SECURITY NOW!

Transcript of Episode #796

Amazon Sidewalk

Description: At the beginning of this podcast, you're going to receive some details about another update to Chrome, and news of a few new high-profile ransomware victims. You'll learn about a breathtaking, remotely exploitable zero-click complete iPhone security compromise, as well as another significant big step forward for DNS privacy beyond DoH. We'll explain the nature of another serious and probably lingering problem within many Android apps. I have a few interesting bits of miscellany and SpinRite news to share. And before this is over, you will have obtained a full working sense for exactly what it is that Amazon has created and why, with their Amazon Sidewalk neighborhood IoT network concept, coming soon to all of your Amazon devices.

High quality (64 kbps) mp3 audio file URL: http://media.GRC.com/sn/SN-796.mp3
Quarter size (16 kbps) mp3 audio file URL: http://media.GRC.com/sn/sn-796-lq.mp3

SHOW TEASE: It's time for Security Now!. Steve Gibson is here. Lots to talk about, including one of the most breathtaking iPhone hacks ever. Thank goodness it's been patched. We'll also take a look at a replacement for DoH known as ODoH. ODoH. And Steve's going to analyze the security and privacy model of Amazon's Sidewalk technology. Should you leave it on? Steve answers that question next on Security Now!.

Leo Laporte: This is Security Now! with Steve Gibson, Episode 796, recorded Tuesday, December 8th, 2020: Amazon Sidewalk.

It's time for Security Now!, the show where we cover your privacy, your security, how the Internet works and a whole lot more with this guy right here, Steve Gibson. He's our Explainer in Chief from GRC.com. Hi, Steve.

Steve Gibson: Yo, Leo. Great to be with you again.

Leo: Nice to see you.

Steve: As we cruise into December. I got my phone. My phone sent off a warning, talking about the California lockdown. It was like one of those Amber Alert things.

Leo: Yeah, yeah. Don't go outside now. You've got to stay home. Stay home.

Steve: It's a strange time we're in.

Leo: We should also mention that now that we're in December, we're getting ready for our Best Ofs. There will be a Security Now! Best Of in two weeks, on the 22nd. And then we'll have a brand new show on the 29th. It's a little weird because of where Christmas is placed. But also that you're going to be part of our TWiT special, our holiday TWiT on December 20th. We decided to do OG TWiTsters, the original gangster TWiT team. So it'll be you. It'll be Paul Thurrott. It'll be Jeff Jarvis. It'll be me. It's going to be a lot of fun. And we'll just talk about the year gone by and some of the big stories of the year. And then of course what's ahead for 2021. So put your bookmarks in for December 20th for our holiday episode of This Week in Tech. And then December 22nd for the special Best Of Security Now!. But meanwhile, what's happening today?

Steve: So without question the most tweeted issue over the last few weeks, I mean, even my best buddy sent me a text saying, I just got this letter from Amazon, and this sounds really bad.

Leo: That's interesting, because I thought people weren't paying any attention to it. So I'm glad to hear there's some awareness.

Steve: Oh, no, no, no. Well, of course - yeah. And of course our audience is immediately, like, whoa, what do you mean you're going to be sharing my bandwidth with the neighbors? How is that a good idea? So this of course is referring to Amazon Sidewalk, which is the main topic of our podcast. But we have a lot to cover, some really cool stuff. At the beginning of the podcast, everybody's going to get some details about another update to Chrome and news of a few, I promise only a few, high-profile ransomware victims that are newly discovered in the last week. Everybody's going to learn about a breathtaking remotely exploitable, and I know you feel the same way about this, Leo, zero-click complete iPhone security compromise. And the details of it are fascinating. We've also got a significant big step forward for DNS privacy beyond DoH. And I love that this is more than just DoH because that just sounds like Homer Simpson, and we had to somehow get away with that.

Leo: Now it's ODoH.

Steve: We're going to explain the nature of another serious and probably lingering problem with many Android apps. I have a few interesting bits of miscellany, including as you and I were talking before the show what it is about the Apple M1 processor that makes it so agile with its ability to emulate the x86 Intel instruction set. I've got a little bit more SpinRite news to share. We of course were talking about the interesting discovery of timing that the ReadSpeed Benchmark has revealed. And then we're going to, by the time this is over, everybody who listens to this will have a full working sense for exactly what it is that Amazon has created and why, with their so-called Amazon Sidewalk neighborhood IoT networking concept, soon coming to all of the Amazon devices that people may have or will be getting in the future.

Leo: Yeah. And it's opt-in. You have to opt out. It's by default turned on.

Steve: Right. Well, so it's not opt-in. It is you are in...

Leo: It's opt-out, yeah.

Steve: ...unless you opt out, exactly.

Leo: Yeah.

Steve: Well, we'll get into this in detail. But for what it's worth, I understand what they're doing. We coined the phrase "the tyranny of the default" a long time ago on this podcast. If it required people to turn it on, it just wouldn't get off the ground.

Leo: Can't do it, yeah, yeah.

Steve: So it makes sense. The good news is, just from a standpoint of people questioning whether Amazon is doing the right thing, is that once the system is running, once it exists, when you install a new device, you will be asked one way or the other.

Leo: Oh. Oh, that's good. Oh, that's good.

Steve: Yeah. So it's only coming up defaulting to on for all the existing devices that are out there. But from then on you will be explicitly asked. So I think they're doing the right thing. And as we'll see, they've really nailed in this in terms of the goal of what this is. You know, 24 hours ago I had no idea. Now I'm going to explain it all, and everybody's going to know. And we do have of course a very cool Picture of the Week.

Leo: Picture of the Week time.

Steve: So this is kind of a fun one, just a proof of concept. You know, no one ever claimed that fuzzing, blurring, or pixelating passwords was secure. You know, it's something you see often in photos or sometimes in screenshots where they'll, like, blur out something like the IP address or something that you don't want someone to know. But this was a nice proof of concept because now we have proof that it is decidedly not secure.

This shows three lines of graphics. The first is pixelized, that is, the result from dramatically reducing the resolution of the screen so that you end up with something very blocky. But what's obvious to any of us who stop to think about this is that all of the information is still there. It's just been altered. So rather than lots of white space and a little bit of lettering, you've got these big blocks. But they show the average amount of ink under that region. And so thanks to the fact that text has only a fixed number of letters in its alphabet, there aren't that many possibilities.

So the second line shows their first stage of recovery of the original pixelated text. And then, I mean, you can just read that. But then they show the original as what was originally pixelated from. Anyway, I just thought it was a cool little reminder that some things that look like, it's like, oh, no one's going to be able to figure that out, well, if they have to, I mean, if they really want to, all the information is there. They just need to get a little clever. Of course, we're all used to, as techies, the annoying zoom in on the license plate on the TV crime shows.

Leo: Yeah, center, zoom, center, yeah.

Steve: Where the back of the car, you know, the license plate is like four pixels. And, but, oh, boy, do they have some serious AI because they zoom in on this thing, and then they press the "sharpen" button. And, yeah, you get the whole license plate back. It's like, oh, thank you very much. I wish I had one of those buttons around. But of course no one does.

A little bit of browser news. Very little. Nothing earthshaking happened in the last week. Chrome did again update its desktop builds, bringing them to 87.0.4280.88. This closes eight holes that had been discovered by researchers, so nothing critical, nothing zero-day that was actually being exploited. Just four high-severity problems and then four that were important. So worth getting. And this, again, as we know, Google and Chrome are my model for this is the way you do it. You're notified of problems. You fix them quickly. You push them out to the world. And, I mean, even if they were zero-days, and they were discovered being in use, the best you can do is remove those immediately. So Google is good at doing that.

It turns out that the world's largest electronics manufacturer, a little company we've all heard of thanks to their relationship with Apple, Foxconn, has recorded revenue last year of $172 billion and over 800,000 employees worldwide. Well, when you have that many employees, that's just that many more opportunities for someone to respond to a phishing email and click a link that they shouldn't have. Their massive facility in Mexico, it's 682,000 square feet. Leo, that's more than a half a million square feet. That's got to be some serious building.

Leo: That's big.

Steve: Whoa. Enough to apparently house between 1,200 and 1,400 servers. I mean, so this must be some serious data facility, as well. Over the Thanksgiving weekend they got hit by malware. The DoppelPaymer gang claimed to have encrypted those servers, between 1,200 and 1,400 servers. They didn't really care about the measly workstations. They just ignored those. They first, however, stole 100GB of unencrypted data, then deleted between 20 and 30TB of server backups that they were able to remove. And they're asking for a pretty penny. I guess they look at $172 billion in revenue from Foxconn, and they're rubbing their hands together. Maybe we'll find out what happened. They wanted 1804.0955 - lord knows where they came up with that number - of bitcoin. 1804, which at today's current rate would...

Leo: It's the commission, 095.

Steve: Just shy, well, it's just shy of 34, I'm sorry, $35 million.

Leo: Oh, my god.

Steve: $35 million.

Leo: OMG.

Steve: And so, you know, they think they encrypted a huge number of servers. They deleted 20 to 30TB of backups. There was some comment somewhere that I saw that there were maybe another 75TB they had not been able to get to, to delete, terabytes of backup. They did, they exfiltrated 100GB of unencrypted data. So maybe we'll find out what happens with this. But anyway, there was a big chip that fell. And I didn't realize, in doing a little digging into Foxconn, that Sharp and Belkin are subsidiaries of Foxconn. These guys are huge.

Leo: Yeah, yeah. Unbelievable. You saw that the latest trick with ransomware gangs is to call you up, literally, from a call center...

Steve: Yes, yes, call you on the phone.

Leo: ...and say, "We see you're trying to get your data back. Oh, I wouldn't do that if I were you." These people are getting bolder and bolder and bolder. $35 million. Unbelievable. Do you think Foxconn will pay?

Steve: It'll be interesting to see if we ever get a follow-up on that. I mean, I'm sure that they're juggling their options. Any of these companies are going to respond rationally. So they're going to look at what got taken.

Leo: Yeah, it's a business decision, yeah.

Steve: Pure business, from a pure business model. How down are we? It apparently did not get out of this one facility. So maybe they just can go, oh, yeah, so one warehouse is in trouble.

Leo: That's a lot of ransom. That's such a huge amount that you've got to think...

Steve: That is, I really - and I think this plays off of what you were just saying about how bold these guys are getting. Maybe they're also getting a little too cocky. It's like, yeah, these guys are big. Let's get $35 million. And it's like, okay.

Leo: The world's got to crack down on this.

Steve: There must be some formula, like how many servers you encrypted times something gives you X bitcoin? Because, like, where would they come up with 1,804.0955 bitcoin? It's like, what? Why not just 1,800?

Leo: Yeah.

Steve: So, wow.

Leo: So weird.

Steve: Meanwhile, Huntsville, Alabama City Schools district, which in 37 schools educates nearly 24,000 students, they were forced to send a note to the parents saying students, families, and faculty and staff should shut down their district-issued devices and ensure the devices remain off until further notice. Additionally, stakeholders should avoid logging on any HCS, Huntsville City Schools, platforms at both home and school. Yes, they were also a victim of ransomware. As was the Metro system in Vancouver. They were knocked offline by the Egregor ransomware, and all of the transit services printers began spitting out ransom demands. As we know, that's another new tactic that's being taken by ransomware to essentially shame the victim into paying by making it more difficult for them to keep the news quiet. Oh, and Egregor also got Kmart. Apparently Kmart is already struggling. It looks like the retail side didn't get hit. Apparently it was their backend systems. But they were hit by ransomware.

And lastly, a large online education company by the name of K12 Inc. has decided to pay a ransom. They were hit by Ryuk ransomware in the middle of last month, in the middle of November. They provide tailored online teaching curriculum for more than a million students that have used this K12 system. Anyway, their concern is the privacy problem. The Ryuk guys got into some back office systems that contained student data and other information. So they decided to pay up in the hopes that, first of all, they were insured, so they weren't bearing the full brunt of that cost. We don't know how much ransom they are paying or have paid.

But they're going to take the Ryuk crooks' word for their promise that they will not release the confidential data that the bad guys were able to obtain from their systems, and so they're going to cross their fingers and hope for the best there. So yes, just to kind of keep it on everyone's radar, ransomware is the scourge of the security industry at the moment, or certainly one of.

The Apple iPhone vulnerability story is interesting. And last week in a lengthy and painstakingly detailed 30,000-word report, which was written by Google's Project Zero's Ian Beer - Ian is someone whose work we've covered through the years, so his name may be familiar to our listeners. We learned, thanks to this report that Ian wrote, how he had spent the first half of this year. Yes, six months. But before explaining, Ian quotes from the February 2020 Offensive Con conference, during which its keynote speaker said: "Exploits are the closest thing to 'magic spells' we experience in the real world. Construct the right incantation, and gain remote control over a device."

So with that quote, this opens Ian's expos, which he began by saying: "For six months of 2020, while locked down in the corner of my bedroom surrounded by my lovely, screaming children, I've been working on a magic spell of my own." He says: "No, sadly not an incantation to convince the kids to sleep in until 9:00 a.m. every morning, but instead a wormable radio-proximity exploit which allows me to gain complete control over any iPhone in my vicinity." He said: "View all the photos, read all the email, copy all the private messages, and monitor everything which happens on those phones in real-time."

So in other words, in the midst of distractions from his screaming kids, he successfully discovered and then fully developed a working, remote, over-the-air, zero-click, total compromise of any Apple iPhone. And doing so was pretty much everything that Apple, as we know, has gone to such extremes to first totally prevent; or, if failing that, then to at least raise the bar of exploitation engineering so high as to make it incredibly difficult to weaponize. Yet Ian succeeded.

In his own words, describing the situation, he wrote: "Of course, an iPhone isn't designed to allow people to build capabilities like this. So what went so wrong that it was possible?" He said: "Unfortunately, it's the same old story, a fairly trivial buffer overflow programming error in C++ code in the kernel parsing untrusted data, exposed to remote attackers." He said: "In fact, this entire exploit uses just a single memory corruption vulnerability to compromise the flagship iPhone 11 Pro device. With just this one issue, I was able to defeat all the mitigations in order to remotely gain native code execution and kernel memory read and write."

He said: "Relative to the size and complexity of these codebases of major tech companies" - like Samsung, Google, Apple, Amazon, any of the big guys - "...the sizes of the security teams dedicated to proactively auditing their product's source code to look for vulnerabilities are very small." He said: "Android and iOS are complete custom tech stacks. It's not just kernels and device drivers but dozens of attacker-reachable apps, hundreds of services, and thousands of libraries running on devices with customized hardware and firmware." And in fact a little bit later we'll be talking about one particular library over on the Google side.

So in other words, with today's massive custom codebases, and more focus naturally being placed upon the fun side of designing and shipping new features over the drudgery of examining code for vulnerabilities, it's inevitable that these systems will incorporate a wide range of flaws of varying severity. You know, everyone knows that, generically, bugs of all kinds are being patched constantly. That's, you know, every second Tuesday is a Patch Tuesday, and Microsoft fixes a hundred-plus bugs of varying severity. So sobering though it is, it should come as no surprise, and it doesn't, that some of them, some few will be really bad.

Now, the good news is that this is Apple, not Android. So the handsets were all updated back in May with the move to iOS 13.5. And that's, as you noted over on MacBreak Weekly, Leo, the same update that gave us the COVID proximity tracking tech, that was back then in May. So at that time this breathtaking vulnerability was quietly eliminated. It wasn't, as far as we know, ever used. But of course you never do know about these things. You know, the fact that it was there raised some questions in Ian's mind, which we'll talk about.

Leo: I think a nation-state might well have had - one of the reasons I think this is - it seems like a natural place to investigate. You're going to talk about the exploit. I won't talk about it. But that's where I would have looked, too.

Steve: Yes. The heart of the problem that Ian uncovered was in a proprietary Apple WiFi-based peer-to-peer mesh network protocol, again of their own design, known as AWDL, Apple Wireless Direct Link. Being a peer-to-peer mesh protocol handler, AWDL needed to monitor and parse all WiFi network traffic. Parsers are a form of interpreter in that they examine a flow of data for the purpose of attempting to understand it and to see whether it's relevant to them. What Ian uncovered was that Apple's AWDL parser contained a flaw. Knowing that, and arranging to exploit it in the environment that Apple has deliberately worked to make so hostile, are two very different things, but Ian succeeded.

And I should note that his paper, I mean, his write-up is just amazingly thorough. It is a step-by-step walk through what it takes to do this. And I would commend it - I have a link in the show notes - commend it to anyone who's interested in like how you go with

Apple against all the mitigations that they have put in place, everything that they have done to, like, okay, even if you knew there was a problem, sorry, you can't get to it from here.

Ian noted that he is one guy who was working alone in his bedroom, albeit with some distractions; whereas the world currently contains, to your point, Leo, many powerful, massively resourced, nation-state, well-organized cyberwarfare groups. Ian feels quite certain that when such resources are brought to bear, other currently unknown exploitable vulnerabilities, maybe this one before he found it, will almost certainly be and maybe are being right now uncovered.

So anyway, no need to go into great detail. But the point was that, to your point, Leo, here is something that is having to listen, to look at WiFi packets. Ian referred to them earlier as untrusted, meaning that they're not data packets wrapped in a WiFi envelope where you've already established a connection. In order for this peer-to-peer system to work, things that just wander into range, where you're not exchanging a username and password, you're just accepting a connection among peers, inherently something is like sending out a beacon, and something is listening to all incoming traffic to see if it's a beacon.

And it turns out that he was able to - the iOS kernel has all of its symbols stripped. And symbols are a huge benefit to understanding what's going on. It turns out the macOS doesn't. It does not, this region at least, doesn't strip symbols. Which is very much more useful for developers who need to trace into something. Or if something crashes, you're able to get a sense for where the crash was. And it turns out macOS and iOS share the same AWDL. So he was able to identify the problem, use the symbols macOS had to give him a leg up, and then map that over onto iOS. Although he still had to perform the job of exploiting this, you know, turning it from a known vulnerability into a weaponized exploit. Again, especially in the world that Apple has created to make that well nigh impossible, he was able to do it. So really, really, really cool piece of research.

Leo: Yeah, I think if you were going to look somewhere, you'd look at AirDrop and say, hmm, that's promising.

Steve: Well, yeah. You would, I mean, depending upon your attack model, if you could briefly get your hands on the phone, then you look at the lightning connector, right, to see what you're able to do there. But, boy. And you made the point over on MacBreak Weekly that, I mean, this is wormable, meaning that...

Leo: That's a big problem.

Steve: ...the contagion could spread. Which meant, I mean, if somebody wanted to bring down the entire Apple iOS, at least iPhone ecosystem, they could have launched a worm into an environment. This thing would have jumped from phone to phone everywhere. I mean, like stick it, you know, start it off in a stadium, and every iPhone will be infected. Those will then, at the end of the game, spread out all over and infect all the other iDevices within WiFi range. I mean, and it's, like, you could have shut down iPhones globally, potentially. So that didn't happen because Ian Beer works for Project Zero.

Leo: Also because nation-states aren't real anxious about sharing their exploits either.

Steve: Exactly.

Leo: So if Israel or Iraq or Iran or North Korea had it, they wouldn't have told anybody. Or the NSA. They wouldn't have told anybody.

Steve: Well, yeah. And you have to wonder if, when this happened in May, there may have been some people here or abroad saying, ooh, you know, darn. Somebody else found the thing that we had been using in order to just beautifully infect targeted iDevices wherever we wanted to. So but again, look at the number of bugs that are constantly being fixed. As I said, every so often there's a really bad one. Well, that really bad one has been there typically for a long time. So I think, you know, we spend a lot of time talking about this or that thing that got fixed and, whew, isn't that good? But maybe the same guys are listening to the podcast going, yeah, Gibson, you just keep thinking that that's, you know, that we're in trouble now. We don't care. We have 12 others that allow us to do this.

Okay. As I said before, I am so happy that DoH is not in the future what we're going to be talking about.

Leo: Doh.

Steve: Because, yeah, exactly. Now we have ODoH, O-D-O-H. And you would be forgiven if you thought that ODoH stood for the Ohio Department of Health.

Leo: Which it does, actually.

Steve: Which it does. I googled ODoH.

Leo: And that's what you found.

Steve: To see what else would come up. And Ohio Department of Health was the first thing I got. But I would not be surprised, if this thing gets traction and takes off, if they don't drop down Google's search results eventually because ODoH could be the thing. We all know about DoH, D-O-H. And if we choose to use it, it provides end-to-end encryption between our browsers, and also more recently our entire operating system, and a remote DoH-capable DNS resolving service. As we know, by reusing the existing and quite mature certificate and protocol technologies of HTTPS, it very nicely does what it was designed to do. It strongly prevents any local agency such as someone monitoring a hotspot or our local ISP from observing our DNS queries. With DoH running, they see encrypted traffic that they cannot get into, and nothing else.

But there's still been one glaring privacy flaw in DoH. It doesn't prevent the DoH service from knowing who's querying DNS and for what. Since the DoH IP connection is point-to-point, the DoH resolver still knows the IP making the query and what domain they're querying for because it decrypts the encapsulated encrypted query in order to provide the answer. So your local ISP may no longer know, but the person you're asking still does.

This is the problem that the new Oblivious, as it's called, it's Oblivious DNS over HTTPS, has been created to resolve. And the solution is elegant because it's so simple. Simply introduce a connection-forwarding middleman into the point-to-point link. Add a third-party proxy which is unaffiliated with the DoH provider. Then the user's connection to their DoH provider routes through the proxy, which in turn forwards the still-encrypted traffic to the user's chosen DoH provider. Since the encryption is still encrypted end-to-end, from the user to the DoH provider, the proxy cannot see anything, can't see into the traffic at all. So although it does know who's asking, it's completely blind to what they're asking for.

The DoH provider in turn, which decrypts the incoming traffic, knows exactly what the incoming connection is asking for. But now it has no idea from whom the request has come, since it's receiving anonymized requests that have been forwarded to it through the intermediate proxy. So it's beautiful. I mean, it just adds an additional stage of blinding, just to the connection IP. And already DoH had no deanonymizing information in it. The actual query is just a DNS query, so there was nothing in the packet that needed to be sanitized. We just need to sanitize the IP connection address. And so routing the connection, the DoH connection through a third party, just a simple connection proxy, does it.

The concept has been co-developed by Cloudflare, Apple, and Fastly. It was announced this morning by Cloudflare. They already have a trio of qualified and fully independent privacy-committed ODoH proxy providers, and ODoH's formal specification is already moving forward through IETF standardization.

Eric Rescorla, who's currently the Chief Technology Officer, the CTO of Firefox, said: "Oblivious DoH is a great addition to the secure DNS ecosystem. We're excited to see it starting to take off and are looking forward to experimenting with it in Firefox." And if any of our listeners really needed, for some reason, or wanted to be on the bleeding edge, it's technically possible to get ODoH working today. But at the moment it takes jumping through a bunch of hoops. And given the speed at which DoH appeared and became widely available, and how simple this addition is, I have the feeling that flipping a switch to enable ODoH isn't far away. So I don't think people will have to wait very long.

Leo: I'm sure Firefox and, because it's Apple, Safari will support it pretty quickly out of the box.

Steve: Yeah. So Google Play Core Library problems. Google provides Android app developers with a component called the Google Play Core Library. The Android developer docs describe this component library by saying: "The Play Core Library is your app's runtime interface with the Google Play Store. Some of the things you can do with the Play Core include the following: download additional language resources, manage delivery of feature modules, manage delivery of asset packs, trigger in-app updates, and request in-app reviews."

So it's an API, essentially, interface to the online Google Play Store that allows apps to interact with the various Play services from within the app itself. So you could dynamically load additional code, additional levels of the game as needed, maybe pool locale-specific resources and of course interact with the review mechanisms. And since this is the officially sanctioned and recommended way to do this, many popular, I would argue well-designed, Android apps utilize this library. Those include Google's own Chrome, Facebook, the Android Facebook app, the Instagram app, WhatsApp, Snapchat, Booking, and even the Edge browser. Facebook and Instagram alone account for five billion and one billion downloads respectively.

So just imagine the total number of Android apps worldwide that have historically incorporated this library at Google's behest. And it's the right way to do it. If your app has some need to interact with the Google Play Store after it's been downloaded and installed, this is the officially sanctioned way to do it.

Okay. So the problem is a quite serious bug was discovered inside this very widespread common app library. And because the library is linked into Android apps to become part of them, this isn't something that Google can fix with an Android update, even for those Android smartphones that would be receiving updates.

So what's the bug? I'll quote the company with the oxymoron name "Oversecured" since this was their discovery. They explained: "The Google Play Core Library is a popular library for Android that allows updates to various parts of an app to be delivered at runtime without the participation of the user, via the Google API. It can also be used to reduce the size of the main APK file by loading resources optimized for a particular device and settings - localization, image dimensions, processor architecture, dynamic modules, that all sounds really cool - instead of storing dozens of different possible versions."

They said: "The vulnerability we discovered made it possible to add executable modules to any apps using the library, meaning arbitrary code could be executed within them. An attacker who had a malware app installed on the victim's device could steal users' login details, passwords, and financial details, and read their mail."

So again, a well-meaning high-volume app installed on an Apple smartphone, I mean, I'm sorry, Android, on an Android device, it's got this library. The library has a bug that allows any other app in the phone to utilize the bug to cause the well-meaning app to download whatever the bad guys want. So essentially it allows any malicious app to penetrate Android's critical inter-app sandbox which exists solely for the purpose of isolating apps from each other to prevent them from accessing each other's stuff. And although Google knew about this earlier this year, and immediately patched the vulnerability back on April 6th of 2020, apparently not all developers received the memo.

Check Point Research took a look at this just last week and explained what it means. They said: "When we combine popular applications that utilize the Google Play Core Library and the local code execution vulnerability, we can clearly see the risks. If a malicious application exploits this vulnerability, it can gain code execution inside popular applications and have the same access as the vulnerable application. The possibilities are limited only by our creativity."

They said: "Here are just a few examples: Inject code into banking applications to grab credentials, and at the same time have SMS permissions to steal the Two-Factor Authentication codes. Inject code into Enterprise applications to gain access to corporate resources. Inject code into social media applications to spy on the victim and use location access to track the device. Inject code into Instant Messaging apps to grab all messages, and possibly send messages on the victim's behalf."

Anyway, we get the point. It's bad. And in their proof-of-concept demonstration, Check Point used a malicious app to steal a login authentication cookie from an older version of Chrome which was built using the original library. Once in possession of the cookie, of course, now you can do session impersonation; right? The attacker was then able to gain unauthorized access to a victim's Dropbox account. So as I noted earlier, the library was updated back in April, eight months ago.

But last week Check Point identified 14 apps having combined downloads of nearly 850 million that are still vulnerable today, eight months later. Within a few hours of their publishing their report, the developers of some of the named apps had released updates that fixed the vulnerability. It only took them eight months public shaming and some outcry. Check Point analyzed the Google Play Store contents and found that, overall, as of September, so a couple months ago, 13% of all Google Play applications used the Play Store Library. So they were well written. That was the right thing to do. And of those, 8% were still using a vulnerable version of the library.

So, yeah, that's far fewer than all apps. But it turns out that a few of them have massive download counts. Like Microsoft's Edge for Android that was and perhaps still is vulnerable. The specific 14 which were just those that they chose because they were popular and had high download counts, this was Check Point: in the social category, Viber; the travel app, Booking; the business app, Cisco Teams; maps and navigation apps, Yango Pro (Taximeter), and also Moovit; the two dating apps, Grindr and OKCupid; Microsoft's Edge browser; and the two utilities, Xrecorder and PowerDirector. All using, as of last week, vulnerable versions of the library. Several of them have been fixed since. But wow.

All the apps were written to do the right thing, to use Google's recommended library for interfacing with the Play Store after the app had been downloaded and run. But their developers had not kept them updated when critical core flaws were discovered and fixed. That's the mistake that they were caught making. And you can't blame it on anybody. Libraries need to be updated. Of course we know that, what, a few months ago Google found the zero-day in their browser which was being used to leverage the flaw in the widely used font library. So, you know, developers who are packaging, who are bundling libraries, are responsible for making sure that those get updated when necessary. And this just hasn't happened.

So unfortunately, while, yeah, the big names, this will come to their attention, and they'll quickly scurry around. I mean, all you have to do is rebuild the app using the new library. So it's not like it's going to be a heavy lift to get this problem fixed. But 13% of apps in the store are using this thing, and 8% of those. And we know there are many apps in the Google Play Store. So they will probably, many of them, never be updated. They will remain vulnerable and always be creating a known vulnerability for their users. If bad guys happened to know that a targeted individual had one of those apps installed, or maybe even arrange to get one of those apps installed, if they didn't already, it would then open them to vulnerability. So anyway, it's an interesting twist, a widespread collection of apps, themselves containing a common vulnerability as a consequence of once having been built with a library that was later found to be vulnerable. And no one's in a big hurry to fix it.

Oh, one thing that I forgot to mention in my coverage last week of the Tesla Model X key fob hack. A friend of mine brought it to my attention. I thought it was really poignant. This was something that Lennert Wouters was quoted as saying in some other coverage. He said of the Tesla Model X key fob: "The system itself has everything it needs to be secure. But there were just a few small mistakes that allowed me to circumvent all of their security measures." I thought that was neat. So it wasn't at all that they hadn't taken security seriously. They had good security people doing a good job designing the security for the system. But they made a couple little mistakes. And that created a crack that Lennert was able to get in through.

Leo: That'll do it.

Steve: Yup. And Leo, I know you, I mean, you just can't stop talking about the miracle Apple M1 Arm processor chip.

Leo: Oh, man. I just love it. It is, well, the thing that's surprising, and at first I was a little disgruntled because a lot of the things I wanted to use wouldn't work on it. But that's changing very, very quickly. Mostly it's just changing as soon as people get an M1, and then they go, oh, I need to switch this and this and this, and it's working. So, boy, is it fast. It's really remarkable.

Steve: So, and one of the things that people have remarked about is that it is arguably a better processor for running Microsoft Windows than Microsoft's own solutions.

Leo: Yeah. It's easily the speediest Arm chip out there. But most Arm chips up to now have been built for low power. And so this is - they're doing some amazing stuff with this.

Steve: So actually the same friend who brought the Lennert Wouters note to me shared with me using Thread Reader a thread that he found from somebody who was knowledgeable about what Apple did. The thread reads: "In case you were wondering, Apple's replacement for Intel processors turns out to work really, really well. Some otherwise skeptical techies are calling it 'black magic.' It runs Intel code extraordinarily well. The basic reason is that Arm and Intel architectures have converged. Yes, the instruction sets are different. But the underlying architectural issues have become very similar.

"The biggest hurdle was memory ordering, the order in which two CPU cores see modifications in memory made by the other. It's the biggest problem affecting Microsoft's emulation of x86 code on their Arm-based Surface laptops, with the result being high x86 emulation overhead. So Apple simply cheated. They added Intel's memory ordering to their CPU. When running translated x86 code, they switch the mode of the CPU to conform to Intel's memory ordering.

"With those underlying architectural issues eliminated, running x86 code simply means translating those instructions into their Arm equivalent. This is very efficient and results in code that often runs at the same speed. Sometimes there isn't a direct equivalent. So the translation results in slightly slower code. But benchmarks show x86 being consistently at least 70% of the speed. In any case, a surprising number of apps already run. Apple seeded developer systems a few months back, allowing people to get their code ready.

"Normally, that wouldn't have been enough time. When you recompile code for a new architecture, it usually breaks." He says: "But as I said above, Arm and Intel architectures have converged enough that code is much less likely to break, making recompiling easier." And then he finishes: "Apple has made surprising choices. They've optimized JavaScript, with special JavaScript-specific instructions..."

Leo: In hardware.

Steve: "...double-size L1 caches."

Leo: Wow.

Steve: Yes, very smart.

Leo: That's why my browser is so fast.

Steve: Exactly.

Leo: It's amazing.

Steve: He said: "And probably other tricks I don't know of. Thus, as you browse the web, their new laptop will seem faster and last longer on battery because JavaScript has become far more efficient, even though other benchmarks show it roughly the same speed as Intel/AMD." So again, Apple did some very, very clever things.

I also wanted to mention that a few months ago I told all of our listeners about the Windows app InitDisk which I had created along the way to - I needed updated USB prep technology for SpinRite, and I would also be needing that for the ReadSpeed Benchmark to allow people to easily create a bootable thumb drive that they'd be able then to boot their system with in order to use ReadSpeed to check the performance of their drives. Since last week's podcast I finally finished all of the low-level driver and benchmark work, and I produced two release candidates. The second one appears to be final and finished. And actually I just made a tweak this morning since I got the podcast ready, and I was waiting for MacBreak Weekly to finish. So there will be another little tweak. But it's, I mean, basically it's done.

Two people had found two problems with InitDisk, so I fixed both of those. One was a finicky security permissions problem that only manifested under Windows 10. Fixed that without too much work. The other was weird. He was using it on a 1GB USB stick, and InitDisk was selecting 512-byte sectors. I mean, 512-byte clusters, you know, single-sector clusters. That's legal, and it got a lot of testing and never had any problem, but this one guy's BIOS didn't want to boot a USB that had been prepped that way. So I fixed that, too. I set a lower limit of a 4K cluster, which is actually part of the FAT32 spec. So it now does that. That fixed his problem, and his system now boots. So anyway, for anyone who's interested, there is now a release 2 of InitDisk over at GRC, and everybody will be getting the benefit of that shortly when they play with ReadSpeed.

And speaking of ReadSpeed and SpinRite, I am pretty sure that next week's podcast will announce that the long-awaited benchmark is ready for use and testing by everyone. I am really close to it. The DOS-based drivers and benchmarking code is finished. Now I'm working on the packaging of it into a version, essentially a version of InitDisk that doesn't just format and install FreeDOS, but also installs a turnkey "Just Press Enter" ready-to-run benchmark. And actually most of that is already finished, as well. Over the weekend I released a first proof-of-concept Windows app which works. So I'm just in the process of getting it polished up.

But Leo, I wanted - we were talking about the read performance measuring last week. I've got two shortcuts. I'd like if you would to bring up the first one.

Leo: Sure.

Steve: You can just click on the link in the show notes, or it's grc.sc/796.

Leo: Ooh, it's tiny.

Steve: You'll probably need to zoom in on that.

Leo: Need to zoom in, yeah.

Steve: So that first one is the verbose output from the benchmark. A lot of that stuff will only be of use if I'm diagnosing a problem. The second to the last big table shows - and this is on my grueling test machine. I have 11 different drives with all makes and models of controllers and things, in order to give this a real workout. So it shows the PCI bus location of everything, the BIOS's drive identification, then the type of device it is - SATA, IDE, or RAID. A bunch of PCI information. The port that it's on, SATA port or master or slave. Which hardware interrupt line that controller has hooked. And then some addressing information.

And here's one of the cool things. You can see in that first table, the top line? It says, under drive, it says <6.0> with little brackets around it. That's because it's noticed that you've got a 6Gb speed drive hooked to a 3Gb SATA link. Meaning that that drive could potentially perform faster if you stuck it on a SATA III 6Gb link. It displays that information and calls it out sort of subtly there, when you use the verbose mode. But if you just do nothing, and it sees that you do have a mismatched drive speed and SATA link, it specifically spells it out in English for the typical user who's doing that.

Then the last table there is the actual benchmark, where you can see 11 drives enumerated. It shows the size of the drive, the drive's identity - you know, Samsung SSD 860 EVO 250GB. And then there's, for each drive, five different locations, the measured performance of the drive. And again you can see how very stable. The Samsung shows 273.9 at the 0% location of the drive, then 274.0, 274.0, 274.0, 274.0 all the way across the drive. So first of all, the drive itself is very stable. But this benchmark is resolving four significant digits with perfect accuracy. One thing you'll note, if we look at like Drive 82, it's an SSD, but the 0% column, the front of the drive, shows 72.1, this is in megabytes per second; whereas the rest of the locations at the 25, 50, 75, and 100% locations are up at its normal speed of 174.9.

Okay. So now let's look at the second of these links, grc.sc/796a. This is the result of adding a /1 after the ReadSpeed executable. It breaks down the large - in every case we're timing the length of time required to transfer one gigabyte of data. So first, if looking at the drive number on the far left, Leo, if you scroll down to 89, that's just my favorite because it demonstrates how solid this benchmark is. That's 115GB OCZ-Vertex2 drive, Drive 89, if you scroll way down.

Leo: Oh, okay.

Steve: You'll see the, yeah, 81...

Leo: Yeah, yeah, yeah. I just was looking at the single line. Okay, here we go.

Steve: Yeah.

Leo: Yeah.

Steve: Now, look at those numbers.

Leo: Very rock solid, huh. Wow.

Steve: And so this is a solid-state drive, and every single one of them, with the exception of the very first one, but the other at 64 times five columns, 275.8 MBps. So just solid. The really interesting thing is if you scroll up and look at 82. Here's an example. This is an SSD. This is the one that Allyn and I were talking about in our email correspondence, where you can see, whoops, now, here's a problem. This thing...

Leo: Look at this, as low as six.

Steve: Yeah. It's nominally - and even 3.4 and 3.6.

Leo: Geez.

Steve: So what's interesting to both of us, and will be to our listeners because, I mean, you'll be able to look at all of your drives like this.

Leo: Can't wait.

Steve: Like something has brought this thing to a near standstill. And in fact there is of course a progress bar on the screen as the benchmark is running. And so you see the bar moving along, and then it just stops. But it doesn't produce an error. It just stops. And it takes a long time, and then it says, oh, and then off it goes again. So what we conjecture is, as I mentioned last week, that clearly this is a spot that this drive is having a serious problem reading. Like nothing else, nowhere else on the drive is there this problem.

So the thing that is tantalizing about this is to see whether rewriting that location will, you know, it might induce the SSD to remove it from service when the SSD has the opportunity to. Allyn noted that SSDs do not do their own rewriting, unlike hard drives that do. So it needs stimulation in order to do that. Or is the problem that the bits in a multilevel cell have drifted a little bit, "charge drift" as Allyn termed it, so that it's necessary to like put much more effort into reading them, and does that suggest that if they're allowed to drift even further, then it could become uncorrectable. Or does this suggest that this is not a very high-end SSD?

And so they just didn't expend any hardware on error correction. And so instead it's dropping into an algorithmic error correction figuring, well, the guy would rather wait than have, like, no data returned. So we're going to do ECC. Anyway, this does create a bunch of questions. But it certainly does say, first of all, that this is a microscope the likes of which we've never had before into the detailed performance of something that we just take for granted, you know, solid-state drives. And it suggests that a future SpinRite is going to be able perhaps to do a lot of preventative maintenance.

And as I mentioned last week, if you look at some of those numbers, 544 MBps is what this thing is obtaining in sustained transfer, which means that a half a terabyte drive, a 500GB drive can be scanned in a little over 15 minutes. So that's about twice the performance I was expecting. That means that 1TB in about half an hour, or 2TB an hour. So this thing suddenly becomes practical to use as a preventative maintenance tool. So anyway, very excited, and I expect to be announcing it next week.

Leo: Sidewalk. I'm really glad you decided to go deep on this. I'm fascinated. And I don't know what to think, to be honest.

Steve: Exactly. And believe me, the press is in the same plight. As I mentioned at the top of the show, Amazon has created quite a stir by sending letters to their customers who own compatible equipment. My buddy is like Echo happy. He's got them all over the place. And the letter informs them that they will be automatically opted in to Amazon's forthcoming Sidewalk program.

Now, for anyone objecting to auto opt-in, you know, as we were talking about it before, I get it, I mean, like, why it makes sense for Amazon to do this. But we know our listeners are special. If you update today, your smartphone's app, even before Sidewalk goes live, you may preemptively disable your network's use of Sidewalk. So my point is that - I just said the "A" word, I realize. Hope I didn't trigger everyone's devices. So it should be already there in Amazon's app for their devices, and you should be able to disable Sidewalk preemptively.

Okay. In any event, their announcement has been met with no small degree of questioning, confusion, and concern over the propriety and the security of what they're doing. The tech press has been carrying stories with headlines like, well, in fact, this is Fox Business said: "Amazon to opt-in customers with Echo, Ring devices to new Sidewalk WiFi sharing feature." And as we'll see, that's not quite true since it's not WiFi sharing. But everyone here will know everything about it by the end of the podcast.

Leo: It's LoRa. LoRa.

Steve: Mashable was taking no prisoners. They said: "How to see if Amazon is stealing your Internet bandwidth for Sidewalk." WGN-TV said: "Got an Echo or Ring? Soon Amazon will use them to share your Internet with a new Sidewalk network unless you opt out."

Leo: Yeah, I think a lot of this is misleading.

Steve: CNET: "Amazon Sidewalk has automatically switched on in your Alexa app. It might be time to check your settings, if you have an Echo smart speaker or Ring camera." TechHive: "Welcome to Amazon Sidewalk. Now, here's how to turn it off." And Gizmodo was the least charitable: "You Need to Opt Out of Amazon Sidewalk." That was the headline of their story. Then they said: "Have you heard of Amazon Sidewalk? Probably not. But there is a good chance that you or someone you know has an Amazon Echo or Ring camera. And if you own one of those devices and live in the U.S., or know someone who does, you need to tell them to opt out of the service as soon as possible."

So for what it's worth, I replied to my best friend, and I said, "You know, Mark, I don't think you need to turn it off." And we'll see, of course, that's a matter of personal taste, but I will tell everybody exactly what this is about.

Okay. So taking a deep breath, I have the 13-page Amazon Sidewalk Privacy and Security Whitepaper. So this answers all the questions, at least in terms of design and intent. As we know, you can't answer questions about execution. It'll take security researchers and academics poking at it and analyzing it more deeply to do that, and time. But let's begin by looking at Amazon's description and their hopes for what this is and why they think it's a good idea to do this. And this is not brand new. The news is that it's about to be unveiled. But this description is dated more than a year ago, on September 27th, 2019.

Okay. They said: "Get connected convenience beyond your front door. Many of the smart devices in our homes today rely on Bluetooth and WiFi connections to stream music to a nearby speaker or help a video doorbell notify us when a package is delivered. But these connections only extend so far. On the other end of the spectrum, 5G cellular is incredibly important when you need reliable, long distance, guaranteed delivery of data, but it can be complex. In the space around homes, that leaves a middle ground for devices like sensors and smart lights that can benefit from low-cost, low-power, low-bandwidth connections." So they're sort of setting this up as something between local WiFi and 5G, or whatever cellular.

They said: "Customers shouldn't have to settle for connected devices that lose functionality past the front door, which is why we're excited to introduce Amazon Sidewalk. Amazon Sidewalk is a new long-term effort to greatly extend the working range of low-bandwidth, low-power, smart lights, sensors, and other low-cost devices customers install at the edge of their home network. Using the 900 MHz spectrum" - okay, so there's a big difference right there. This is not a WiFi frequency. Ham radio operators like you, Leo, know this as the "33-centimeter band." Anyway: "Using the 900 MHz spectrum, we are developing a new protocol we project can increase the connection range of these devices by more than one half mile/one kilometer."

Leo: That's why you don't use WiFi. It's too high-frequency. It stops, 150 feet, yeah.

Steve: Correct, correct. And 900 MHz has really good building penetration.

Leo: Absolutely, yeah, yeah, yeah.

Steve: But at a cost of bandwidth. If the carrier is a lower frequency, you can't modulate it that quickly. And in fact, there is an incredibly clever modulation scheme known as LoRa we'll get to in a second which uses chirps, chirping up or chirping down, which, well, anyway. We'll get there.

They said: "With Amazon Sidewalk, customers will be able to place smart devices anywhere on their property and know they'll work great, even in dead spots where WiFi and Bluetooth won't reach. Using the 900 MHz spectrum to help devices communicate is not new. In fact, it's been around for decades, providing reliable secure connections for long-range devices like the radios used by emergency services and the digital pagers carried by doctors on call. It's by combining this tested communications network with an innovative new protocol developed by Amazon that we arrived at Sidewalk, a new way for the next generation of low-cost, low-bandwidth sensors and smart devices to work together to create a secure network of long-distance connections bridging the connectivity gaps around our homes.

"The immediate benefit of a 900 MHz-based network is the ability to use your favorite connected devices even if they're located far away from the router inside your home. Today, Ring Smart Lighting Bridges use connections in this spectrum to extend the range of smart lighting products, and soon additional devices including the latest generation

Ring Floodlight Camera and Ring Spotlight Camera will also help customers extend the network connections around their homes and control those 900 MHz devices at much greater distances.

"Better network connectivity can also help keep devices safe and up to date. Today, when customers place a smart device at the edge of their home network, poor network connectivity can prevent that device from receiving important feature and security updates. By extending long-range, low-bandwidth connections using the Amazon Sidewalk network, customers won't have to worry about smart devices that don't have access to the latest security updates or work as intended because they're out of network range.

"In the near future, we also see the potential to help customers get more from 900 MHz connections in their neighborhoods, creating a broad network among neighbors that can be used to extend connectivity all the way to your mailbox out at the street where a smart sensor lets you know exactly when your mail has been delivered, or to a water sensor that lets you know it's time to water the garden in the backyard.

"For example," they said, "just a week ago" - now, this is a week ago a year ago - "Amazon employees and their friends and family joined together to conduct a test using 700 Ring lighting products which support 900 MHz connections. Employees installed these devices around their home as typical customers do; and, in just days, these individual network points combined to support a secure" - and we'll get to security because of course that's super important, and I've got it nailed here in the podcast - "to create a secure low-bandwidth 900 MHz network for things like lights and sensors that covered much of the Los Angeles Basin, one of the largest metropolitan regions in the United States by land area.

"This neighbor-created network demonstrates the potential of Amazon Sidewalk - a broad coverage network, great for low-bandwidth, low-cost devices that require no complex setup or maintenance for customers. But the benefits don't stop there. With Sidewalk, we also see the opportunity to deliver new devices and experiences that delight our customers."

They said: "As one example, this week we announced" - and this would have been a year ago - "Fetch, a compact, lightweight device that will clip to your pet's collar and help ensure they're safe. If your dog wanders outside a perimeter you've set using the Ring app, Fetch will let you know. In the future, expanding the Amazon Sidewalk network will provide customers with even more capabilities like real-time location information, helping you quickly reunite with a lost pet. For device makers, Fetch also serves as a reference design to demonstrate the potential that devices connected to a broad, reliable network can provide to their customers.

"Extending the convenience of a long-range network will take time, but we're already working quickly to bring this future to life for customers. For device makers, we plan to publish protocols that any manufacturer can use to build reliable, low-power, low-cost devices that benefit from access to long-range, low-bandwidth wireless connections. In the meantime, you can sign up to be notified when more information is available." And they finish: "Amazon Sidewalk is a long-term effort, but we're excited to get started and can't wait to see what device makers build and how customers benefit. The possibilities are endless."

Okay. So clearly what they are planning and are in the process of bringing is a new radio which will be added to all of their Amazon devices. And that new radio will be 900 MHz, the 33-centimeter unlicensed amateur radio band. It also works with Bluetooth Low Energy, alternatively. Now, of course Amazon has been selling devices for quite a while. Today, all Echoes except for the first generation, so the second and third and fourth gens in 2017, 2019, and 2020 are all Sidewalk-compatible. This year's fourth-gen Echo also has the 900 MHz radio. All of the Echo Dots are Sidewalk compatible, but none of them have the 900 MHz radio. So they're limited to using Bluetooth Low Energy. The same is true for the Pluses. There are five Echo Shows, and this year's Echo Show 10 is 900 MHz-capable. And then those two devices they mention, the Floodlight Cam and the Spotlight Cam, those are both last year's devices, and they already have the 900 MHz radio.

So in the show notes I've got a link to this whitepaper which I've read and digested and understand. In there they make a couple of points that I'll share. They said: "A simple control is provided to enable and disable participation in the neighborhood network. When customers first turn on a new Sidewalk gateway device, they will be asked whether they want to join the network. For customers with existing devices that are Sidewalk-capable, an over-the-air update will connect them to the network. No action is needed. These customers will first receive an email about the pending update and instructions for how to disable, if that is their choice."

So that's how, as I said at the top, that's how Amazon sort of walked this fine line of really wanting this network to take off, but recognizing that they are using the Internet connectivity of their customers in order to create, they hope, over time a true 900 MHz Wide Area Network which can do all kinds of stuff, like fitting into a gap that exists today. And I think, for example, of the person who has a long drive from their home out to the end of their driveway where they have an automated gate, and they would like access to it, but WiFi won't reach. This technology would be perfect for that. And it might even be that the gate is using the Amazon Sidewalk bandwidth of the house across the street, which is closer than their own. And I'll explain how this works and why it can be safe.

Amazon also said: "As a crowd-sourced community benefit, Amazon Sidewalk is only as powerful as the trust our customers place in us to safeguard customer data. To that end, this document outlines the steps we have taken to secure the network and maintain customer privacy. These efforts are core to our mission and will continue to evolve and improve over time." And, finally: "The maximum bandwidth of a Sidewalk Bridge" - and I'll explain some of these terminologies. There's five new definitions we have. "The maximum bandwidth of a Sidewalk Bridge to the Sidewalk server is 80Kbps," they say, "which is about one 40th of the bandwidth used to stream a typical high-definition video. Today, total monthly data used by Sidewalk-enabled devices, per customer, is capped at 500MB, which is equivalent to streaming about 10 minutes of a high-definition video."

So, I mean, the whole point of this, you know, nobody is going to use your Internet connection out literally on your sidewalk to download something. That's not what this is. You don't have, this is not an Internet protocol extension. There's no IP, Internet Protocol, on this 900 MHz. It is a message-passing, signaling level network. So the overall network can be visualized by, well, can be visualized as consisting of five things. There are gateways like the Ring Floodlight. There are endpoints like a humidity sensor in your backyard. There's the Amazon-operated, what they call the "Sidewalk Network Server." And that's distinct from, and I'll explain how, from application servers which are the things that the endpoints communicate to. And then there are message packets, which is the medium of communication.

So the gateways, which they also call Sidewalk Bridges because it is a bridge from either your WiFi or your wired LAN to this participation in the 900 MHz network, the bridges forward packets to and from the Sidewalk Endpoints and through your LAN connect to the Sidewalk Network Server which is Amazon's device. The gateways right now are Amazon devices, like the Ring floodlight cam, that use this 900 MHz band, or Bluetooth Low Energy, to provide connection to the Sidewalk network. At 900 MHz it either uses this LoRa (L-O-R-A) modulation or simple FSK (Frequency Shift Keying).

And LoRa is a very clever technology, which as I mentioned it uses bidirectional frequency chirps. So it's inherently broad spectrum because a chirp is. But it allows it to obtain extremely good range at very low bandwidth. It makes good use of receiver sensitivity. And because the carrier is actually chirping, it solves the problem of particular carrier frequencies being blocked, just naturally blocked by some substances. You know, if something were in the way, which happened to be absorbing some particular band or a spike because it happened to be resonant at that frequency, it would absorb the energy. But the chirp spans that so the chirp still gets through. Anyway, the point is this is a very different technology meant for a very different application.

Okay. So we have the gateways. The endpoints, which they also call "Sidewalk-enabled devices," may be known as edge devices, endpoints, or applications. They're able to roam around on the Sidewalk network by connecting to Sidewalk gateways, whether your own or somebody else's. The system is completely agnostic about whose gateway you're connecting to, and it makes that secure. The endpoints are low-bandwidth, low-power smart products like leak sensors, door locks, lights, or devices attachable maybe to valuable things like luggage tags or a pet which is wandering around. The endpoints can be built and maintained by Amazon or by third-party developers, so the system is open.

The gateways can also act as an endpoint themselves and receive Sidewalk benefits like maintaining functionality when the device falls offline. And that's interesting. For example, it would mean that if your own router and cable modem froze, your Sidewalk-enabled IoT devices like lighting or your door lock would normally fall offline. But this allows them to remain connected, thanks to a neighborhood-wide active Sidewalk network that would allow it to automatically ride over someone else's bandwidth - again, not exchanging lots of data, but down at the message-passing level.

We have the Sidewalk Network Server, which is Amazon's. It's responsible for verifying that the incoming packets, and I'll explain this in a second, are coming from authorized Sidewalk devices, routing packets to the desired destination, which is an application server, or in the other direction to an endpoint or a gateway. And it also keeps the network time synced. Time is an important component here, as we'll see. They have very cleverly used time to cryptographically rotate all of the IDs of all of the devices, exactly analogous to the six-digit PINs that we're all used to now with our one-time passwords. So everything needs to know what time it is. But given an agreement about time, this is how tracking is prevented over time. But anyway, I'll explain that in a second.

Finally, the Application Servers are different from Amazon's what they call the Sidewalk Network Server. The Sidewalk Network Server is the protocol endpoint, but that then forwards the packets, routes them to the Application Servers. So, for example, the company that provided the moisture sensor in the backyard or the leak detector, it would have its own app and its own server, which today we are connecting to, if you have WiFi. In the future it would be able to work over Sidewalk.

So the Application Servers are managed by the endpoint manufacturer, which could be Amazon or some third party. So, for example, say that the garage door opener manufacturer Genie were to create a smart Sidewalk-enabled garage door opener. It would normally be connected to your home WiFi, and it would normally be offering Sidewalk connectivity to your neighborhood. But reciprocally, it would also be able to use the neighborhood's Sidewalk network if, for example, your WiFi was not available. So if you needed to reach it while your home LAN was down and you weren't, you know, your router hung or your cable modem froze or something, the Genie Application Server would route through Amazon's Sidewalk Network Server to reach your Sidewalk-enabled garage door opener via a neighbor's Internet connection. And all of this is transparent. And the last component are Packets, also known as Messages, which are the things exchanged between the Endpoints and the Application Server going both directions through the Gateways and Amazon Sidewalk Network Server.

So that's the architecture. Looking at it, the network's design reveals that Amazon has put a great deal of time, attention, and design work into creating a system that provides the security controls that Amazon requires for the network to operate safely while also blinding Amazon to all of the network's messaging traffic. Amazon can see nothing about the messaging level.

Our listeners know well about the Onion Router network, this concept of - it's termed an "onion" because it's consecutive shells of encryption. Well, that exactly mimics the design of this network. Sidewalk uses three layered wrappings of encryption. The innermost encryption is the application layer, which protects the privacy and security of the communications between the endpoint, like out in your backyard, and the application server which needs to talk to that device.

So this is the layer that does the actual signaling work, the message passing. And it is end-to-end encrypted using the state-of-the-art means that we know of to do that today. So that creates an encrypted tunnel between the far extremes. The application layer encryption is then in turn encrypted, also at the endpoint. So the endpoint encrypts first for the application server. Then it encrypts that for the Amazon Network Server. This conceals and protects the Sidewalk packet as it's moving over the air. And the plaintext data encrypted by this layer is accessible only to the endpoint and the Amazon Network Server, nothing in between.

And then, finally, what they call the "Flex Layer" is added at the gateway device. So the endpoint encrypts twice, once with a key known only to the application server. Then it encrypts that with a key known only to Amazon's Network Server. That goes over 900 MHz to the Sidewalk gateway device. It encrypts that for the Amazon Network Server using a trusted and tamperproof reference for message-received time and adds an additional layer of confidentiality. That's then what it transmits, either over your wired LAN or WiFi, to Amazon's Network Server.

So as I noted above, ultimately the communication is between the endpoint devices and their application servers, with the Sidewalk gateway devices and Amazon's Network Server functioning as intermediaries. So consequently the innermost wrapper of encryption is end to end between endpoint device and the device's matching application server. Neither the gateway that facilitates the communication at the neighborhood end nor the Amazon Network Server that facilitates the communication over the Internet are able to see anything about what's being transacted.

And looks like I quoted from Amazon's document. They said: "Amazon has carefully designed privacy protections into how Sidewalk collects, stores, and uses metadata. Sidewalk protects customer privacy by limiting the amount and type of metadata that Amazon needs to receive from Sidewalk endpoints to manage the network. For example, Sidewalk needs to know the endpoint's Sidewalk ID to authenticate the endpoint before allowing the gateway to route the endpoint's packets on the network. Sidewalk also tracks a gateway's usage to ensure bandwidth caps are not exceeded and latency is minimized over a customer's private network."

They said: "Information customers would deem sensitive, like the contents of a packet sent over the Sidewalk network, is not seen by Sidewalk. Only the intended destinations, the endpoint and application server, possess the keys required to access this information. Sidewalk's design also ensures that owners of Sidewalk gateways do not have access to the contents of the packet from endpoints," whether or not they own those endpoints which may be using their bandwidth.

"Similarly, endpoint owners do not have access to gateway information. The Sidewalk Network Server" - that's Amazon's - "continuously 'rolls,' or changes transmission IDs" - they call them "TX-IDs" - "and Sidewalk Gateway IDs every 15 minutes to prevent tracking devices and associating a device to a specific user. The IDs use a time-based cryptographic system like our TOTPs so that the endpoints are continuously and autonomously reidentifying themselves using a periodically changing ID, and the Amazon server shares the underlying key and thus can determine who's who. But no one monitoring the metadata could determine whether the same or some other device was communicating" from one series of events to another.

From the view of the endpoint, the device using someone's gateway device, it's only able to view information that pertains to the normal operation of its device, whether the smart light is on or off. It's unable to see routing information or even what gateway, for example, if it's not owned by its owner, the smart light is receiving support from, nor any information about that gateway and the gateway's owner. The gateway information is encrypted behind the Sidewalk network layer and the flex layer. So again, it's a well-designed system of deliberate blinding layers so that only the information needed is visible. Everything else is encrypted in a lower level layer. And the thing down at the low level has no awareness of what's going on at the higher level.

From the viewpoint of the gateway device, it is unable to see what the endpoints, whether or not they're owned, are receiving from their gateway. They have no idea what types of endpoints are connected, nor the times in which they are connected, or information about the owner of the endpoint. All of that information is encrypted as it passes by the Sidewalk Application Layer. At the far end, the application server is unable to see any information pertaining to the gateway owner because that's been stripped by Amazon's Sidewalk Network Server. It only has access to the endpoint information, since those outer wrappers and metadata, like the gateway ID, will have been removed by Amazon's Network Server.

And as we would hope, the registration time establishment of unique identifying credentials assure that only trusted and known devices can enter the Sidewalk network, which prevents unauthorized devices from joining. The Sidewalk Network Server, the Application Server, and each Sidewalk device, both the gateways and the endpoints, are provisioned with a unique set of Sidewalk credentials that are used during the Sidewalk device registration process to mutually authenticate each device's identity and to derive unique session keys for use between them. Rolling encryption keys are periodically derived from their respective session keys.

Amazon also noted that to protect their customers' privacy, the routing data that they were necessarily using to link the location of a known endpoint device to perhaps someone else's gateway by network but also probably by geographic location is deliberately wiped and discarded on a rolling 24-hour basis. So it's only retained for a day.

So that's the system. It's not neighborhood WiFi. It's an encrypted IoT low-speed communications signaling solution. It's initially primarily Bluetooth Low Energy, since all of Amazon's various devices have that. And we know that's, what, 30 feet maybe, 10 meters, so not a great distance. But over time all new Amazon devices will certainly include this newer 900 MHz radio that will really start to give the system some useful range. And as I said, while it certainly will need to survive a deeper analysis by crypto people and academic analysts, it's clear that they really thought this through. They worked hard to create and deliver a state-of-the-art secure messaging solution. And if it were to succeed, we might be in a world where a cool low-frequency, low-bandwidth, low data rate message passing network was pervasive. And it would allow things that right now have a hard time staying connected to be connected. So I think it's cool. That's what Sidewalk is.

Leo: We're going to rely on you to keep an eye on this and kind of fill us in on the risks because of course it's really unfortunate. Mashable's not alone, Business Insider, every publication, looking for traffic - I want to underscore that - looking for traffic and saying, oh, my god, it's opt-out. And, I mean, it could be bad. I think there's a kind of a kneejerk reaction, "Not with my bandwidth, you don't." But it's not - it's a tiny amount of bandwidth. And you made a really good point. You're also getting bandwidth from your neighbors. So it probably is a zero-sum, I mean, it's all going to balance out if you use it. And, you know, if you've got a dog, and you get one of these collars, you'll use it.

I think there are a lot of potential uses for this. And it'd be really a shame for us to lose the benefit of Sidewalk because of imagined or overblown privacy concerns. If there are some - and that's why we're going to count on you. Because if there are some, or security issues, we want to know about it. I know you're going to keep an eye on it. But if there aren't, I don't think, just because Mashable wants to get viewership, we should assume that when they say it's a privacy nightmare, that it really is. And sounds like it's not. Sounds like it's well designed.

Steve: I completely agree. I think it is as well - one of the things that we're seeing is, and I've said this so many times, all of these problems are solved. We now know how to do secure end-to-end encryption. We now know how to rotate keys based on time, using a crypto algorithm. We know how to do all of this. And so the idea of successive layers of encryption, where at each layer only the information that's necessary is available to the layer above and not below, I mean, it's clearly designed right. Whether it's implemented right, whether dumb third parties create bad implementations, I mean, you know, there are some things you can't help.

Leo: Always possible. They've done that with WiFi, you know. I mean, that doesn't mean WiFi's a bad thing.

Steve: Right.

Leo: So, yeah, and this is why people have to listen to this show because we know that you will give us a reasoned - you're not link baiting. You're the opposite. You're going to give us a reasoned look at this. And, I mean, at first it sounds a little sketch, potentially. But it sounds like they've done the right thing. And this is my problem with just kind of this blanket privacy thing is you could lose some real benefits. If there is no risk, then let's go for the rewards.

Steve: Well, Leo, had the privacy people been involved in the design of the Internet, we would still be waiting for it.

Leo: That's a good point. We wouldn't have email, that's for sure. No websites.

Steve: It would have never happened.

Leo: Yeah, no, that's for sure.

Steve: It would have never gotten off the ground.

Leo: This is actually a little different because it's been designed from upfront to be secure and private, as opposed to the Internet, which was not designed that way by any means.

Steve: Yeah, you wouldn't be able to drive because there would be taut string connecting paper cups all over the world.

Leo: Yeah. But we'll keep an eye on it. You know, I've been going back and forth. Should I turn it off? Should I leave it on? And I think the problem is it only works to the degree that everybody leaves it on.

Steve: That's right.

Leo: If nobody uses it, then it's not going to do anything, and then it's another technology just down the tubes.

Steve: It's like the COVID contact tracing.

Leo: Yeah.

Steve: You know, it's like, you know...

Leo: Right, right.

Steve: Yeah, I see nothing wrong with it. The design is solid. They thought this thing through. And it may be the kind of thing, I mean, so they're going to end up seeding the world with these devices.

Leo: Yeah.

Steve: They will exist. And so they could urge people later to please reconsider, maybe give you a discount on your Amazon Prime if you turn it on, because they'll know if you've turned it on. I mean, there's all kinds of things that could happen. But it can't happen unless the protocol is designed and designed well, if they can demonstrate that they are really not misusing people's bandwidth, and if the radio hardware is in place. Well, the radio hardware is going to be in place. And so I think...

Leo: Yeah, it's amazing, isn't it, actually.

Steve: ...once that happens, you know, we then might start to see some compelling use case for it. I mean, it is sort of a chicken-and-egg thing. You and I remember when people were saying, well, the web doesn't make sense because there's no...

Leo: There's nothing on it. No one's on it.

Steve: Nothing there yet. And then people were saying, yeah, and if nothing there, then one's going to go there. So no one's ever going to put anything there if there are no people to look at it. It's like, yeah, well, it happened anyway. Just like chickens. We got chickens.

Leo: It's actually a very interesting play that only Amazon can make. Although I should point out LoRa goes well beyond Amazon. It's been around for a while. In fact, it's a French technology.

Steve: Yes.

Leo: So others could potentially implement it. But who else has this many IoT devices spread out all over the country?

Steve: Right.

Leo: Thank you, Steve. That's Steve Gibson. You know him from GRC.com. SpinRite, you ever hear of that? Yeah, the world's finest hard drive maintenance and recovery utility, now for more than just spinning drives. I'm really excited now that I know SSDs are going to benefit so much. I can't wait to try that benchmark on the new Mac. So let me know the minute it's available. Do I have to sign up to get in some beta program to do it?

Steve: No, no, no. Everybody will be able to get it next week.

Leo: So exciting. We'll talk about it next week. If you go to GRC.com, get SpinRite. But then you can also get copies of this podcast. Steve has a couple of unique versions, a 16Kb audio version, really tiny, shrunk down. There's an even smaller human-written transcription so you can read along as you listen. That's all at GRC.com, along with a 64Kb audio. We have 64Kb audio and video available at our site, TWiT.tv/sn. There's also a YouTube channel for Security Now!. Every show is up there. You can subscribe in your favorite podcast client. It's been around for a few years, so you should be able to find it. Just search for Security Now! and subscribe, and that way you'll get it automatically.

Steve is on the Twitter at @SGgrc. That's the place you can DM him. His DMs are open. If you have a thought, a suggestion, a tip, a leak, a funny picture for his Picture of the Week, you can tweet him. You can also go to the feedback form at GRC.com/feedback. And if you get a copy of SpinRite now, you'll be first in line to get a copy of 6.1 when it comes out. That'll be a free upgrade for you. So another good reason to get it.

I have to get my serial number. I have to call your guy and get my serial number so I can update. Thank you, Steve. We'll see you next week on Security Now!.

Steve: Thanks, buddy.

Copyright (c) 2014 by Steve Gibson and Leo Laporte. SOME RIGHTS RESERVED

This work is licensed for the good of the Internet Community under the Creative Commons License v2.5. See the following Web page for details: http://creativecommons.org/licenses/by-nc-sa/2.5/