in the EU anymore. And I don't think the EU will allow it. So it's just a mess. It's so - it's such a mess. Oh, my god.

Steve: Well, yeah. And so here we have a problem. We're advancing privacy and security for web browser users, yet there's, I mean, the other side of this is that - so from the research I did, in the U.K., ISPs are legally forced to block certain types of websites.

Leo: Right, right.

Steve: Such as those hosting copyright infringing or trademark content. Some ISPs also block other sites at their discretion such as those that show extremist content, adult images, child pornography, and so forth. These latter blocks are voluntary and are not the same across the U.K. But most ISPs usually tend to block child abuse content, which seems like a good thing. Unfortunately, of course, we all know that this isn't a strong protection anyway. I mean, so the idea being that there are filters that the ISP manages that match on known domain names and do not return an IP address or return some placeholder IP, you know, redirection page, if you try to get that.

In mid-May, Baroness Thornton, MP for the Labour Party, brought up the DoH protocol and its impending support from browser makers in a session of the House of Commons, calling it a "threat to the U.K.'s online safety." And, similarly, GCHQ, Britain's intelligence service that we'll be talking about later in the podcast, has also criticized both Google and Mozilla, claiming the new protocol would impede police investigations. And of course I cued on that because it's like, wait a minute. How would this impede police investigations unless there was passive eavesdropping going on over on the wires that would cause matches? And then GCHQ continues, saying it could undermine its existing government protections against malicious websites. Okay, but how tunneling your DNS queries impede police investigations is really unclear.

The Internet Watch Foundation (IWF), a British watchdog group, also with a declared mission to minimize the availability of online child sexual abuse content, also criticized both Google and Mozilla, claiming the browser makers were ruining years of work in protecting the British public from abusive content by providing a new method for accessing illegal content. Now, of course, remember that, like, any VPN does this; right? I mean, this is kind of a VPN for DNS. Whereas the way it's always been is that we've just been spraying unencrypted UDP out onto the Internet, now it's in a browser tunnel. I mean, they're not able to see into HTTPS content. Now we're simply HTTPS tunneling DNS queries. So yeah, that's, you know. But, oh, it's a new method for accessing illegal content.

In their coverage of this, ZDNet noted that essentially Google and Mozilla support for DoH effectively narrows down to the same moral dilemma that surrounds the Tor Project and the Tor network. Which, yes, it upsets people because it allows people to be anonymous. Browser makers, ZDNet wrote, "Browser makers must now decide if it's

worth supporting a tool that brings privacy improvements to millions at the expense of a few that may have to suffer."

When ZDNet asked Mozilla for a comment on its nomination, right, the Villain of the Year nomination, Mozilla replied: "We're surprised and disappointed that an industry association for ISPs decided to misrepresent an improvement to decades-old Internet infrastructure. Despite claims to the contrary, a more private DNS would not prevent the use of content filtering or parental controls in the U.K. DNS-over-HTTPS (DoH) would offer real security benefits to U.K. citizens. Our goal is to build a more secure Internet, and we continue to have a serious, constructive conversation with credible stakeholders in the U.K. about how to do that," Mozilla said. "We have no current plans to enable DoH by default in the U.K. However, we are currently exploring potential DoH partners in Europe to bring this important security feature to other Europeans more broadly."

I was unable to see anything that definitively talked about Mozilla's plans for default enablement. We do know, you know, one of my favorite coined terms is the "tyranny of the default." We know that most users will never dig down into any of their browser settings and flip any switches. So our listeners, and you, Leo, and I, you know, we have DNS-over-HTTPS. I just explained how to do it if you are a Firefox user. And it's coming soon to a Chrome browser, and probably a Chromium-based browser near you, which expands that field even more broadly. So it's a matter of turning that switch on.

What really matters, then, is whether our browser vendors decide to make it the default. And so what Mozilla is saying is no, no, no. We have no plans to make it the default in the U.K. I don't know whether that will change over time. Who knows what? But again, I thought this was a perfect example of where, yes, just improving things, making things more secure and private, does generate some backlash. And unfortunately, Mozilla - there are two other contenders for Villain of the Year that this association has named. We'll see who ends up getting the...

Leo: Strike Mozilla from the list because, as of today, the ISPA has withdrawn Mozilla.

Steve: Oh, yay.

Leo: I think they're a little stung: "In the 21 years the event has been running, it is probably fair to say that no other nomination has generated such strong opinion." Then they go on about to show all the great things. "The villain category is intended to draw attention to an important issue in a lighthearted manner. But this year we clearly sent the wrong message, one that doesn't reflect our genuine desire to engage in constructive dialogue. We are therefore withdrawing the Mozilla nomination."

Steve: Yay.

Leo: They still think it's important to "scrutinize the plans." And they say, oh, yeah, data protection, security, online safety, user choice, user consent. You've got to pay attention, Mozilla. So they got their attention. The award ceremony is day after tomorrow, in case you're curious.

Steve: What this certainly does is, if they want to impose these sorts of filters, they'll have to do it in a different fashion. But it is certainly, I mean, it's not robust to use DNS anyway. So this probably means they will end up with better filtering of the stuff they want to filter. And the vast majority of people will end up with way more privacy.

I did want to segue from this to mention DoH in SOHO routers. This is a feature that our listeners might want to be on the watch for when they're next shopping for a small office or home router or consider upgrading. The advantage of doing this, of course, is that then all of the network's DNS queries within your local LAN would be forwarded through the router to its own tunneling to Cloudflare or Quad9 or whomever. My current favorite router that I've been talking about, Netgate's SG - no relation - 1100, which runs FreeBSD and the wonderful pfSense firewall router, uses the unbound DNS server which has supported DoH from the start.

So anyone with pfSense, or even if you didn't use that router, if you're running pfSense on an old PC because, I mean, it's very lightweight. FreeBSD and pfSense, they have a bootable package, easy to install. So it's simple to set up a router. Anyone with pfSense can follow a link that I have in the show notes to the Netgate blog posting titled "DNS Over TLS With pfSense," which shows how to quickly enable this slick feature. And when you do that, your little router sitting there will open - it'll establish a TLS, you know, HTTPS tunnel to whichever provider, free provider you choose. I'm blanking on the DNS provider that we've talked about for years. I think they offer...

Leo: OpenDNS?

Steve: OpenDNS, yes. My bet is that they also offer a free DoH service.

Leo: I bet, too.

Steve: I haven't mentioned them, so I just wanted to touch on them. Also anyone using an OpenWRT-based router, that is, a router where you have reflashed the firmware to run OpenWRT, it also can do DoH for you. So if anyone is interested in pursuing this, and of course this is - not only does this then encapsulate your browser's DNS queries, but all of those made by your OS, all of your iOS devices running on your local WiFi connections and so forth.

Leo: OpenDNS uses DNSCrypt, which has the same effect, right, as DoH?

Steve: Yes, it does. There you install a local client in your machine as sort of a proxy, and then your OS's queries go through that client and over to OpenDNS.

Leo: They don't - "We're not using SSL. While we make the analogy DNSCrypt is like SSL in that it wraps all DNS with encryption, it's not the crypto library being used. We're using elliptic curve cryptography." Maybe they prefer it to SSL.

Steve: That's all good. Try...

Leo: It's a lot easier. You don't have to install anything.

Steve: Well, try OpenDNS space DoH and see if that...

Leo: I did. This is what I found, yeah.

Steve: Oh, okay. So if they're not, they're probably going to because they're going to want to not lose a bunch of people to Quad9 and Google and Cloudflare. And it's looking like DoH, I mean, DNSCrypt was - I would regard that as an early pre-standard proof of concept. But it's like it didn't win. And DoH is what we have an RFC 8484 for, and it's what all the browsers are going to be supporting. It's what you get, as I was mentioning, in our SOHO routers. So I think that's going to be the standard that ends up taking over. So I would think OpenDNS will probably bring up a service to offer that.

Leo: Yeah, unless they're all in on DNSCrypt. But you'd have to install DNSCrypt on your side to use it, so that's not so good.

Steve: Yeah, yeah.

Leo: It also breaks DNSSEC, which is another issue. Does DNSSEC still work with DoH? I'm sure it does.

Steve: Yeah, yeah.

Leo: Doesn't matter.

Steve: Yeah. Because DNSSEC are just internally signed replies where your local resolver says, oh, yeah, the signature works.

Leo: Right.

Steve: So remember all those crime shows 20 years ago where a computer's facial recognition system would bring up a photo of some suspect. And that big photo is always over on the left-hand side of the screen. I don't know why it's always on the left, but that's where it always is. And then a bunch of lines would be drawn over it by a computer to form a facial recognition map.

And then over to the right, and it also is always over on the right, the upper right of the screen, the computer would then scan through a gazillion faces, like just flicker through them super fast, presumably pretending to be matching them, and suddenly it would - sometimes it would slow down if it was a cheesy show, or it would just stop, which is what you would hope because it's not a roulette wheel. And you would be looking at the matching face. And then, after a dramatic pause to make sure the audience had all caught up, the personal details of the individual's criminal record would paint down the screen below the matching photo. So we've been seeing that for decades. And it was never true.

It turns out it's true now. This is one of those things which has sort of quietly happened. And it's being done at a level that surprises people who haven't been really paying attention. We've got, as we know, this explosion of mass storage that allows everything to be stored. Just that. It's like, not what is being stored. The answer is everything is being stored. And we've got crazy computational power. We've got a jump in AI capabilities recently, enabled by the crazy jump in computational power. And so this is all - what used to be fiction is now absolutely happening.

ZDNet picked up on some coverage in the Washington Post which did some investigative reporting based on work conducted by Georgetown Law's Center on Privacy and Technology. Over the past five years researchers at Georgetown issued public records requests which have revealed that federal investigators, specifically the FBI and ICE, have turned state departments of motor vehicles databases into what they described as a "bedrock of unprecedented surveillance infrastructure." And to be clear, this doesn't appear to be a direct violation of any privacy laws. And this is sort of one of the points of this podcast is that technology is often enabling things for which society has not caught up.

Leo: Yeah. We gave them our pictures for those drivers licenses, never thinking what the consequences might be once face recognition became widespread.

Steve: Exactly.

Leo: And there's nothing you can do now; right? I mean, that's it. It's done.

Steve: Yes, our face is out.

Leo: Grow a beard.

Steve: Yeah. Anyway, in recent congressional testimony...

Leo: Although I did read, and you've got to do this, that...

Steve: Okay.

Leo: You don't know what Juggalo makeup is. But you can look it up. Juggalo makeup effectively obscures face recognition. So I think, get ready, there's going to be an invasion of Juggalos.

Steve: Uh-oh.

Leo: Well, yeah. "Uh-oh" is right. I'll pull up a Juggalo while you continue, and you'll see what I'm talking about.

Steve: Okay. In recent congressional testimony, members were not happy. And having watched more than my share of congressional testimony, no one wants to have Jim Jordan, who is a very, shall we say, excitable and interactively aggressive Republican congressman...

Leo: No, he's scary.

Steve: ...from Ohio.

Leo: This is really scary. This guy is wearing Juggalo makeup, just so you know now.

Steve: Well, Leo, if my mother wore that, I wouldn't know who she was.

Leo: Right. It works. And I suspect we're going to live in a sci-fi future where people are walking around with this weird makeup. It comes from a band called the Insane Clown Posse. I'll leave it at that.

Steve: People don't like clowns, Leo.

Leo: No.

Steve: This is really - this is bad.

Leo: Juggalos are scary.

Steve: So as the Washington Post put it: "Rep. Jim Jordan (Ohio), the House Oversight Committee's ranking Republican, seemed particularly incensed during a hearing into the technology last month at the use of driver's license photos..."

Leo: Good.

Steve: "...in federal facial" - yes, good - "in federal facial recognition searches without the approval of state legislators or individual license holders. Jordan said: 'They've just given access to that to the FBI. No individual signed off on that when they renewed their driver's license, or got their driver's licenses. They didn't sign any waiver saying, "Oh, it's okay to turn my information, my photo, over to the FBI." No elected officials voted for that to happen.'"

Leo: Didn't need to, though; right? I mean...

Steve: That's my point is that we've just jumped ahead of legislation. The Washington Post said: "Despite those doubts, federal investigators have turned facial recognition into a routine investigative tool. Since 2011, the FBI has logged more than 390,000" - okay,

390,000 - "facial recognition searches of federal and local databases, including state DMV databases, the Government Accountability Office said last month, and the records show that federal investigators have forged daily working relationships with DMV officials. In Utah alone, FBI and ICE agents logged more than 1,000 facial recognition searches between 2015 and 2017, the records show. Names and other details are hidden, though dozens of the searches are marked as having returned a 'possible match.'"

The Washington Post said, back in May: "Both Democrats and Republicans blast facial recognition technology in a rare bipartisan moment. A committee hearing becomes the venue for worries about privacy and civil rights." They wrote: "Facial recognition technology endured fierce resistance in Washington on Wednesday" - that was May 22nd - "as both Republican and Democratic lawmakers criticized the artificial intelligence software" - and this, by the way, comes from Amazon - "as a danger to Americans' privacy and civil rights.

"At a time when most issues in Washington generate a starkly partisan divide, members of the House Oversight and Reform Committee were startlingly bipartisan in their condemnation of the technology, which federal and local law enforcement agencies are already using across the country to identify suspects caught on camera. Members blasted the largely unregulated technology as inaccurate, invasive, and having potentially chilling effects on Americans' privacy and free expression rights. Several voiced support for passing federal laws to restrain the technology's use before, as Rep. Mark Meadows (R-NC) said, 'it gets out of control.'

"Others voiced worries about the technology being used in the United States as it is in China, where it is critical to the government's systems of public monitoring and social control. Committee chairman Elijah Cummings said 'there's a lot of agreement' among lawmakers that the technology should be regulated. The question, he said, is whether the systems should face a moratorium while the technology is assessed or refined, or whether it should banned outright. The committee's ranking Republican" - that's Jim Jordan, who I mentioned before - "compared the technology to Big Brother in the dystopian George Orwell novel '1984' and said it threatened Americans' First and Fourth Amendment rights covering free speech and protections against unreasonable searches. 'Seems to me it's time for a timeout,' he said. 'Doesn't matter what side of the political spectrum you're on. This should concern us all.'"

So there was that. And then, in a related story that I just happened to catch, a CNN special report on the increasing prevalence of automobile license plate scanners, which I guess have sort of very quietly been installed everywhere, based on the story. In this particular story, a license plate was entered, and the entire past of the vehicle, everywhere its plate had been seen and scanned, back through time, was shown in this huge connectivity graph. And frankly, it was chilling. And not because someone did something wrong. We want to catch the bad guys. But it was just the idea of that sort of power in anyone's hands is a little bit chilling.

And of course it brought to mind our recent discussion of Google's Android phone tracking database that, remember, went back more than 10 years, the Sensorvault repository, which was accepting queries from law enforcement of the form "Give us identifier tokens for every phone that was in such-and-such a location at such-and-such a time." Anyway, so I guess my point is that none of this is science fiction anymore. Just a very short time ago it would have been impossible because we didn't have, you know, storage was too expensive to record everything. Cameras were expensive. So they weren't readily deployable everywhere as they are now. We didn't have the connectivity that we do now that glues all of this together, allows everything to be seen, everything to be scanned by AI, features to be extracted, those all to be stored forever, and then all to be glued together into massive databases.

So what we have is an ongoing debate as we struggle, and we've been focusing on this from time to time where this intersects the technology that we discuss on the podcast, is over the question of an individual's right to have absolute privacy. And of course we've talked about this notion of the expectation of privacy. That's one of the legal arguments is saying, like, well, when you're walking around out in public, you don't have an expectation of privacy, which you arguably do when you do things within the confines of your home, for example. But anyway, I think that, over time, this is going to be one of the interesting things where we watch our expectations and legislation catch up with the fact that technology has outstripped us at this point.

Leo: You've been talking, I mean, I remember when you talked about not giving your fingerprint to Disneyland.

Steve: Yes, yes.

Leo: And I think people just weren't aware of it. And so you've been talking about this for a long time. But who doesn't have a picture in the database at DMV? Who doesn't? I think it's up to the states to say no, you can't have that information.

Steve: So a rather liberal-leaning city, San Francisco, has banned the use of facial recognition.

Leo: We did a Triangulation with the guy who advanced that. There's another city in the country that's done it, too. I hope to see more cities doing that.

Steve: Yeah. Well, and I think that maybe pending legislation from Washington, local municipalities will take it upon themselves to just say no. And then, you know, until we figure out what kind of controls we're going to have on this. Because right now, I mean, for example, there's no search and warrant process. You know, we have a Constitution in the United States that says that privacy is not an absolute. If law enforcement can demonstrate probable cause to obtain a warrant to search something, then absolute privacy is conditional.

But in this instance, the FBI is not obtaining search warrants in order to search the DMV photo database. They're saying, hey. In fact, in this reporting, I didn't put it in the show notes, but there was a note that a letter was sent to some DMV office from the FBI saying, "We want to search your database." And the DMV said, "Yeah, okay. You're the FBI."

Leo: You remember the President's been telegraphing that they're going to do a big ICE sweep throughout the nation, and this is one of the most immediate uses of this because in many states, even if you're undocumented, you can get a driver's license. Oh, it is going to be a boon. And I don't know. This is a very scary prospect.

Steve: Well, and then you match the driver's license with figuring out the person's name. That connects them to the plate on the car. And now you have the plate scanners.

Leo: You know where they are, yup.

Steve: In the CNN story the police cars had roof-mounted cameras pointing in every direction.

Leo: Oh, man.

Steve: And as the cruiser drove down the street, this thing was scanning the plates of every parked car on both sides and doing a deep dive, looking for any problems.

Leo: City of Petaluma decided to upgrade their meter maids with, instead of having to do chalk marks, license plate recognition. There's a little side effect of that. Now you have a database of license plates and locations and times.

Steve: Yeah. Because a database of chalk marks really doesn't go very far.

Leo: Wow. It's a scary world we live in. I'm going to get Juggalo makeup for my Tesla. Steve, back to you, my friend.

Steve: Well, speaking of enhanced connectivity, we were talking recently about Russia and China's ongoing work to proactively, well, to attempt to proactively manage and control their citizens' access to unfiltered global Internet content. Both governments clearly see it as in their respective nation's best interest to restrict the flow of uncensored communications. As we've noted, this is at least theoretically possible for land-based wired communications.

We talked about how Russia is going to arrange with Roskomnadzor, whatever their technology group is, to force all Internet traffic through some pinch points that allows them to control it. And that means they'll have to have their own internal DNS, so they're setting that up. And our listeners reminded me after that discussion about wired solutions that the problem is much bigger for radio. And then I was reminded of this when I saw a story yesterday about Amazon's recent application to the United States FCC, our Federal Communications Commission, asking for regulatory approval for their Project Kuiper.

Wikipedia has this to say about Amazon's Project Kuiper: "In April 2019, Amazon announced that they would fund and deploy a large broadband satellite Internet constellation called 'Project Kuiper.' It is expected" - is it Kuiper? I think that's how you pronounce it. The Kuiper Belt? Yeah.

Leo: Yeah, the Kuiper Belt, Vancouver Belt, yeah, yeah.

Steve: Yeah. That "u" in front of the "i" always kind of throws me off a little bit. It's like, wait a minute.

"It is expected to take up to a decade to fully deploy" - get this - "all 3,236 satellites planned for the full constellation in order to provide Internet to 'tens of millions of people who lack basic access to broadband Internet.' The satellites will use an orbit with a height between 590 and 630 kilometers. Kuiper will work in concert with Amazon's previously

announced large network of 12 satellite ground station facilities." That's the AWS Ground Station Unit announced back in November of 2018.

"Amazon filed communications license documents with U.S. authorities in July of 2019" - that's this, so I was impressed that Wikipedia was right up to date - "which included information that the wholly owned Amazon subsidiary that intended to deploy the satellite constellation was Kuiper Systems LLC, based in Seattle, Washington. 'The Kuiper System will consist of 3,236 satellites operating in 98 orbital planes at altitudes of,' in this case I'll do it in miles, 370 miles, 380 miles, and 390 miles." So I guess three different shells.

"The Kuiper System includes high-performance satellites, terrestrial gateways, internetworking technologies, and a range of customer terminals," says Wikipedia. ZDNet wrapped up their coverage noting that Elon Musk's SpaceX Starlink broadband satellite system was deployed about a month ago. It uses 60, six zero, satellites, and all but three of those are functioning as intended. Additionally, SpaceX has been given permission by the FCC to deploy up to 7,000 satellites in the future. And both OneWeb and Facebook have outlined their own plans to "monetize space." Love that term.

Leo: There's going to be 100,000 satellites up there. It's crazy.

Steve: Right? It is nuts. And who's going to keep them from bumping into each other is what I want to know. You'll have to - how do you launch a rocket without running through the flight path? So, yeah, literally tens of thousands of satellites.

Leo: Yeah, Elon wants to do 12,000 by himself.

Steve: Whoa.

Leo: So just add Amazon, Facebook, and everybody else. I love the idea of Internet, high-speed Internet in low Earth orbit, low Earth Internet satellites, to everywhere in the world. That would be amazing. In fact, it'd transform my life because I could do this show anywhere.

Steve: So it appears...

Leo: But I also want to see the sun once in a while.

Steve: Why is the sky gray? Well, once upon a time it was blue. But then we gave ourselves a cloak of satellites.

Leo: Google the Kessler Syndrome because that's the real fear of all of this, that if they start to collide, you'll have a chain reaction.

Steve: Oh, I know.

Leo: When you have so much debris around the Earth, we won't be able to save it.

Steve: So of course this makes you wonder what happens with China and Russia because we've got all these satellites zipping around all over the place. I guess they could be programmed, and presumably they would be, if they're passing over nations that don't want their citizens to have access. They would just go dark during their overfly in order to not be present. But, you know, it'll be interesting to see how this all evolves because, boy, as you said, Leo, the sky is going to be full of Internet.

Ashley Cawley, oh, he posted in the newsgroup, and I wanted to just highlight this for our listeners. He says: "A couple of us at work today installed Daniel's great WordPress plugin for SQRL and gave it a whirl. It worked beautifully for us and made a great demo." He wrote: "I'll be demoing it to more of my work colleagues tomorrow. I also have a number of ideas for further promoting SQRL. Keep up the great work, all." And you mentioned this last week, Leo, and I created one of my new GRC shortcuts: grc.sc/sfw, grc.sc/sfw. SQRL, well, not only Safe for Work, but also SQRL for WordPress.

So there it is, an official, on the WordPress.org site, an official SQRL plugin that allows SQRL to be used to log into WordPress sites. And it's been 100%. Someone was confused that it hadn't gone into the control panel to turn it on, that kind of thing. But basically it's there, and it's working.

Leo: Five-star reviews. And Daniel just updated it 13 minutes ago, so it's in active development, so that's good.

Steve: Is it zero - did he updated it to the .7? I think that's where he was, 0.0.7?

Leo: 1.0.2.

Steve: Oh, my goodness.

Leo: He's been busy.

Steve: He took it to 1.0. So, yeah. RayG posted, he posted in the - oh, as a consequence of this, I created another forum for - oh, I'm sorry. I'm getting my postings confused because I jumped to someone named SilverSword posted: "Installed from WordPress on a clean system. Giving the error: WordPress site running without SSL. Wondering if that's the reason." And so I wrote back to him, and I thought I would share with our listeners, that we did think years ago long and hard about whether to allow SQRL to be used with non-secured sites.

And what's interesting is that the protocol, SQRL's protocol is so robust, it doesn't need security. Unlike with usernames and passwords and one-time tokens, et cetera, the core SQRL technology itself can provide secure authentication even without the authentication and encryption that SSL/TLS provides. And in fact, my early implementations of SQRL worked over either HTTP or HTTPS.

We finally decided that since it really makes no sense to sign in to a non-secured site, since the browser's session cookies are readily sniffed to allow passive impersonation and session hijacking a la Firesheep - we all remember that mess - we didn't want to have

SQRL associated with those sorts of nonsecure sessions. I mean, like, yes, it's nice to have that authentication. But you can imagine people saying, well, look, I logged in with SQRL, but then my session was hijacked. It's like, yes, but that's not SQRL's fault.

Anyway, so even though it could have provided secure authentication reliably, we ultimately decided to have it require HTTPS. And so it would have probably the side effect of helping people who want to use SQRL to move their sites to HTTPS because of course now it doesn't cost anything to do that, thanks to the ACME protocol that's able to issue certificates on the fly. Oh, and I also saw this morning someone, RayG, posted in SQRL's Web Browser Extensions Forum. He said: "I have it running in MS Edge Chromium, and it is all working okay. Imported my ID and set it up, and I can log into the forums and the GRC demo sites."

So we also have, SQRL has a web browser extension which is running - it was developed under Firefox. But because it uses the common browser extensions architecture, the API that we finally now have across browsers, it runs under Chromium, and that means it's good for Chrome, Google's Chrome, and also, as Ray has just posted, Microsoft Edge, the Chromium version of Microsoft Edge. So it's getting around.

Some feedback from our listeners, closing the loop. And here was a person who, again - and I'm going to step in this here in another few minutes. David D. tweeted: "@SGgrc I can't believe you're discussing retroactive decryption of people's conversations for the sake of law enforcement. Why stop there? Let's record everyone's private conversations for easy access at any time we figure they may have done something. Sure, the FBI is giving mass murder as justification. But everyone knows once they have that tech in their hands it'll be a free-for-all. At the first indication of any wrongdoing, your digital life would be fair game."

And so I'm going to be clearer about this when I talk about this Ghost Protocol. As I mentioned at the top, I'm not advocating this. I'm just saying that arguments that this is about technology are absolutely wrong, and that an argument founded on a weak premise, on a fallacious premise, isn't going to hold. So what we need to do is decide what it is that we want, and then have the technology implement that. Again, as I said, I'll be clearer about that than I have been. But these things get, the different arguments get sort of mixed together. And lawmakers can sense that there's something not right about that. And so it weakens the argument overall.

Oh, and we have a - this was sent from "a relieved IT professional" who has made himself deliberately anonymous. He said "Location Somewhere." The subject was "The Florida Ransomware Attacks." He said: "Hey, Steve. I've listened to your podcast for a few years now. I thought, and hoped, my town would never make the podcast. But now you've probably seen the ransom of $460,000 that was paid here in Lake City, as I heard Leo mention on TWiT. But it was almost the same time that Riviera Beach paid theirs, so I don't recall it making Security Now!."

And actually he followed up later, after hearing that podcast where we were talking the next week about Lake City. And he said: "Thankfully, that was not the agency I work at, but I thought you'd be interested that this seems to be the same attack attempted on many government agencies in Florida, including the one I work at. We had a mass email come to us from a hijacked account at a certain state-level agency which contained an MS Word document with a malicious macro. The document was made to mimic an error dialog, and tried to coerce the user into clicking 'Enable editing,' which of course allows the macro to run."

He says: "I de-obfuscated the macro to find the link it was downloading from, in order to verify that if those that it was mailed to opened it on our network, that it was properly blocked by our firewall."

Leo: Smart. Good man.

Steve: Yes. "Thankfully," he says, "the link was blocked as it was hosted in Russia, which we geoblock." Which again, super smart. Why allow outbound connections of a local municipality to Russia? Just seems like that's only going to be a bad thing. So very nice IT work. He says: "Everything I've been told about the City Hall attack lines up perfectly with this attempted attack that went to many local agencies around Florida, so I suspect they didn't have particularly restrictive network firewall rules."

And that's probably why Brian Hawkins got his butt booted is that other, you know, they probably said, City Hall probably said, "Hey, why didn't other people get hit?" And the answer was, well, we have firewalls that block access from Russia. And then City Hall says to Brian, "Why didn't you do that for us?" Anyway, I also should say I got a lot of email from people who were very sympathetic to - and so on behalf of everyone else who wrote to me saying, "Hey, Gibson, we have no budget, we have no time, we have no equipment. You know, we're screaming that we want this stuff. Nobody will give it to us. We're doing the best we can."

And oh, by the way, and this was not these individuals themselves, but people explained that typically third-tier techs are the people who are put in charge of these things in local municipalities, you know, not top-end people. So again, it's a sense of that, you know, lack of budget, lack of time, doing the best they can, maybe not being like listeners to this podcast, and so not knowing everything that is possible.

Anyway, he said: "Sadly, the attacks seem to be going around here, and just this week" - and then he refers to Georgia's court system being hit. He says: "So the lesson here: Tighten up your network firewalls; and, more importantly, keep cold backups." And I think he made a - oh, it may have been in a follow-on message that he talked about keeping backups offline. So anyway, thank you very much, a Security Now! listener filling us in on some details. Those are interesting. And the idea of geoblocking by country absolutely makes sense.

Leo: At .cn and .ru, for sure.

Steve: Uh-huh, yeah. Also, I'm sure you know about this, Leo, because there's been dialogue in various groups. Charlie sent with the subject "Help Spamgourmet." And he said: "The fellow who runs Spamgourmet is seriously ill. He's transferred some of the duties to family members and friends. Now would be a good time to support this valuable resource with donations or technical support. Currently they have stopped accepting new users. Please consider making a mention of this on Security Now!. Charlie from Natick." And he said: "Only one Natick in the U.S.A.," so we have to Google it.

Leo: It's in Massachusetts. Natick, Mass.

Steve: Ah.

Leo: Natick.

Steve: And he said: "PS: Even older than you, my first flip-flop was made with a dual triode." Oh, that's cool, a dual triode, because you could get two triodes in one tube. And so if you cross-connect them, a triode could be set up as an inverter, and so you cross connect them, and you get a flip-flop. Anyway, he said: "My first flip-flop was made in a dual triode. This convinced me that digital electronics would go nowhere. Maybe I was wrong, but we'll see." So Charlie has a sense of humor.

Frank Pielhau in Kenosha, Wisconsin. Subject is "Mailman and Security Now! Listener." And I got a kick out of it. He said: "I'm a mailman in Wisconsin; and as I deliver the mail, I love listening to your show."

Leo: Perfect.

Steve: Yeah. "Fifteen years ago I was downsized from my IT job and, with a new baby at home, applied for all jobs I found in the paper. The post office called first. As time went on I started to get comfortable with the steady work hours as a mailman, not being on call, and the crazy late nights," meaning his life as an IT person. "A couple of years ago decided it was time to start studying and preparing to get back into IT now that the kids were older. Security Now! was one of the podcasts I started listening to. The more and more I listen to your show, the less and less I want to get back into IT." Yeah. "This is nothing against you."

Leo: No, I think being a mailman's nice. It's a nice walk. You get to enjoy the people. Life is good.

Steve: ITProTV.

Leo: Yeah.

Steve: And he says: "This is nothing against you, and I love your show. I ask myself every day, do I want to take on the risk and responsibility of managing the security of a network?"

Leo: Crazy.

Steve: "The digital world can be a scary place. By Episode 999, my studies in underwater basket weaving should be complete, so we can retire together..."

Leo: Excellent major.

Steve: "...and move on with our lives. My one question is this: When you retire from Security Now!, will you still dabble in IT? Or will you move on to a new hobby and leave the IT industry for the birds?"

Leo: Good question.

Steve: I have no answer to that question. Depends upon what else is going on around then.

Leo: What he's talked about doing is writing his own operating system for his PDP-8.

Steve: Yeah, I've decided against that.

Leo: Good. I want you to learn - I want you to, first of all, get rid of all the Windows computers in your house. The only reason you have those is because of this show and SpinRite. Once you retire, no more Windows. Okay? Can we agree to that?

Steve: I think that would be good. Yeah.

Leo: Get yourself a nice BSD box.

Steve: That's what I want to do. FreeBSD.

Leo: Although, you know, I've been playing with OpenBSD. I've always had trouble getting BSD installed on hardware. But older ThinkPads and OpenBSD, it's practically a no-brainer. And I know FreeBSD is reputed to be more secure, but I think all the BSDs are pretty the same.

Steve: Oh, they are. They are very, very strong.

Leo: Yeah, yeah. Open BSD is the one I've got on my - I've got it right here, actually, on a USB key. I'm going around installing it.

Steve: So I got email from a Bill Rakosnik near Athens, Georgia. The subject was "SpinRite 6.1 Level 4 Speed and Empty Sectors." He said: "Steve, on a recent Security Now! episode you said that SpinRite 6.1 will be able to complete a 2TB drive in just over three hours." That's right. He says: "I'm sure that assumes that the drive is healthy and doesn't need data recovery." Okay, that's true, except for what it's worth, if there were problems, they would be small relative to that. So, and you don't mind having it slow down and get to work if it's actually doing data recovery somewhere.

What you want is you want a maintenance pass that finds problems and causes the drive to spare them out before they become in need of deeper data recovery. You want that to be practical. And 2TB in three hours moves this into practicality for maintenance. He says: "What about the speed of SpinRite 6.1 on Level 4, assuming a healthy drive?" Now, of course that's going to be a lot more time, well, somewhat more time consuming because SpinRite reads the track, inverts the data, writes it, reads it back, reinverts it, writes it again, and reads it again. So it's going to be, what is that, like five times slower? And that's, again, so that's going to slow things down.

I would argue that a Level 2 is where a lot of people are going to be spending much more time productively because error correction has advanced to such a degree that, if maintenance, if a Level 2 scan is done periodically, that's going to catch problems. But,

for example, maybe you run a Level 4 scan before you deploy the drive. So you just set it up and let it go, and now it's going to take 10 hours for 2TB, which, again, sounds like a long time.

But has anyone recently tried to format a drive without using the quick format option? That is, to actually go out and, you know, what formatting a drive does, if you don't do the quick format, is it goes out and reads the whole drive. And it takes a lot longer than three hours to do that for a 2TB drive because it's not super optimized the way SpinRite 6.1 will be with its 32MB read buffer that never misses a revolution. You know, that's how we're able to get it to go so fast. So anyway, 4, because it's doing so much more work, will definitely take longer.

I will do everything I can to make it as fast as it can be. And, well, in fact, what I can guarantee is it is not possible to do it faster than 6.1 will do it because it will never miss a rev. So it'll be constrained by the raw data transfer rate of the data off the drive, which is to say, how fast is that drive spinning? Drives that spin faster will be able to go faster with SpinRite. Anyway, he ends up saying...

> **Leo:** That's a good slogan. I like that. Drives that spin faster will be able to go faster with SpinRite.

Steve: I'll hold onto that one, yeah. And of course I don't know what that means for SSD. They ought to really scream.

> **Leo:** They don't spin at all, yeah.

Steve: I think SpinRite will just really move through an SSD. Okay. So the Gem Hack. I wanted to talk about this because this was a nice anatomy of, I mean, we get to take a look into exactly what happened and how it happened. The headlines covering this that I first encountered read "Ruby Library Strong Password" - that's the gem - "Contains a Dangerous Backdoor." And then it went on to summarize: "An attentive developer located a bogus version of a new Ruby library that he used for his software."

And I'll just note that, of course, if you're going to have a language called Ruby, then its libraries have to be called Gems; right? So they said: "The uploaded version carried a dangerous backdoor that enabled the attacker to execute code remotely. Admins are urged to downgrade to the previous version, as they are running a risk with the latest one." Yeah, no kidding.

So then, dipping in sort of to the next level of detail, developer Tute, T-U-T-E, Costa has recently discovered a serious problem with the "strong_password" v0.0.7 Ruby library that injects a middleware into the code when deployed on production systems. The library was hijacked by hackers to enable them to silently and remotely execute arbitrary code on the compromised machine. The backdoor would send information about the infected URL to its command-and-control server via HTTP, with the instructions arriving as cookie files that were then executed through the eval function. If the deployment occurred in a production machine, the gem would download its payload from Pastebin.com, the popular text storing and sharing website.

Tute Costa has published the details of his discovery, which makes for a terrific anatomy of a repository breach. I have the link in the show notes. And Leo, I've got a couple code snippets which are simple enough that it'll be kind of fun to show them. So Tute wrote: "I recently updated minor and patch versions of the gems our Rails app uses. We want to

keep dependencies fresh, bugs fixed, security vulnerabilities addressed, while maintaining a high chance of backward compatibility with our codebase. In all, it was 25 gems we'd upgrade.

"I went line by line linking to each library's changeset." And he says: "This due diligence never reported significant surprises to me until this time. Most gems have a changelog.md file that describes the changes in each version. Some do not, and I had to compare by git tags or commits. The jquery-rails upgrade contains a jQuery.js upgrade, so the related log was in another project."

He says: "And I could not find the changes for 'strong_password.' It appeared to have gone from 0.0.6 to 0.0.7, yet the last change in any branch in GitHub was from six months ago, and we were up to date with those. If there was new code, it existed only in RubyGems.org." He says: "I downloaded the gem from RubyGems and compared its contents with the latest copy in GitHub. At the end of lib/strong_password/strength_checker.rb version 0.0.7 there was the following." And he shows a little four-line snippet.

He says: "I checked who published it, and it was an almost empty account, with a different name than the maintainer's, with access only to this gem. I checked the maintainer's email in GitHub and wrote to him with the prettified version of the diff." So he took the difference between the 0.0.6 and the 0.0.7. That identified this four-code change that it made to 0.0.7. When you run it through a prettifier, basically it makes it more legible because in this case the indentation of the loops and the flow of control was obscured by the fact that it had been condensed down into just four lines.

The prettified version is very readable, and it shows that, at the beginning of that strength verifier, a new function is defined with the name _!. And that is started and set up to ignore any exceptions that are raised. So it'll skip over any errors. The function, when executed, creates a new thread, which loops. The first thing it does is obtain a random value from zero to one, which it multiplies by 3333. Well, there are 3,600 seconds in an hour. So this is, basically, it's obtaining a random - it's that thread after being created sleeps for a random amount of time up to just short of an hour, up to 3,333 seconds.

Then it calls the eval function, which will execute anything that it is given, that is, the argument that is given. In the eval function is an HTTP GET query to a Pastebin URL. So in other words, this dynamic - so after waiting a random amount of time, somewhere between zero and just shy of an hour, this will query this particular Pastebin URL, get its contents, stick that into the eval, which the system will then run. So it's a very clever, simple backdoor. It's four lines of code, of Ruby code, added to the end of this function.

So 15 minutes after the person who discovered this wrote to the maintainer, Brian McManus, who is the maintainer, wrote back saying: "The gem seems to have been pulled out from under me. When I log into RubyGems.org, I don't seem to have ownership now." He said: "Bogus 0.0.7 release was created 6/25/2019." And then our author, who's writing this, says: "In case the Pastebin got deleted or changed, I emailed the Pastebin that was up on June 28th." Basically he extracted what that was that was being sent down. And so he then shows us the code which would be executed on the system.

And he says: "While waiting for their answers, I tried to understand the code. If it didn't run before, that is, checking for the existence of a dummy Z1 constant, it injects a middleware that evals cookies named with a double underscore id suffix, only in a production environment" - that is, not a debug environment - "all surrounded by the empty exception handler function that's defined in the hijacked gem. This opens a door to silently execute remote code in production at the attacker's will."

And so, anyway, they basically reverse engineered this. I'm confused here by my notes. Rafael Franca replied 25 minutes after he had been contacted. Someone at RubyGems yanked that off of the repository, and it was cleaned up. They got a CVE issued, CVE-2019-13354, which was used to announce the potential issue into production installations since it had been there for a while, and anybody who had updated to it would have this backdoor installed. So you would want to roll back and find out, also verify that your system had not been compromised. Maybe if you have any cold backup from before the update, just go back to that because this gave attackers remote command injection capability into a person's system.

And then, finally, on July 8th, the author explained how he believes his account was taken over. And this is our final takeaway. He said he had his RubyGems account for long enough that two-factor authentication wasn't even an option. Back then he did not use unique passwords for different websites, and since then many services have been breached. Attackers may have guessed his credentials.

So the takeaway is use password managers, rotate weak passwords, activate two-factor authentication wherever possible. And I would suggest that that's another good reason to revisit any passwords you have not changed for a long time. Certainly new passwords are being created uniquely. You want to make sure that old ones have not still been allowed to languish somewhere where they are sensitive.

Okay. So the Ghost Protocol. As I mentioned last week, I ran across some interesting back-and-forth in the Lawfare blog, an "Open Letter to GCHQ" on what they called the "Threats Posed by the Ghost Protocol." The beginning of it is where most of the meat is, and it's short, so I'll just share this. They said: "Last fall, Lawfare published a piece by Ian Levy and Crispin Robinson of GCHQ entitled 'Principles for a More Informed Exceptional Access Debate.'"

They said: "Our organization, the Open Technology Institute, has worked alongside other people and organizations to coordinate a response from an international coalition of 47 signatories, including 23 civil society organizations that work to protect civil liberties, human rights, and innovation online; several tech companies and trade associations, including providers that offer leading encrypted messaging services; and 17 individual experts in digital security and policy." So you can imagine the position they have. You know, "No."

Anyway: "Our coalition letter outlines our concerns that the GCHQ proposal poses serious threats to cybersecurity and fundamental human rights." Okay, now, and here's where I say, wait a minute. Cybersecurity and fundamental human rights are two different things. So mixing them together is a mistake. Anyway, "...including privacy and free expression. We shared our letter with GCHQ officials on May 22nd and are now releasing it to the public as an Open Letter to GCHQ." I have a link in the show notes, if anyone's interested, to a PDF that's hosted over on Amazon.

They continue: "In their Lawfare piece, Levy and Robinson set forth their proposal for 'silently adding a law enforcement participant to a group chat or call.' This proposal to add a 'ghost' user into encrypted chats would require providers to suppress normal notifications to users, so that they would be unaware that a law enforcement participant had been added and could see the plain text of the encrypted conversation. Levy and Robinson state that they offer their proposal in an effort to have an 'open and honest conversation' about how law enforcement can gain access to encrypted communications. We appreciate this call for a discussion and have organized our coalition in response. Lawfare has already published other pieces addressing the GCHQ proposal." And then they have two links to things.

"Our letter explains how the ghost proposal would work in practice, the ways in which tech companies that offer encrypted messaging services would need to change their systems, and the dangers this would present. In particular, the letter outlines how the ghost proposal, if implemented, would 'undermine the authentication process that enables users to verify that they are communicating with the right people, introduce potential unintentional vulnerabilities, and increase risks that communications systems could be abused or misused.' If users cannot trust that they know who is on the other end of their communications, it will not matter that their communications are protected by strong encryption while in transit.'" Okay. Again, that's a different issue. So my argument here is that we get these very different things, technology and policy, intermingled. And that doesn't help.

"These communications will not be secure," they say, "threatening users' rights to privacy and free expression." Again, we're mixing things. "Our letter concludes by urging GCHQ to abandon the ghost proposal and any other approach that would pose similar risks to digital security and human rights." And I'll just step back and say, okay, what they're saying is there is no way that we will ever be happy with this being done. Which is their right. That's fine. But when they try to claim technology reasons, that's where they fall short, in my opinion. And they say: "And by noting that we would welcome further dialogue on these important issues." Uh-huh. "The Open Letter to GCHQ is available here." And then they had another link.

So for me, this response is disappointing because it is not factually accurate, and thus as an argument it is weakened. Some of our listeners from whom I often hear misunderstand me when I discuss the notion of any weakening of our presumed perfect, but far from it in practice, which is what this podcast is all about for the last 12.5 years, coming up on 13 - or, wait, maybe we're in 12 now, I can never keep it straight. Elaine always reminds me. [Year 15 begins August 20, 2019.]

So let me be clear. I am not advocating for government intervention in encryption technology. I am not. It would be fine with me if, as a society, we were to finally decide that some things need to remain totally private. Period. End of discussion. And if that's what we want, then that's fine. We really can't seem to pull it off in practice, but certainly it's a worthy goal. And I would agree with people that anything that deliberately weakens it is weakening of it in practice. So my only complaint is that the designed-to-frighten nonsense being spewed, that there is no possible way for the technology to be adapted so that this could be safely accomplished is simply not true. And basing the argument on an untruth, as I said, weakens it.

So take the example I've been using for years, which remains true today, which is that, unless users manually manage the encryption keys for the conversations themselves, like for example Threema encourages - which is why, if I had a conversation where I absolutely have to have encryption or privacy, first of all, I wouldn't do it on a smartphone because it's just not safe. I mean, it just - it isn't. The idea that we have encrypted end-to-end communications with a smartphone, that's an illusion which, I mean, for all the reasons we've talked about on the podcast. But it makes people comfortable, so okay, fine. Okay.

But unless you're managing the key yourself, then you are inherently trusting someone else because keys have to be managed. So if you're not doing it yourself, you're inherently trusting someone else to manage them for you. And if that other party is not worthy of your trust, then you have no actual security. Yes, you have the illusion of security, but not true security. And that's fine for most people. Apple manages our iMessage keys transparently for us. Behind our backs. Which is exactly what most people want. They don't want to mess with all that. And Apple has made it very clear that they go to great lengths to protect us, and we believe them. We have every reason to believe them.

Leo: We have no option but to believe them because they don't disclose anything.

Steve: Right.

Leo: Which is why I don't use Threema, either. It's not open source.

Steve: Right. But iMessage supports multiparty group communications. And we assume that Apple is not allowing law enforcement to request the silent and hidden insertion of themselves and their encryption keys into any of those communications. So what if Apple were to add a facility to allow the silent and hidden insertion of an additional party into specific selected iMessage conversations under court-ordered search warrant? Like obtaining a warrant to bring up a phone tap in the old days. This actually does nothing to weaken the encryption. Okay, yes, theoretically, the more people you have in a conversation, from that standpoint is it weaker? Yes. As has always been said. If you want to keep a secret, don't tell anyone. The moment you tell someone, it's no longer a secret.

But from a technology standpoint, it isn't any weaker. The system is already designed to allow multiple participants to securely converse. This just silently adds another party to the encrypted conversation. Okay. Therefore, any decision about whether to allow law enforcement to eavesdrop on encrypted communications must fall entirely within the realm of is this what we want, which is where it properly belongs, rather than in it's not possible without collapsing the entire system. That part is not true. And lawmakers do correctly smell that that's not true. They don't believe it when people from Silicon Valley go testify. They just - they've seen what technology can do, and they're right. Any argument that's based, as I've said, upon a fallacy will eventually collapse.

So we are far better off if we argue based on the principles of what we want, what we agree we want the technology to do. Then we design the technology to implement that intention, which is what technology always does. And so I just wanted to - I wanted to clarify because this keeps coming up. And, I mean, this is a big issue, a big question for our society now. And it's going to probably, maybe, it's going to vary from government to government. As our listeners know, those who've been with us for a long time, the reason I bailed on creating CryptoLink, my version of an encrypted VPN, was that this felt like this was happening. It just felt like governments were going to outlaw encryption that they could not see into through some means. And, you know, this is very slow rolling.

And I'm glad this is rolling slowly because that allows these sorts of debates to be had over the course of months. And I think GCHQ is doing the right thing by saying, okay, how about this? I mean, that's the way you advance the dialogue. And anyway, so I wanted to just sort of weigh in that we have existing encryption systems that can allow this to be done without weakening them. The golden key analogy was a bad one. Backdoors are bad. None of that...

Leo: Yeah, but isn't this a backdoor?

Steve: No. It's not a backdoor. Right now iMessage allows group encryption, which Apple asserts is secure. In order for that to work, everybody in the group's phone has to be encrypting the conversation for every other participant. Yet that's all being hidden from us.

Leo: That's why, if you're intelligent, you don't use it. Here's my big fear with this is that people who understand this will, of course, including most terrorists and bad guys, will use some other system.

Steve: Yes.

Leo: No terrorist is using iMessages anyway. So who's compromised by this?

Steve: Yes. Yes.

Leo: It creates this huge have and have-nots between people who understand crypto and can use it protecting themselves, and those who can't. Those who can't will be the ones who will be surveilled.

Steve: Yes.

Leo: And the true bad guys will not because there always will exist a way to use true encryption that isn't government breakable. Period.

Steve: Yes. As we said years ago, that is out of the bag.

Leo: Yeah. So if I want - and that's why I don't use Threema or WhatsApp or even, frankly, Signal, although Signal is convenient, but it's tied to your phone number. PGP, as far as I'm concerned, or Gnu Privacy Guard, as far as I'm concerned, is a sensible way to do it. I would love for you sometime to look at a company called Keybase.io. They have a PGP-based encryption system. You generate your own key. You can create group chats through them with keys that you generate, not them, so they're not shared with anybody except your group. But you're right. You have to manage it. But Keybase makes it somewhat simpler by putting your public keys on a server. I would love to hear what you think about Keybase because I know a lot of people, including sophisticated security people, who believe that that is a secure way to group chat, and certainly a secure way to message. Keybase.io.

Steve: Cool. I will take, yeah, I have had people pointing me to it, and I just, you know, haven't made the time.

Leo: I feel - this is my biggest fear. Not that - I actually don't worry about the backdoor and stuff. I just feel like sophisticated criminals are going to know how to keep themselves private. Is that right? You can stipulate that?

Steve: Yes. I completely agree. Although, I mean, it's weird, too, because you do find evidence on smartphones.

Leo: Oh, there are stupid crooks.

Steve: Yes.

Leo: And this is what law enforcement's always told me privately is, well, those are the ones we can catch. We can catch the people who aren't sophisticated. And maybe the biggest threats aren't necessarily all the sophisticated ones. You're not going to get Ernst Stavro Blofeld, but you're going to get the average doofus who saw a video about extreme Muslim terrorism and decided to blow himself up and a bunch of other people. That guy you might catch. So maybe that's their thought. And actually I'm all for that. I just fear that the people who want and deserve real privacy aren't going to have the skills or information or knowledge to get it.

Steve: Yeah. Well, and I think that people who really need it, I mean, and the classic example was Glenn Greenwald and Edward Snowden.

Leo: Edward Snowden, yeah.

Steve: Where Snowden kept trying to get him to get PGP working, and he's like, oh, I didn't get around to it yet, blah blah blah. Because Edward knew that, you know, if you really need to have email encryption working right, you have to use PGP in order to do it, you know, not to just send text messages back and forth.

Leo: Yeah. And I guess it is true that you only have to make one mistake. Somebody's pointing out that the Internet Research Agency got caught because they forgot to use a VPN once. And so people make mistakes, and so maybe that's the other way you can...

Steve: Well, and we've talked about the idea that, sure, it's great if you've got indestructible, let's say military grade encryption over the wire, but you're typing on a keyboard that you didn't write.

Leo: Well, there's all sorts of - yeah, right.

Steve: Yeah, I mean, so there's the whole pre- and post-encryption issue, too. So, I mean, the idea - and this is why part of my annoyance is that, exactly as we were saying, this security that everyone says they want, they don't actually want. I mean, sure, if you check the box, would you like security or not, yeah, oh, yeah, check that box. How about pay $10 for it? What? No, I don't want to pay anything for security. It's supposed to be free.

Leo: So it really is a great debate. And you're right, this GCHQ paper was not, like, let's make this a law. It wasn't lawmakers. It was a proposal to discuss. And, you know, I don't care because I'm never going to use - I'm never going to plan my plot to take over the world on iMessages.

Steve: Well, and I fully recognize I'm not a good case because I'm saying to Lorrie I'll be a few minutes late for dinner.

Leo: Yeah, right.

Steve: I mean, that's what I use iMessage for.

Leo: I don't mind, exactly, I don't mind using SMS, which is also horribly insecure, for that kind of message. Who cares?

Steve: Yeah. And, I mean, the truth is, if I actually had something that absolutely had to be proof against interception, I wouldn't use any of this stuff.

Leo: No.

Steve: I mean, it's just, I mean, not really.

Leo: No.

Steve: We're 12 years into how badly all this works, unfortunately. In fact, the podcast is really misnamed. Should have been Security Never!.

Leo: Security What? Are you nuts? Steve Gibson is the man in charge at GRC.com, the Gibson Research Corporation. Go there. Hie thee to GRC and get SpinRite, the world's best hard drive maintenance and recovery utility. That's his bread and butter, but there's lots of free stuff there, including the latest on SQRL. Oh, so much stuff. GRC.com.

He also keeps the podcast there, 64Kb audio, 16Kb audio for the bandwidth impaired. And the most efficient form, plain text, ASCII text, created by the fine Elaine Farris out of each and every nugget that Steve puts out. She types it in, and you can read along.

Steve: Highly compressible. Highly compressible text. Because there's so much redundancy in what I have to say.

Leo: The thing that Elaine does so well, because honestly people don't realize this, but no one speaks as well as the written word. There's ums, and there's pauses. You drop a thought. You begin another thought. And a transcript can really highlight how weird the spoken word is.

Steve: Well, and when you and I say "Threema," she goes and finds it and makes sure that it's exactly spelled correctly.

Leo: Right. Good for her, yeah. She does a great job of making us sound halfway literate. GRC.com. We have our own audio and video available at the TWiT website, TWiT.tv/sn. And so of course if you subscribe there you will get a copy whenever you

want it, as soon as it's available, hot off the presses for your Wednesday morning commute, if that's how you listen. Or if you're a mail carrier, for your Wednesday morning deliveries. Just go to TWiT.tv/sn.

We also stream it live as we do it, 1:30 p.m. Pacific, 4:30 Eastern, 20:30 UTC every Tuesday afternoon or evening. Just go to TWiT.tv/live for a host of live audio and video streams. If you're doing that, please join us in the chatroom. There's so much fun in there. It's a great bunch of people. Vetman says, "Let's call it Insecurity Now!." How about that? Insecurity Now!. Irc.twit.tv.

Steve, thanks so much. We'll see you next Tuesday.

Steve: Thanks, buddy.