SECURITY NOW!

Transcript of Episode #545

Three Dumb Routers

Description: Steve and Leo catch up with the past week's small amount of security news, then they talk a bit about Steve's discovery of a rare and wonderful true EEG sleep monitor and various other miscellany. Then Steve digs deep into home consumer router operation to explain why no fewer than "three dumb routers" are required for full, true, securely isolated network operation.

High quality (64 kbps) mp3 audio file URL: http://media.GRC.com/sn/SN-545.mp3
Quarter size (16 kbps) mp3 audio file URL: http://media.GRC.com/sn/sn-545-lq.mp3

SHOW TEASE: It's time for Security Now!. Steve Gibson's here. Once again we talk about the Internet of Things and how to secure your home from all these little doohickeys that are online. Turns out Steve's got the best way to do it. We've mentioned guest networks before. We've mentioned a two-router solution. But the real robust way to do it, three dumb routers. Stay tuned. Security Now! is next.

Leo Laporte: This is Security Now! with Steve Gibson, Episode 545, recorded Tuesday, February 2nd, 2016: Three Dumb Routers.

It's time for Security Now!, a show that in this case is very aptly named, two dumb routers, plus a third, just you. No, no. Steve Gibson is here. We are going to talk about routers in just a little bit. He is the guy at GRC.com, the creator of SpinRite, the man who does the best job ever of explaining, not just security, but technology in general. And now, 10 years in, people are loving this show, I know.

And I hear from people all the time. Who did I just get an email from? Oh, I know what it was. I just read a post, "My Favorite Podcasts." And the fellow who wrote the post is on Medium. What did he write? I've got to find the quote because he wrote a very nice thing about Security Now! and how geeky the show is.

Steve Gibson: Oh, and believe me, we will not disappoint him this week.

Leo: So, Steve-o, hello. What's on the...

Steve: Yo, my friend. So we did not have a lot of news, which is just as well. That and the Iowa Caucus. I knew I wasn't going to get, you know, because I'm a complete political junkie. And so yesterday was - there was no way I could be producing a 20-page podcast of notes beforehand.

Leo: Wait a minute. What were you doing? I mean, what were you watching? You were watching Iowans gather and...

Steve: I was watching slow counts of delegates.

Leo: Very slow. Very slow.

Steve: Come in to see whether Bernie was going to actually get ahead of Hillary, or whether she was going to eke out a...

Leo: You do take this seriously, if you take Iowa seriously. Did you see our podcast, our Triangulation yesterday?

Steve: No.

Leo: Oh, you should see it. The guy was the, for four years, advisor to then-Secretary of State Clinton on innovation and technology. And he's a Silicon Valley guy. And I asked him to give us kind of an inside-the-beltway view of what it looks like to Washington wonks...

Steve: Neat.

Leo: ...when the real world, you know, and so forth. It was really very interesting stuff. But we both concurred, Iowa is the craziest way to start this campaign season. It all comes down to Super Tuesday. There are so many delegates then. And until then, it's not really that important.

Steve: No, and in fact for me as a junkie - it's funny, too, because I was thinking about your comment about my enthusiasm for various TV series that I watch. And last week you said, wow, you know, you watch a lot of television. And I thought, you know, Leo, how many hours of sports do you have on your television? Because I have zero.

Leo: I don't watch that much sports. I watch, maybe I watch an NFL game for 16 weeks a year, once a week. No, I'm not a big sports fanatic. But I do, I watch - no, hey, I'm not critical because I watch a lot of TV, too. Lately I've been playing more Minecraft. But that's another story for another day.

Steve: You have been, really?

Leo: Oh, yeah.

Steve: Because I know that for a long time you haven't, like, understood what that was all about, and I've watched you...

Leo: I didn't get it.

Steve: ...the last few weeks beginning to sort of, like, understand the whole Minecraft phenomenon.

Leo: Well, Lisa's son Michael, who's 13, has been a Minecraft buff for, like, five years, since it really came out, and is quite adept at it, really adept at it. And he's been begging me to create a Minecraft server in our house. Actually, it ties in very well to the subject of the show today because I've always said to him, "Oh, no, no, it's too dangerous to run our own server. You can rent Minecraft servers. There's public servers. Just use those."

But finally I got a Raspberry Pi, and I was trying to think, what could I do with this Raspberry Pi? And I thought, you know, this actually would be - it's just a little Linux, $35 Linux computer. This would be a great Minecraft server. So I set it up. And it was so easy that I said, gosh. But you can only get five people in at a time because it's not very powerful. In fact it kind of lags.

Steve: And so is that the reason for having a server is that he would then be able to host his own groups?

Leo: His own clubhouse, yeah.

Steve: Ah.

Leo: Servers, the nice thing about Minecraft is the work you do is persistent. So it creates a world which is then "the world," and you and your - and it's infinite. It's huge. Not completely infinite, but I think it's limited to 30TB of data. So it's functionally infinite. And it's persistent. So if you build a house and come back tomorrow, the house is still there. So he and his friends come in, it's like their clubhouse. And now with kids today they don't really get to go play, right, go out in the street and play. You can't do that anymore.

Steve: No, no, you might, like, actually...

Leo: Stranger danger.

Steve: You might skin your knee.

Leo: Yeah. So, and he's kind of inclined to sit in front of the computer anyway. So I thought this would be great, and it night be a way for us to bond a little bit. So we set up this server. Then I realized, I need a more powerful computer. And I had that Mac Pro in a corner. So it's running on a very powerful computer now.

Steve: Yeah.

Leo: In fact, so much so that I've put two more servers because then I thought, well, that was easy. Maybe I - I wonder if I could find the old TWiT Minecraft server. And it turned out...

Steve: So is there a Minecraft server for Mac? Is that, like, [crosstalk]?

Leo: Yeah. And I'm running a third - oh, there's many. There's the official...

Steve: And what was Minecraft written in?

Leo: Java. That's why. Java.

Steve: That's what I thought, yeah.

Leo: So it runs everywhere.

Steve: I remembered there was something about it, yeah.

Leo: It runs everywhere. And the server, because it's not a GUI, is really easy to be portable; right? So it's Linux, Windows, OS X, whatever.

Steve: Well, we will, in this podcast, as you said, it's perfect because I will explain to you why, if the Minecraft server was known by bad guys to have a flaw, that would allow them...

Leo: Right, right.

Steve: ...to completely get into and take over your home network.

Leo: See, that's my concern. Because of course I'm using port forwarding. I don't DMZ the computer.

Steve: Correct. Doesn't help you.

Leo: It's one single port, UDP, TCP/IP.

Steve: Doesn't help you.

Leo: They go in. But you're right, if there's a flaw in the server - and by the way, there probably is, it's just...

Steve: How could there not be? Look at the stuff we talk about, like OpenSSL.

Leo: Right. And this is just some kids wrote it; right? I'm using a third-party server.

Steve: Yeah, exactly.

Leo: Yeah, yeah. So, good, I would, A, it behooves one to keep it up to date; but, B, I'd like to know how to make it a little bit safer.

Steve: Okay. So today's podcast is titled Three Dumb Routers. And not Three Blind Mice, Three Dumb Routers. First I titled it Router Topology. I thought, well, that's just sort of dry and boring. So, and Three Dumb Routers is actually a much better title because what I'm going to explain, and this is going to be a deep in the weeds, we're going to be talking about ARP broadcasts and IP-to-Mac address resolution. I want to explain why no two-router solution can work.

Last week I suggested that the IoT devices be put on what I would call the "interior router," the router inside your main router. And I got a whole bunch of people saying, "Oh, Gibson, you got that backwards." Well, yeah, I understand that. Ten years ago - and Leo, in fact, if you go to - there's a link to it at the very end of the show notes, but also just in the GRC menu it's under Research > General> NAT Router Security, I think I called it.

Ten years ago, in 2006, in August, I first presented this idea of daisy-chaining routers, chaining them together. And I drew a picture of a router, sort of like one-way valve where stuff could go out, but it couldn't come back in. And on those pages I did put the high-security network on the inside. The problem is, that's not secure, either. So my point is that no two-router solution...

Leo: Yup.

Steve: There's the picture. So I sort of showed it as a valve with a flap, where stuff could go out, but it couldn't come in. And if you scroll down about halfway, you'll see a diagram with two routers, further down, a little more, down, down, down, further, yup, there it is. So there's the super-secure LAN on the internal NAT and the semi-secure LAN sort of in between the two. And the problem is that's got problems. So it really doesn't matter which way you put them. And I was trying to compromise, and I shouldn't have.

So this week is the zero-compromise, this is the way you do it if you have dumb routers and you just want absolute security, absolute network isolation between an Internet of Things network and your regular home network. There are all kinds of ways to do this with fancy routers, with, like, pfSense and firewalls and rules and so forth. You really probably want to use VLANs, Virtual Local Area Networks, in order to get true what's called "broadcast domain" isolation. Anyway, I'm going to explain all about that at the end of this podcast. That's the topic. And we didn't really have much news.

Leo: Second week in a row. Those hackers are getting - they're slackers.

Steve: Has been quiet. One thing I got a kick out of, speaking of Java, is that, in a sort of a, well, we're not the first people to give up on browser plugins, the so-called Java Platform Group at Oracle formally announced the end of the browser Java plugin. And it's like, I mean, if anything, this podcast could be called "Why Has This Taken So Long?" Because for years, you know, 10 years ago, Leo, for years, starting at the beginning of the podcast, it was email viruses. It was like, "Okay, Microsoft, turn off scripting in email. Turn it off." And it just took forever. And similar, it was like, "Microsoft, turn on the XP firewall by default." Well, that took them until Service Pack 2 to get around to doing that. So this is, you know, "Oracle, kill the Java plugin." And, yes.

Leo: There's only one left now, and it's Flash. And kill that, and then we're done; right? Everything's going to be secure. Everybody's going to be happy.

Steve: Exactly. I got a kick out of this blog post because this was like, okay, so here's what they said: "By late 2015" - so, right, a couple months ago - "many browser vendors have either removed or announced timelines for the removal of standards-based plugin support, eliminating" - this is Oracle speaking - "eliminating the ability to embed Flash, Silverlight, Java, and other plugin-based technologies." So of course, not that it was always a horrible idea. It's that, well, you know, everybody else is saying they're not going to support this anymore.

So continuing, Oracle says: "With modern browser vendors working to restrict and reduce plugin support in their products, developers of applications that rely on the Java browser plugin need to consider alternative options, such as migrating from Java Applets, which rely on a browser plugin, to the plugin-free Java Web Start technology. Oracle plans to deprecate the Java browser plugin in JDK [that's the Java Development Kit] 9. This technology will be removed from the Oracle JDK and JRE [that's the Java Runtime Engine] in a future Java SE release." And then they give some links about JDK9, talking about how it's coming.

So, yes, this got picked up by the tech press saying, yes, finally, Java is being removed from the browsers. And mostly what Java is saying is, well, you know, we didn't want to give up, but the browsers are refusing to host our plugin anymore in one form or another. So we're not going to fight it, we're killing it off completely. And then so what they'll have is they'll have this Java Web Start technology I've not taken a look at yet. But it's not a plugin in the browser.

And let's just hope it doesn't have its own set of catastrophic problems. It sounds frightening. I mean, the only thing that might save it is if it requires a lot of user interaction and verification before it runs something that you obtain from just promiscuous web surfing, which is never safe when Java is your target. I mean, Java's a full-strength programming language. As you were just saying, Minecraft was written in it. You can do anything with it that you need to, which is also part of its benefit. The problem is you don't want to stick it in a web page for all the reasons we've been talking about for the last decade. Also yesterday...

Leo: Is NPAPI the plugin that they're referring to? That's the deprecated plugin from Firefox. It must be, the NPAPI.

Steve: No, NPAPI is Netscape's own browser API. So, but there's Java plugins for all browsers historically.

Okay, so yesterday Google gave us their Nexus Security Bulletin for Android. And these are fixes for February. Short version is you want to, if you have a Google Nexus device, you'd absolutely want to update because these are not really bone-chilling, but they're a concern. And it's funny, as I was looking through this and pulling this together for the show, I thought, okay, this sounds exactly like a set of security notices, like that we've been covering for years, for like a Mac or a PC. It's like, oh, yes, Android is a full-blown operating system. Even though it's hiding in a smartphone typically, it's as much of a connected OS as any of the ones we've been discussing. So it's good to see that movement in the direction of this consumer device, which is as much an OS as the desktop devices, being given the same kind of attention for security.

There were two critical vulnerabilities found and fixed by this update yesterday, found in Broadcom's WiFi driver, which is part of Google's Nexus build for Android. And the concern is that anyone who leverages this vulnerability can potentially execute code remotely, but only if they're on the same WiFi. This is why I said this wasn't as bone-chilling as some that we've seen before. So there are two remote-code execution vulnerabilities, but they involve the way the Broadcom WiFi kernel driver deals with wireless control message packets. So it's a subtle problem, only affects WiFi LAN, that is, it is not exploitable at a great distance. So it would be somebody on the same WiFi network as you.

The problem is that, if this doesn't get patched, if it can be turned into a remote code execution vulnerability, no one is saying yet that that's been done. All they're saying is that they're able to corrupt kernel memory, which typically means that externally provided data can be forced into the system. And once the bad guys figure out how to execute that externally provided data, that gives them a remote code execution opportunity. So that you want to fix.

Mediaserver, this very troubled module, which of course gave us StageFright and lots of coverage last year, is continuing to deliver. We've got two critical security vulnerabilities, additional ones, that have just been found and fixed in it. And those, unlike this Broadcom WiFi problem, are remotely exploitable. Again, this is somebody sends you something from anywhere, and that can cause problems. So web browsing, email, MMS files, just basically your device needs to process a maliciously crafted media file. Anything that some bad guy can do to get that to happen can potentially compromise your security. And then there were some moderate and - there were four high-severity and one moderate.

But anyway, as I said, as I was going through this, it's like, wow, this sounds just like a regular OS getting its security fixed. And in fact that's what it is.

Our friend Mary Jo Foley noted in her ZDNet column, and the Hacker News and Beta News and everybody picked on the fact, that Windows 10 upgrade has, as promised, moved Windows 10 upgrading from optional to recommended in the Windows Update event. So I imagine, here we are on February 2nd, the first Tuesday of the month. On the 9th, next week, which will be Patch Tuesday for Microsoft, we may find that there is now a recommended update for people who haven't yet from Microsoft.

So if you haven't yet, if you don't want Windows 10 for whatever reason, you won't want to wait more than a week before doing what you can. And we've talked about that often, the various, like the GWX Control Panel is the slam-dunk easy thing to do to block Windows 10, and there are, as we covered last week or the week before, Microsoft has now actually got an update that adds features to Windows 7 and 8.1 to allow you, using Microsoft's sanctioned approach, to prevent the Windows 10 update on a system where you don't want that to happen.

I got two tweets that I thought were interesting, I just wanted to share, on this whole ongoing and very interesting dialogue about where we stand with encryption. Matt tweeted me yesterday morning, he said, "@SGgrc You keep talking about Apple being able to do safe warrant access crypto. What about all the others that can't, but would have to?" And I thought, wow, that is a good point. It's not something that I had even thought of. And so I wanted to thank Matt for bringing that up and wanted to share it because, I mean, that's - I think it's a good point.

I have talked about Apple's billions of dollars and their budget and their clearly proven ability to create the equivalent of a very high-security safe where individual unlock keys for every single one of their phones would be kept under this hypothetical solution that I proposed as sort of a compromise, which would not be a backdoor, but would be a way of allowing Apple under court order to provide a key. And then of course last week we also talked about the idea of also requiring physical access.

But what I hadn't considered and that Matt brought up is that, well, okay, but what about all the other companies? I mean, if this was the way things worked, suddenly everybody would have big safes of users' keys. And that's clearly a deal killer. I don't trust, I mean, I barely trust Apple. And all of our experience is that, in general, companies can't keep secrets. It's incredible difficult to do. And as a result, people are having all their personal details published and passwords lost, and everyone's having to run around and change their passwords all the time. So anyway, I'm really glad that Matt brought that up because it just - it wasn't on my radar; and it's like, whoa, that's a very good point.

And this came in a DM, and I didn't have the guy's permission to share it, so I won't share his name. But he said, "The U.S. might have laws preventing unreasonable search, but a lot of countries in which Apple does business have no such protections. If you make the phone technically accessible to U.S. authorities, you make it technically accessible to every country's authorities." And that's not necessarily the case. So I wanted to rebut that a little bit and use that opportunity to clarify that this sort of compromise I've been talking about would require that the vendor of a technology like a smartphone keep a safe of individual keys.

Now, it's true, if, for example, a foreign government required those keys to be stored in their country, then that might limit the vendor's control and security management of those keys, and once again we're in trouble. So maybe there's no way to do this. I mean, I think this has been, if nothing else, a useful thought experiment because that's the only compromise I can come up with. But holes are getting punched in it that are good holes. So I wanted to share those.

We talked last week about Netstat as a useful, ubiquitously available since the beginning of UNIX, well, the beginning of the Internet on UNIX, which is where the Internet was born, command line tool. And a number of people tweeted and reminded me that our friend Mark Russinovich has TCP View. It is a Windows-only solution. So one of the reasons I wanted to talk about Netstat is it is available for the Mac and Linux machines. I mean, if you have a TCP/IP stack in your machine, along with it will be commands like ping and traceroute and netstat.

But I did want to mention TCP View because there are so many Windows systems, and it is free. It's a cute little, I think it's 285k download, nicely written like Mark's stuff, doesn't require registration. It's just an EXE that you run. The first time you run it you agree to a license, and I think he must make a mark in the registry or something because it doesn't ask you every time, but you don't have to even install it. It's just an EXE that you run, very much like mine. And it does provide a very nice dynamically updating GUI display as opposed to a command prompt. So I would say that's way more practical for most users who are using Windows. And there's likely utilities for the other OSes that are similar.

So I've been talking a little bit about sleep. And Leo, you know that I've been focusing on sleep and insomnia. And I mentioned last week that I was working on refining a solution which was going to work for me to keep me asleep all night. The particular type of insomnia I have is where I'm able to go to sleep very quickly; but, after about four hours, like around maybe 3:00 a.m. or 4:00 a.m., my eyes open.

Leo: Right.

Steve: And I'm, like, wide awake and then in the past have laid there for a couple hours, trying to go back to sleep, sometimes succeeding, sometimes not. When my friends at Starbucks see me, it's on mornings when I just gave up, and I went in and was at Starbucks when they opened. I also mentioned previously that, when I decided I was going to tackle this, back in like around early, I guess toward the middle of October, I purchased a couple so-called "sleep monitors." And these were things like one was the Jawbone UP3, and I also got the top-of-the-line Fitbit. Did some research; they looked like the best things available. And, sadly, they're very limited. They're so-called "ActiGraphs," or they use "actigraphy," which is to say they monitor how active you are.

Leo: It's a fancy word for "Did he move his arm?"

Steve: Yeah, exactly. No, and they have, you know, they can measure body temperature, heart rate, maybe respiration. You get respiration, interestingly enough, because whenever we exhale, our heart rate slows down just a little bit. And when we inhale, our heart rate increases. And so by doing beat-to-beat, inter-beat interval measuring, a device which is monitoring your heart rate can also determine your rate of respiration. So there's a lot they can get. But as I had mentioned before on this podcast, unless you are hooked up to someone's head, you just don't know what's going on inside. There is no way. And as you just mentioned, if you lay really still and aren't flopping around in bed, it thinks you're asleep. So I purchased, tried, and returned both of those.

Leo: Oh.

Steve: Because they were just junk. I would compare what they said to what I knew had happened, and there was no correlation. I mean, yes, am I disco dancing, or am I in bed? That it could determine. But it had no idea otherwise. So as you know, because I shared this with you a couple months ago...

Leo: In top secrecy.

Steve: Top secrecy because I knew the moment I mentioned this they would disappear, and I wanted my closest friends to have access to this technology, and then in successive spheres I would make it more available. There was a company founded in 2003 called Zeo. It was originally named Axon, cool name, Axon Labs, founded by four Brown University students at the very end of 2003. And their goal was to create a true, practical, real EEG sleep monitor. They ran for 10 years. And during the course of those 10 years, from 2003 to 2013, they produced three products.

The first was a bedside clock radio format where it had a display, and it was an electroluminescent display, and you put this headband on which went around your head, sitting on your forehead, and it had a little pod on it about two inches by one inch. And the headband snapped onto the pod. And on the inside were, in three in a row, three silver cloth sort of, well, they were electrodes which were able to measure your EEG, your brainwaves, from the front of your forehead. And also, due to their position, they were able to pick up what's known as "muscle artifact," which is the electrical signals caused by your eye muscles. And of course that's useful because rapid eye movement, REM sleep, is one of the acknowledged stages of sleep. So by being positioned above your eyes, on your forehead, they could actually get your brain waves and pick up your eye movement muscle artifact.

The problem was, well, okay. So that product was developed and existed. And it could dump out onto an SD card, and then you were able to use the SD card to transfer it to other things. This was sort of, what, 2003. Where was the Internet in 2003? I'm not completely clear on that.

Leo: It was on my server. No.

Steve: Was it beginning to happen? It was probably...

Leo: Oh, yeah. No, it was happening big-time. Remember we even had the bust by 2000.

Steve: Oh, that's right, okay. So...

Leo: It's funny, that was a long time ago, though, wasn't it.

Steve: It was. It's like, wow, okay.

Leo: Twelve years ago, yeah.

Steve: What did we have then? Then they did a less expensive version that had a docking station that you docked this pod in. And then, finally, their very last one, and I actually...

Leo: Steve's reaching over to get it.

Steve: Yeah, reaching, is this thing, which was called the Zeo Sleep Manager Pro. And they reduced it down to just the pod with the headband and a smartphone because later in the company life they could assume that everybody had smartphones. The original device was not dependent upon a smartphone.

Okay. So it turns out this is the real deal. And, I mean, it works unbelievably well. If you go to GRC.com/zeo, you will find a web page I just put up this morning. What happened is a surplus reseller on eBay purchased a bunch of stock of these final Zeo units - the pods, the headbands, their charging cord, and so forth - and have been selling them. The problem was they didn't have enough. I wrote to them and said, okay, I know that there are people who listen to my podcast who are going to think the idea of actually monitoring their sleep is cool. But 10 isn't going to do it. So how many do you have? And they responded that they have many hundreds, maybe a thousand.

So I've been waiting until they relisted them, which happened on Saturday. So I tweeted the news. First I let everyone in GRC's health newsgroup know so that they could get them. Then I tweeted the news on Sunday so that my Twitter followers would know. And now I'm telling everybody on the podcast, which is essentially the entire universe. So what's cool is that they are selling these brand new, never used, sealed, not in the final packaging because the packaging doesn't exist. This was just - these were purchased from, probably out of bankruptcy or after the company closed as it was liquidating its assets.

But for $40 on eBay right now, these little Zeo pods are available. You must have a Android device because Google Play store still has the Zeo software, but iOS yanked it shortly after the Zeo company died. So it did run on both smartphone platforms, but you can't get it anymore for iOS. So it requires an Android device. I bought, for this purpose, an Amazon Kindle Fire so that I had something to receive the signal. So again, GRC.com/zeo will take you to the page where I have links to the two eBay sales. One is the $40 pod, headband, and charger. The second one is the headbands only. And the headbands, Zeo said, you should replace them about every 90 days. But users with experience said that they lasted as long as a year. And I always wipe my forehead off with an alcohol wipe in order to get a better connection for my night's EEG reading.

I also, lower down on that page, have a couple links so that you can see what EKG or - I'm sorry, I keep saying EKG - what EEG the Zeo produces, which I've been recording. And I have to say I am getting very near having a practical formula. It has already worked for a couple other people who've been following along and experimenting with what I call my "alpha release" of the HSF, the Healthy Sleep Formula. So I think I'll have something to say about that before very much longer.

But I did want to let anyone who's interested in monitoring their sleep at night for what I consider a bargain of $40 - there's only, like, 90 of them there because they just - they keep not - I don't know if they're going to raise their price, or if they're going to relist them. They saw a large number of them sell, a couple hundred, after I tweeted on Sunday. I'm sure that this current auction will get wiped out quickly. But they do have more.

So if you follow the links, and it may say that it's been relisted, follow the relisting. I'll update the links as I find that they've been relisted. But I can say that I've been using it now for 10 weeks, about 70 nights, seven zero nights, and I have a complete chronology of a breakdown showing the amount of time spent in REM, in light sleep, in deep sleep, and waking up in the middle of the night and trying to go back to sleep. I also separately have a complete log of all the supplements that each night generated. And I'm beginning to see correlations now out of this mass of data that I have collected. So Zeo Sleep Manager Pro. Oh, I should mention, too, that there's now a Zeoband.com company considering launching to create replacement bands, Zeoband.com.

Leo: Oh, that's nice.

Steve: Zeoband.com. At this point they're just gathering interest. So anyone who's interested should go there. People from - shoot, can't think of the name. There's a neat hobbyist electronics company that manufactures all kinds of crazy widgets for makers. It'll come to me. And I have it on that page. They produce, or they offer for sale, silver-impregnated cloth, exactly like what the Zeo headband uses. And people have successfully repaired their headbands after the silvering of the cloth has worn off. That's what happens after some number of nights of use is that the electrodes start losing conductivity because the silver just flakes off. So SparkFun. That's the name I was trying to come up with.

Leo: Oh, neat, yeah, yeah. They're great.

Steve: SparkFun has three different links to three different grades and types of silver cloth. And if you Google "Zeo," Z-E-O, there's a huge culture of users. People have made third-party firmware. You can get other displayers and viewers of this information, all kinds of stuff. So there's, I mean, this thing was popular enough and successful enough that it created an ecosystem of its own that has continued to live on.

I should mention that the one caveat, aside from the headband's un-infinite life, is that inside is a 3.7-volt LiPo cell. And these things cannot be user opened. At some point we will all be cracking ours open in order to replace the LiPo cell. It's the same one used by the little microcopters, the little micro quadcopters. So it's a very small but widely available 3.7-volt lithium polymer cell. Those, of course, have a limited life. So, and it's not just shelf life that's limited, but it's also cycle life.

So at some point I expect that our Zeo pods will start dying. I just want to give everyone a heads-up. If I'm still into this and tracking my EEG sleep, I'm sure I will tackle cracking mine open and share the information about how to do that. But so will the ecosystem that exists. So anyway, GRC.com/zeo, Z-E-O. That'll take you to my page where I've got links to the auctions, and also links to show the graphs that I've been producing using mine.

We were talking about Syfy, and like on the question of what has happened? How was "Childhood's End" so good? Why is "The Expanse" so good? And by the way, all morning - I tweeted this yesterday - all morning Syfy has been doing a marathon of all previous episodes leading up to tonight's double episode season finale. And when I saw last week that it was the season finale, it's like, what? It's like, oh, crap, you know, now we have another thing like...

Leo: "Game of Thrones."

Steve: Exactly, like "Game of Thrones," where the seasons are short, and you have to wait a year between them, which is [crosstalk].

Leo: Well, you know, they put a lot of effort into these, obviously, yeah.

Steve: Anyway, so a number of people were happy to have that. But I wanted to answer the question, what has happened at Syfy? And there was a Wired podcast during which they recently interviewed Bill McGoldrick, who is the new head of programming at Syfy.

And Wired said: "This past week Syfy premiered 'Childhood's End,' a six-hour adaptation of Arthur C. Clarke's classic first contact novel. The show is part of an ambitious new slate of book-to-TV adaptations being overseen by Bill McGoldrick, Syfy's new head of original programming. And while Hollywood is known for misguided rewrites of sci-fi classics, McGoldrick was determined to create a faithful adaptation of Clark's novel. McGoldrick said: 'We all just wanted to honor the book and really give him [meaning Arthur C. Clarke] the recognition that he was just so prescient because all the themes and all the things he was writing about are still so valid today.'"

Then Wired writes: "For years, Syfy has tried to broaden their appeal beyond science fiction fans, populating the channel with ghost hunters, pro wrestlers, and low-budget creature features like 'Sharknado' and 'Mansquito.' And while that did pull in new viewers, it also alienated sci-fi fans." And I'm adding, and how. "McGoldrick was brought in with a clear mandate: Lure the fans back with smart, ambitious shows. Adapting classic books is part of that plan. McGoldrick said: 'We want to honor that core fan base that is passionate about the material. We're really trying to focus on that core audience. And I think the way to do that is to respect the stuff that they really liked in the first place.'" Which of course is music to my ears.

And Wired says: "One thing fans are passionate about is space opera shows like 'Farscape,' 'Firefly,' and 'Battlestar Galactica.' But in recent years, Syfy simply lacked the budget to create those kinds of shows. McGoldrick said: 'If you don't have the budget to go up into space and try to make that feel authentic, you might have to do some things that don't play to the core as much as sci-fi fans would like.'"

And then Wired finishes, saying: "But things have changed. The success of core genre shows like HBO's 'Game of Thrones' and AMC's 'The Walking Dead' have persuaded Syfy's parent company, Comcast, to invest big in the channel. That means new sci-fi shows like 'Childhood's End' and 'The Expanse' are full of gorgeous visuals and jaw-dropping special effects. McGoldrick promises that future book adaptations, which include classic works, will have a similar focus on quality. McGoldrick says: 'The wallet will open for the right show. And that's what makes it so exciting to have this job right now.'"

And I, of course, say, "Yay." And again, "The Expanse" is turning out to be 100% worth watching. And the ratings, IMDB, it's all up in the high 8s and 9s out of 10 every single episode, and it deserves it. It had a little bit of a rough start. But I think it's found its footing. And so let's hope we have many more seasons of it. And at only 10 episodes per, they've got a lot, I mean, and believe me, I read the four books. This thing can go for - it can outlive the podcast. There's even that chance.

Leo: Who wrote "The Expanse"?

Steve: Oh, shoot, I don't know. If you google it, you'll find it instantly. It's not an author that I recognized, and I've read nothing else he's done.

Leo: Yeah. All right. Interesting.

Steve: Did it come up?

Leo: I haven't googled it yet.

Steve: As far as I know, that's all he's done.

Leo: I thought the chatroom might...

Steve: Ah.

Leo: Let me google it real quickly here.

Steve: And Leo...

Leo: It is by James S. A. Corey. Oh, he is...

Steve: Yes. Oh, yeah, and he does have a lot of other books.

Leo: That's actually two people, interestingly, Daniel Abraham and Ty Franck. And they did "Leviathan Awakes," which I liked quite a bit. You recommended "Leviathan Awakes," actually.

Steve: And that's the title of the second of tonight's two episodes.

Leo: Okay. So that's one of "The Expanse" stories?

Steve: Yes.

Leo: Didn't realize that. Okay. Now I'm putting two and two together. Yeah, because I read that on your recommendation. All right.

Steve: Yeah, and it's good science fiction. I absolutely enjoyed it. And for what it's worth, if people like - if watching the - I guess we ought to find out if it's available on Audible. I didn't think to do that.

Leo: Oh, I'm sure it is.

Steve: If watching the series has whet your appetite...

Leo: That's how I listened to it, so I know it is, yeah.

Steve: Oh, okay, good. So available on Audible. Then that's what you want to do for your Audible picks because you could jump ahead. The books are always going to be better than a...

Leo: That's my problem with filmed or taped sci-fi, frankly.

Steve: Yeah.

Leo: You cannot spend enough money to do what you can do in your imagination. Ever. You just...

Steve: Well, and I would argue that the medium doesn't provide the richness of description. Actors have to act what they're thinking, where the author can give you, I mean, detail what's going on in their head.

Leo: Yeah, precisely.

Steve: Okay, now, this is more - this is not a joke. It's real. But you'll get this because we've talked about this from time to time. I received an email today, the Temperfect Mug Update #34.

Leo: Oh, lord. Yeah. Yeah.

Steve: And it was titled "2016: The Year of the Fire Monkey." And so I got a kick out of it. Then the email is trying to make me feel special by saying "For backers only." And I was thinking, you know, guys, at this point you really don't get to limit your audience or pick who gets to read this. And to give everyone a sense for this, first of all, I'll remind everyone, the Temperfect Mug was a Kickstarter project that - who even knows how long ago? It was one of the founding projects of Kickstarter.

Leo: It was a long time ago.

Steve: And with a very cool idea. The concept was that, when you initially make a pot of coffee, it's too hot to drink. And so you have to wait for it to cool down. Then it passes through a range of drinkable temperature, after which time it becomes too cold to be very interested in being drunk. So these guys said, okay, we're going to...

Leo: I like how you put that. The coffee is uninterested in being drunk at this point. I'm sorry, you may not drink me.

Steve: I'm not longer interesting to you.

Leo: No.

Steve: The idea was to not use a vacuum containment, or not only a vacuum containment, because the problem with that would be that it would successfully keep the coffee at the uncomfortably hot temperature for too long. So you first surround the coffee container with something with high thermal inertia, like rock. Or copper, a huge copper, a thick copper sleeve would be wonderful because then - but so their concept was you pour the super hot coffee into this thing. And the environment, the surrounding sleeve immediately takes up that excess heat which was making the coffee too hot to drink, bringing it immediately down to drinkable temperature. Now you've heated up this collar around the coffee. And around it is a vacuum. So the heat cannot get out, thanks to vacuum containment. And but you've taken the heat away from the fluid into the solid. And that now holds your coffee at the Temperfect temperature for hours. And wouldn't it be nice if anyone actually had one of these.

Leo: Why is this so hard? By the way, you know that the reason this came up is that there is now a cup called Ember, which is actively maintaining the temperature. You set the temperature, it's got a heater and a cooler, I guess, some sort of heat pump system, and it's actively maintaining the temperature.

Steve: Right. You can do that with a Peltier cell.

Leo: Yeah, something like that, yeah.

Steve: In order to pump the heat. Anyway, so to give everyone a sense for this, Update #34.

Leo: Oh, lord.

Steve: "January saw much progress on the Temperfect Mug project."

Leo: Oh. How exciting.

Steve: "Things are coming together."

Leo: Yes, indeed.

Steve: "A number of sourcing subprojects were conducted successfully. Loose ends were tied. And all mug parts and supplies from several factories were brought together in one place for the culmination of the Made in China phase of this project, the 'stuffing' of the shipping container arranged by our sourcing agency to bring everything to the U.S. Early in the month we wrapped up the trials and final adjustments to the lid mold, which was the last tool to be finished, and all the lids for our Kickstarter rewards got made on time. All the plastic and rubber parts are now molded, inspected, consolidated, and ready to go into the container. These are the lids, gaskets, sleeves, feet, and shutter pivots that dress the stainless steel mug bodies.

"And the bodies? Most of the parts for those were made during my last trip to the factory in December. Forty-five bodies were assembled without trouble and finished and looked good before I left. Then later in December..."

Leo: What?

Steve: "...the factory had a problem that..."

Leo: Something went wrong? No.

Steve: "...that required a large number of those assemblies to be scrapped."

Leo: Oh, crap.

Steve: "(They were not strictly following the work instructions, the problem reported in last month's update)." That would be Update #33. "In January, the factory started to have other difficulties with the assembly, it seems because a machine was maladjusted." Yeah. Yeah. Not only the machine.

Leo: Okay. This is starting to sound like a work of fiction. I'm sorry.

Steve: Oh. "After some time they found a solution and were able to rework the parts and get back on track with production."

Leo: Oh, hallelujah.

Steve: In the interest of our listeners' ears bleeding, I skipped a chunk and finished with, I just couldn't resist this one last paragraph: "We do know we won't be getting the shipment out of China before Chinese New Year."

Leo: Oh, man. Well, these things happen.

Steve: "Elvis has left the building. The workers" - they actually wrote that. "The workers are heading home to celebrate, and there's no one there to finish and pack and ship our mug parts. They'll be back February 19th." Stay tuned for Update #35.

Leo: Oh, my god. It is possible that these are well-meaning, well-intentioned people who've just had a...

Steve: Oh, Leo, I've not been sharing the updates. For six months we went through the battle of not liking the way the shutters were being bent.

Leo: They're perfectionists, these guys.

Steve: They are.

Leo: Yeah.

Steve: And I love them for that. Then there was the vacuum containment problem.

Leo: Well, you know how that is, yeah.

Steve: Not only in Zurich with the Large Hadron Collider has there been a vacuum containment problem...

Leo: Yes, no, yes, yes.

Steve: ...but with the Temperfect Mug, as well, both using similar technology. Three of the four...

Leo: There you go. It's Large Hadron Collider technology. There you go. No wonder. In fact, it's amazing that it works at all.

Steve: And you and I have - I remember you're down for one, and I am, too.

Leo: How much did we spent on that?

Steve: I think it was $189. I got the Darth Vader black one, I think, the fancy one, because I was absolutely convinced this thing was going to be the answer to my coffee prayers.

Leo: Well, it sounds like a great idea.

Steve: So, yeah, we'll see.

Leo: I'm sure it is.

Steve: Speaking of answer to someone's prayers.

Leo: Yes.

Steve: I won't drag everyone through a long testimonial. I'll just mention that Dennis Stevens publicly tweeted yesterday afternoon, and I saw this, and I thought, okay, I'll just mention this on the podcast. He said: "Steve. I use for my job" - oh, I'm sorry. He said: "@SGgrc System I use for my job, work from home, died." So that's a system he has at home, he works from home, he uses for his job. So work from home died. "Ran my copy of SpinRite overnight. No errors reported. System booted fine. #Happy." So to Dennis and everyone else who's having success with SpinRite, I thank you for your support and for letting me know and sharing your successes. And I wanted to mention that 76 Zeo pod systems have sold in the last hour.

Leo: Already? Just now?

Steve: Yes.

Leo: You're a monster. They should never have gone out of business.

Steve: Well, and see, what happened was that they were unable to compete with the ActiGraph. It's like, oh, wait, you mean it'll count my steps, and remind me that I'm not standing enough and that I need to walk more and what my heart rate is, and also monitor my sleep? Oh, well, then, what do I need this Zeo thing for?

Leo: Right, exactly.

Steve: Except of course none of them can tell you what's actually going on in your head, unless they're measuring the electrical activity in your head. So nobody who might think this will be cool will be disappointed. So, with the caveats that the headband wears out, the battery will need replacement at some point...

Leo: And it's not good for television personalities because you come into work with four dots on your forehead.

Steve: And I wanted to ask you because you mentioned the three divots.

Leo: Yeah.

Steve: But that doesn't last more than 10 minutes; does it?

Leo: Oh, yes. It lasted all morning. In fact, I went in...

Steve: Maybe you had it too tight.

Leo: Yeah, could be. I don't want to know how badly I'm sleeping. I am sleeping terribly. I know that.

Steve: Actually, Leo, I truly am about to have a solution.

Leo: I'm going to wait for the solution. I'm going to let you do the guinea pigging.

Steve: Yeah, that's what I do.

Leo: He's the guinea pig.

Steve: Yup, I'm happy.

Leo: Guinea pigging.

Steve: Happy being a guinea pig. Anyway, so this is a - we'll call this a benefit to the listeners who listen to the podcast live. They were definitely able to get the Zeo pods if they wanted them. I know, I've already had conversation with the seller this morning because they wrote to say, "What happened?" And I said, "I told you I was going to tweet about this. And now I'm going to do a podcast about it. And so you ain't seen nothing yet." So if the links don't work, if it's sold out, then I'll update the links. So be patient. And I'll have an update, a brief update, I promise I'll make it brief, next week to remind people, if there are any still left in their big box. They said they had hundreds, and they've only sold at most a couple hundred because I've been keeping an eye on it. So they probably have lots left. I hope.

Okay. So the question is, what we're going to answer, what I'm going to explain for the balance of this podcast is why no fewer than three dumb routers can create true network isolation. I'll explain why I was wrong last week, why I chose that last week, why what I have on the web page that's 10 years old, it was dated August of 2006, why I did it that way then, and why that was wrong because - okay. So, well, and exactly what the problem is. Why is it that this is the only solution?

So the goal is to create robustly, knowably isolated networks. There's no question, none, that we're going to be seeing Internet of Things security disasters. Anyone who's been listening to this podcast couldn't believe for a minute that that's not going to happen. We're seeing them weekly with mature operating systems from companies whose entire technical focus is on not having problems, instead of some Chinese light bulb manufacturer that wants to sell light bulbs, and for whom security, I mean, absolutely couldn't care less.

So the danger will be that there will be devices which phone home, or which map a port using Universal Plug & Play, or as we call it here, Universal Plug & Pray, through your router to allow some web management system to access the device. Then Shodan, the search engine for the Internet of Things, will scan the port space of the Internet and will find that light bulb where security was an afterthought. And that will mean that the instant that a problem is found, you know, it will have to buffer overruns. There will have to be exposed telnet ports, I mean, all the kinds of things like even supposedly secure routers have. You have to know that other things that are wanting to be on the Internet just for the sake of being on are going to have problems. And they will be exposed to the 'Net.

The problem then is that bad guys, the moment a vulnerability is found, can use Shodan to find the 50,000 of those which are currently plugged in in random people's homes the world over and use that to create a beachhead, to get into the device, and essentially create a connection back through the router into the user's LAN. And once there, they can do whatever they want to. They are essentially a bad guy has gotten onto a device on your network, and we know what that means. Not good. So it is imperative that, I mean, truly it is imperative that this class of what we know are going to be sketchy security, yet connected to the global network, devices not be connected to your internal LAN. So really for the first time there's a big need for network isolation.

In the fullness of time, I expect that router manufacturers will address this, that we have this notion of a guest network. Although we talked about last week how, at least in the case of one Netgear router that one of our listeners wrote in about, when the secondary network is put in isolation mode, to isolate it from the primary network, then the individual devices on that guest network cannot see each other. And it's also a huge question, what is the nature of this isolation?

And I'll explain when we get into talking about IP-to-MAC address mapping and ARP, the address resolution protocol, which is what we're about to do, why that idea worries me. It's one thing to isolate at the IP level. But it is entirely possible that these things would not be isolated at the Ethernet level. And it is completely possible to operate at the Ethernet level, not the IP level.

So the question is, how good will the isolation be in the future, when we really get it? And is it too much isolation? Because there are instances where we want the various Internet of Things devices to see each other. NEST has now a growing family of devices that are all on the same network and may need to be talking to each other. So having them physically isolated, each essentially on their own little guest spoke of the network, that may not work, either. So we want absolute bulletproof isolation which is independent of how any router functions, so that we don't - so that it's brain-dead, I mean, just brain-dead simple.

So that's why I titled the podcast "Three Dumb Routers." Because it turns out that, if you have three dumb routers, three absolutely feature-stripped, they don't have to know how to do anything except be a router, then you can achieve absolute security. The reason I have been reluctant until now to promote this is just because it seems like a lot to go through. Three routers. That's two more than one router. And one router is what most people have. And two routers gave you more security. And I'll explain why it's not enough security. So you can have one router with no security, or cross your fingers; two routers with better security; but it does take three to have absolute security.

And the reason I do like this solution is, over the course of the last 10 years, routers have dropped in cost so that you can get them for 15, 20, 25, 30 bucks. The main router doesn't even have to be WiFi. I should explain this is in a Y connection as opposed to a series connection, so that you have two primary routers, a secure router, and we'll call it an insecure router. And they both have their WAN ports connected to sort of the root or the hub router which connects to the Internet. So they're connected to its LAN ports. And you could put other stuff on that first router, but it's better not to. It's better to keep everything behind these routers.

And if you needed a third completely isolated network, then you could go to a four-router or a five-router. That is, the idea being that networks, to be isolated, have to be behind their own router. And those routers, however many you have, two or more, then need to share the primary, sort of the - we'll call it the root router.

Why? Why all of this? The problem is that it is possible for something malicious that gets on your network to either use IP protocol or one of the Ethernet protocols in order to obtain access to your network traffic, which you absolutely don't want.

So let's first discuss the scenario that I described last week, which is the reverse of what I proposed 10 years ago. Last week I said that the IoT devices should be on the inside router. So we just had two routers daisy-chained, that is, linked in series. And frankly, I'm not sure why I said that last week. I've thought, okay, what was I thinking? Because it's difficult to come up with a justification for that configuration, compared to what I had 10 years ago, which makes more sense.

The problem with last week is that, while, I mean, I know what I was thinking, and that was that the external network, the middle network, the one between the two routers would have a different network address than the one behind the secondary router. So anything malicious that was scanning its own network behind the second router, which is what I proposed last week, it would only see all of its other IoT devices.

However, a number of people pointed out that the weakness with this is that, if one of those malicious devices behind the second router did a traceroute out to the public Internet, the first hop of the traceroute returns its own gateway IP. That is, it's the IP of its router where it sends packets that are bound for the Internet. The second hop of a traceroute would be the external router's gateway IP. In other words, the second hop of a traceroute would reveal to something malicious on the innermost router the intermediate network's network numbering. And it could then scan that network. If it sent IP packets out to the Class C network, the 254 IPs in that network, the router, the gateway IP would tell it, essentially it would allow it to infer the network numbering of the exterior network. And then probes sent out to those IPs would get routed to devices on the intermediate network, which is bad. We don't want that.

So the problem is, if the secure network, I'm sorry, if the potentially malicious network is inside, on the inside router, it can - and again, we can't depend upon security by obscurity, so we have to assume that the malicious network is as smart as it needs to be, or a malicious device on the IoT network is as smart as it can be. A simple traceroute, that is to say, setting your packet with a TTL of two, a time to live of two, the first router, its own router, decrements that packet's TTL to one and forwards it to the first router. It decrements that TTL packet to zero, the TTL in the packet to zero, and sends back an ICMP message saying that this packet is expired. The ICMP message contains the IP of the interface that bounced the packet back. That will be the gateway IP that allows something malicious on the inside network to determine the exterior network's numbering, and then it can simply use standard IP probes to scan the network in the middle that I was saying was secure. And that's bad.

Okay. That's how that breaks. Now let's reverse it. Ten years ago, when I created the NAT security chaining idea, I said we're going to put the super-secure stuff on the interior router, still just two, and they're daisy-chained. Super-secure stuff on the interior router. The idea being that the router, functioning as a NAT, prevents unsolicited packets from coming in, only allowing them to go out, and so only solicited things come back in. So that's very much like the routers we all have now. We have routers on the Internet. Lord knows the public Internet is a hostile environment. So our routers are right now protecting us, our single routers are protecting us from the hostility on the Internet getting into our LANs, and we know that works.

So that was the model I had proposed 10 years go. But there's a difference between the Internet and a LAN because what this would mean would be, in this configuration, the Internet of Things devices, the potentially malicious devices would be sitting on the network in between the two routers. That always made me uncomfortable because that meant that the traffic from our secure, what we want to be our own super-secure private LAN, that traffic is passing through the environment, the network, where a potentially malicious device could reside.

Well, what could that device do? The device can scan that local network and might even find that there's - might even be able to determine that there's a router there. But it can't get in, anymore than somebody out on the Internet can get into the routers that we have right now protecting us from the Internet. But devices on a LAN have an additional power that things out on the Internet don't have. And that's that they're on the same Ethernet network.

So let's step back for a second and remember a little bit of the dynamics of how Ethernet works. We have covered this in complete detail in past podcasts. If anyone's interested, go look for Address Resolution Protocol, ARP, Address Resolution Protocol. We did a podcast that explained it once [SN-029]. But for now, I'll sort of summarize that.

The idea is that nothing about Ethernet - Ethernet was a networking technology that was developed at PARC, Palo Alto Xerox PARC, Xerox's Palo Alto Research Center, by Bob Metcalfe. And it was used to connect systems together, completely separate from the Internet. It doesn't use packet routing. It doesn't use IP addresses. There are no IP addresses on Ethernet. That was all grafted on later. Ethernet uses MAC addresses, Media Access Control, MAC addresses. And our network-savvy friends have probably seen them. They are expressed as six pairs of hex digits, typically with colons separating them, so it'll be like 00:FE:02: and so forth, six pairs of hex digits. That represents, since each pair of hex is eight bits, that means that three of those is 24 bits, and six of those is 48 bits. The MAC address is a 48-bit ID. And by convention, the first three are a manufacturer ID, and the second three pair is a serial number.

So the idea was, in the original concept for Ethernet, every single Ethernet NIC, as they were called, Network Interface Card, it would have a globally unique address. That meant nowhere would two different NICs, Ethernet adapters, have the same address. And this was very clever, especially back then. This was a long time ago. This was in the '80s. The idea being that, if every single thing that could ever be hooked up had a unique address, then you didn't have to worry about them having the same address. You didn't have to have jumpers to, like, set the address. Everyone remembers, you know, IRQ disasters and COM port jumpers and all of the mess that happened when you didn't have a guarantee of uniqueness.

Well, Ethernet solved that from the beginning. The idea was, if every manufacturer had their own high 24-bit number, and that manufacturer never made two adapters with the same lower 24-bit number, you can by definition never have two that had the same 48-bit number. And that's the way Ethernet is. But none of that had anything to do with the Internet Protocol, which came afterwards.

So now we have a problem. We've got Ethernet, and we love the way that works. Things are wired together. Ethernet works. Who was the company, Leo, the early networking company? With an "N."

Leo: 3Com? With a "N." Oh, Novell.

Steve: Novell.

Leo: Novell, yeah.

Steve: Yeah, 3Com and Novell. Novell was really big in the early days. And remember, these Network Interface Cards used to be full-size. They cost a couple of thousand dollars, a thousand dollars each, just for one of these things. And I remember at GRC back in mid-1980, we had a big - the office was sort of two suites put side by side. And our Ethernet backbone was a single "U" that - it was a coax cable that was terminated at each end.

Leo: It was serial. So if anybody unplugged their computer...

Steve: Right.

Leo: ...the whole network went down.

Steve: Right.

Leo: Like Christmas lights.

Steve: That was one way to do it. The other was you had taps.

Leo: Right.

Steve: So you had this...

Leo: You'd have a ring, yeah.

Steve: Exactly. It was called 10Base2. Or, wait, no, 10Base...

Leo: 10BaseT.

Steve: "T" was twisted pair.

Leo: Twisted pair.

Steve: 10Base2. Oh, yeah, I think it was 10Base2, and then 100Base2 was when we went - so it used to even be 10Mb, which back then was blazing speed.

Leo: Mm-hmm.

Steve: So anyway, the point was that it worked. Now along comes IP. And we want to give machines, computers and routers, we want to electrically connect them with the Ethernet, which we already have, and it works. But we need to give them, not these wacky MAC addresses, but IPs. Because the other thing about the MAC address, it is not routable. The whole Ethernet technology is a nonroutable technology. It is, by definition, point to point. Everything is on the same big network.

And because everything has a unique IP, you just say, hey, you just send something to that - I mean, sorry, not a unique IP, a unique MAC address. You just send something to that MAC address, and it gets it. I mean, because - oh, and what happens is everything is listening all the time. All Ethernet devices listen to everything going on. So they're sucking in all of the traffic because they're all connected to the same big single backbone, tapped off of it. So they're listening to all the chatter going on. And as every packet is received, they check to see whether the destination for the packet is their MAC address. And, if so, it's for them. And so then they process the packet. If not, they discard it. But, and this is key, they do receive it.

Okay. So we want to add a routable protocol. That was the brilliance of the IP protocol, was this notion that we would take 32-bit addressing, and we would partition at an arbitrary place. We would cut that 32 bits so that the uppermost chunk of bits was a network number, and the balance of the 32 was a device on that network. And so what makes this routable is that this forms a hierarchy of networks that allows routers all over the world to look at the destination IP and figure out where to send it to get it closer to its destination using routing tables, which are able to be efficient due to the same hierarchy.

Okay. So we need to add a routable numbering system to a network Ethernet that doesn't have any. It's just MAC addresses, point to point, no, there was never a concept of routing back then. So what was created was an additional protocol called ARP, Address Resolution Protocol. Like its name sounds, it is used to resolve addresses. And the way it works is as follows: Every device that is on the Ethernet, and there's like one Ethernet, gets an IP address. And that's manually assigned. That's why, for example, DHCP assigns an IP address to our computer when we ask the router for one. We just say, yeah, I don't care what my IP is, just give me one. And so the router hands it out. Or we manually configure an IP address. Typically, you don't manually configure a MAC address. You can, more recently, override the default MAC address that NICs have. But typically you don't need to because they're still going to be globally unique.

So the individual endpoints are assigned kind of arbitrary IPs. They have to be on the same routable network so that they can talk to each other, and so that things outside that network know how to get in, and so that those devices know to send things to the gateway IP if they're bound for outside the network. So you still need to obey IP routing rules. But the particular IP any given device with an Ethernet adapter receives doesn't matter.

So one of the things that is configured for any IP device is the IP address of its gateway. By definition, the gateway is the IP to which will be sent any packets bound outside the network. And so I'm sure anyone who's looked at their IP configuration has seen, oh, look, there's my gateway IP. And it's, on 192.168 networks, it's typically 192.168.0., sometimes 0.1, more commonly .0.255 or 254. It's up at the end of the block of IPs. Or sometimes .100. It doesn't really matter. Again, as I said, IP numbering is arbitrary, as long as it is consistent over the use of that device on the network.

So when a computer turns on and comes up on an Ethernet network, in the background, part of the booting up process, is the kernel driver knows that it's been assigned a certain IP, and that it's been told the gateway has a certain IP. So it needs to know the MAC address of the gateway. That's the key. Remember that it's hooked to Ethernet. And Ethernet uses MAC addressing to send packets around, not IP addressing. IP is sort of - it's an associated layer on top. But the actual packets are sent to MAC addresses, not to IP addresses.

So in order for this computer to connect to the outside world, it has to know the MAC address of the gateway. So what it does is it sends out what's known as an ARP broadcast. The Ethernet protocol, just like the IP protocol, has a concept of a broadcast address. It's an address to which something can be sent that everybody will receive. And the ARP broadcast uses the ARP protocol. In the Ethernet packet header is some bits for protocol number. And so, very much like with IP, where we have the TCP protocol, UDP, ICMP and so on, Ethernet had bits reserved for different protocols, and ARP is one of them.

So that's what makes a packet an ARP packet is it's got the protocol bits set to ARP. And this particular one is a broadcast that says to everybody, who has this IP? And so it uses the address resolution protocol specifically created for the IP protocol to say, hey, listen, everybody, do you have this IP? And everybody receives it. And if, for example, you're asking for the IP of the gateway, you're asking for 192.168.0.255, for example. And so you say, "Who has that IP?"

Well, the router receives that ARP broadcast, which is an Ethernet broadcast, and looks and says, oh, that's me. And so it responds to that device, saying I am the MAC address that is responsible for this IP. Now the computer that we've just turned on knows how to send its IP packets over the Ethernet network. It knows, if it wants to send it to the gateway, that they need to be addressed to that MAC address. So ARP creates this mapping.

The problem is that there's no other security protocol in place. That is, this assumes everybody on the network can be trusted. And in our deliberately adversarial, malicious, IoT device model, that's absolutely what we don't have. We need to tolerate a malicious device on whatever network it's on. And where we left off with this was good stuff goes on the inside, IoT stuff goes in the middle. And after we hear from Leo, I'm going to explain why that breaks.

Leo: We got caught in an ARP hole. But we're back. And now we get to the valves. That's next. You know, actually, "valve" is a good description. Isn't that what they used to call, what the Brits used to call radio tubes were "valves."

Steve: Yes, because they are.

Leo: And transistors are valves; right?

Steve: They're electron valves.

Leo: Yeah, one-way tickets. All right, Steve. Three Dumb Routers, Part 2.

Steve: Okay. So we have the address resolution protocol, which is used to allow Ethernet devices, which are on the same Ethernet network, because remember, Ethernet is nonroutable, to allow them to navigate and negotiate routable IP addresses. Oh, and by the way, I was going to mention, you can now understand how it is that you can have multiple IPs on the same device. It's often called multi-homed, the idea being that a single Ethernet adapter could be assigned multiple IPs.

At first people think, wait a minute, how is that possible? Well, now we know. Remember that the ARP broadcast says, "Who has this IP?" And so if the OS behind a single MAC adapter, you know, NIC, if it has been configured with three, for example, Internet protocol addresses, IP addresses, it says, "I do. I'm the one who gets packets that you want to send to that IP." So that's how you can have a many-to-one mapping of IP addresses to Ethernet adapters.

The problem is there is no security in this system. And this was all designed back in an era when we assumed everybody was going to be a good actor on the Ethernet. So there are terms like "ARP spoofing" that have been used, and people may have heard about, if they don't know the details. Well, here's the details. We have to assume that a malicious device is sitting, like a light bulb, is always on, listening to the network, knowing what's going on. The mapping, this ARP technology, has a timeout. And there's additional complications I won't get into.

But it means that, from time to time, devices will be needing to refresh their mapping. That is one of the ways that allows devices to change their IP address and eventually be recognized by equipment at a new location within the same network. So this is not something you only do once. The reason I mention this is that, if you screw in a malicious light bulb at some point in the future, so that it's late to the conversation, it's still going to be able to monitor all of the Internet traffic. And if it sees, or I should say when it sees, a broadcast asking who has the gateway IP, and of course it knows the gateway IP...

Leo: I do, I do.

Steve: Sorry?

Leo: I do, I do, I do.

Steve: Yeah. It can respond with its own MAC address before the router does. And so...

Leo: Oh, clever.

Steve: Yes.

Leo: So it can infect the whole network.

Steve: Exactly. A malicious device, sitting on an Ethernet, will hear the "who has, who has, who has" from every device on the network. And if it's been designed to be quick, or just is quicker, it's able to get its response back to the querying device first. And that device will think it's the gateway. Which means that all of the traffic goes to it, rather than to the gateway. But even without that, in a shared environment, if, for example, some of these devices are wired, if they're not behind a switch, then it's just like the old days where everything is on the same physical wire, and all the packets are on the wire.

So there are active and passive ways that something malicious on an Ethernet network can, in the worst case, monitor what's going on, but very likely create its own man-in-the-middle position. If it were to respond to the interior router before the exterior router can, the interior router will send everything bound for the Internet to the light bulb's MAC address. And that then becomes a man in the middle, able to filter all of the traffic. And it would then forward it to what it knows is the real MAC address, so that it would also remain invisible. All the traffic would be going through it. Nothing on the interior network, which believes it is super secure, would know.

So that's why super-secure inside, with malicious network able to see and receive the traffic in the middle, doesn't work. And of course we first covered why the reverse doesn't work because something malicious on the inside router can figure out the exterior network. Now, routers block Ethernet. Routers are IP routers. They are not Ethernet routers. So, for example, ARP doesn't, it doesn't make any sense for ARP packets, the address resolution protocol, to go through a router. They don't. They are always constrained to a single LAN. So what that makes is that these dumb routers are just smart enough to route IP, but not to route Ethernet because Ethernet is by default a nonroutable protocol. So they block, they all, absolutely all, because it makes no sense, block the address resolution protocol.

So now what we have is we want security. And as I said at the beginning of this, and I have now demonstrated, the problems with any two-router configuration is that either the router, the malicious router on the inside could gain IP access to the IP network on the outside, that is, in between the two routers, and that's not good. It can't get Ethernet access, and it can't use ARP, but it doesn't need to. It can use IP packets and use the routers to route the data into the network that you wish was more secure. If we switch things around, now the malicious device can use the Ethernet because it's on the same Ethernet as the link between the routers, and it can use Ethernet protocols, like ARP, in order to get up to mischief. So neither of those work.

The only secure solution, and the beauty of this is it is utterly bulletproof, is we put each of the networks behind their own router. That constrains them to their own LAN, and the routers block Ethernet and block ARP in any way for them to get out. So each of those networks behind their own router is able to send data out. Now, in the worst case, let's say now that one of them is malicious. It could do what I first proposed, is it could do a traceroute. If it did a traceroute, it would see its own router. Then it would see the shared router, which is the one, the Y'ing router, where these separate routers' WANs are plugged into the LAN connections of this root router. It could determine that router's gateway IP. That could tell it that intermediate network's IP numbering, but nothing is on that intermediate network, this "Y" network, except other routers' WAN connections, and we know those are safe because those are what right now protect us in a single-router configuration from the Internet. Of course, assuming that those routers don't have security problems or other problems.

But this gives us what we want. It means that essentially, by having a single root router with then spokes off to additional routers, each of those networks, and it doesn't matter now which one is secure and which one is not. You could have three. You could have four. You could have as many as you want, all concentrating down to a single shared root router that goes out to the Internet. And each of the interior routers completely isolates its network from all the others. That's secure.

I've been shying away from going this far because three routers is a bigger pain than two routers, which is a bigger pain than one router. But the beauty is they're cheap now. Probably you've got some in the closet, so it may not be a problem. That root router could be a non-WiFi router. In fact, it probably does not want to be WiFi, unless maybe you used it for a guest network. But it's really better if nothing, if it has no devices on it at all except other routers. Those are where you would want to have your WiFi for your truly secure network and your probably not-so-secure Internet of Things network. Many of them are going to be hooked up with WiFi, like the various NEST devices and so forth, and the infamous light bulb that is trying to break into our network. That's the only way I can see it being secure.

The beauty is this doesn't require any configuration, anything fancy, any settings of any kind. It's just it uses the basic NAT concept that all routers share. It blocks Ethernet, it routes IP, and it fully isolates the networks that are behind each of these routers so that they cannot see each other. And I don't think there is any - I don't think there's a simpler way to do this. Yes, if you have a fancy router, if you've got a firewall, if you're using pfSense, if you really know what you're doing, you could pull this off with a single device. But for I think 99.9% of us, if we're going to start playing with Internet of Things devices, give that its own router, take your main router, and have them both feed into a router that goes out to the Internet. Then you've got isolation, and you don't have to worry about any of their settings.

Leo: Are you going to update your page to reflect this new setup? Because I think I need a map.

Steve: Yeah, just I don't want to...

Leo: No. I guess not.

Steve: I've got to work on SQRL. So I'm...

Leo: Let's see. How do we, I mean, I have the show notes. I guess that's the best way to do it, right, is to have people...

Steve: The show notes don't really show anything.

Leo: They don't have a diagram.

Steve: Actually, the current page, there is a link on that page, the router configuration, where I go into much more detail. And so there's two pages there. There's the main page linked off of the GRC website menu. And several times I refer to another page. If you read those, you will not have any problem setting up a "Y" configuration.

Leo: Oh, okay. All right.

Steve: You'll have everything you need.

Leo: Yeah. Because I think I need to do that. And you're right, I have routers all over the place.

Steve: Yeah. We have all got - we have all of the old ones that we're no longer using. This is the way to do it. Simple. At the expense of a few more routers, it creates absolute network isolation.

Leo: Yeah, yeah. Somebody in the chatroom said, "Leo, here's your map. The letter 'Y'." Yeah, that's true.

Steve: And I think you'll find it just works. You just plug them together, it's like, oh, it works. And that's what people are going to experience. It just works. And the beauty is it is absolutely secure.

Leo: So none of them in bridging mode, each of them doing their own NAT.

Steve: Right.

Leo: That's the key. And the bottom of the Y is coming to public, is out to the outside world.

Steve: Exactly.

Leo: The center part is of course your main...

Steve: Your root router.

Leo: You have a root router. And then you have your two subsidiary routers, one of which is an IoT router.

Steve: Yup. And doesn't matter which is which. And you could have three, if you wanted to have three isolated networks. What this does is each router...

Leo: Can have a menorah, if you want.

Steve: Menorah, yes.

Leo: Have a menorah. The whole idea is that there's that root router, and then they're all subsidiary off of that, and none of them talk to each other. Each of them, it's basically a handmade VLAN, is what it is. If your root router could do VLANs, that's exactly what it would look like.

Steve: Yes. Yeah.

Leo: Yeah, yeah. But segmenting is not enough. It really - it needs to be a VLAN.

Steve: Yeah. And a VLAN you need to be careful with because there you're using VLAN tags to differentiate networks.

Leo: Right.

Steve: But unless you have a switch, which is also isolating networks based on VLAN tags, then you still have a situation where something on the Ethernet can see all the packets. You just don't want...

Leo: Where this falls down, of course, is if you're on one leg of the Y, and you somehow need to talk to the IoT device.

Steve: Correct.

Leo: Because it's isolated. You can't.

Steve: Correct.

Leo: And that's the whole point. It can't talk to you, and it can't hijack you.

Steve: Correct.

Leo: And some IoT devices, of course - but you could just join that network.

Steve: Yes.

Leo: It looks like another network.

Steve: Exactly. The idea would be your iPhone, for example, has that WiFi network in its list. And so if you need to talk to that device, you just switch over to that wireless network, talk to it...

Leo: You name that WAN "Caution: Insecure IoT Network."

Steve: "Don't stay here."

Leo: "Don't live in this."

Steve: "Visit briefly."

Leo: And you want to make sure, and unfortunately way too many devices do this, that you don't have devices that will automatically hunt between different access points. Because you don't want it to actually get in there. I guess you'd have to forget it manually each time.

Steve: Well, as long as it did not know your...

Leo: Right.

Steve: As long as it didn't know your WiFi password...

Leo: You have to forget it. But it does, the minute you use it, it does. Now you have to forget it after using it. So "Caution: Insecure IoT network, do not use, and make sure you forget it after you use it, otherwise it will automatically log into it next time."

Steve: I see. Right, right, right, yes, yes.

Leo: It's a long SSID. I don't know what the limit is. Good stuff. Again, it's GRC.com slash...

Steve: I don't have an IP for that.

Leo: Okay. A subdivision or whatever, yeah.

Steve: I mean, I don't have a simple URL.

Leo: Okay.

Steve: It's Research. Then the first...

Leo: Under the Research menu.

Steve: The first item under Research > General > NAT Router Security.

Leo: Okay, that's easy to find.

Steve: That's easy to find.

Leo: The thing that's so great about GRC is it's his, you know what, it's Steve's Minecraft server. He's been digging holes there, building edifices there for years. And there's so much stuff there, and it's really fun just to browse around. Now, of course I've got to remind you that the main thing, the thing that pays the bills for everything else, is the great SpinRite, the world's best hard drive maintenance and recovery utility. That's it. That's the only thing you have to pay for there. Everything else is free, including Security Now!. He has audio versions and transcripts, written transcripts of each and every show there, plus the show notes, and you're going to want to read these show notes, I think. That's at GRC.com.

Now, we also have copies of the show, audio and video, for no apparent reason, at our site, TWiT.tv slash - well, people like to look at you - at TWiT.tv/sn. That's the URL for that. But it's also, you know what, you can subscribe to this show, one of the oldest podcasts in the world. Ten years we've been doing this. It's on every podcatcher there is. And if it's not, shame on them. I mean, and there's five Apple TV apps, and it's on every one of those. And there's a Roku app, and there's like an infinite number of apps on iOS and Android and Windows Phone. Get a TWiT app, or get a podcast app if you don't want to get a TWiT app, and subscribe because you do not want to miss this show. In fact, what you want to do is burn DVDs of the show and put it on the shelf because it's an education.

Steve: Hey, don't we wish we had tapes of all of the old Call for Help and Screen Savers episodes. And so, you know...

Leo: I do. I mean, somebody does, I guess. But that would be great. You know, they found Super Bowl I.

Steve: So someday this will be the same way. I don't know if you and I will still be around, but still...

Leo: I've got the archive. They found Super Bowl I. And the NFL doesn't want it, but they won't let the guy sell it. It's like, come on.

Steve: Wow.

Leo: Nah. Anyway, thank you, Steve. We will see you all next time.

Steve: My friend, a pleasure. We'll do a Q&A next week. And we're down to only four Zeo pods left in the auction, down from a hundred.

Leo: Wow.

Steve: So our live listeners got the benefit.

Leo: Wow.

Steve: And I'm sure that they will restock, and there should be more available for those who are picking up the podcast on their...

Leo: I'm glad I got mine.

Steve: On their devices.

Leo: Wow. I bought, like, four of the refills, so I figured that'd get me through a year, anyway.

Steve: Good. And I'll have a healthy sleep formula soon.

Leo: I need the healthy, healthy sleep formula.

Steve: Many people do.

Leo: Thank you, Steve. We'll see you next Tuesday, 1:30 Pacific, 4:30 Eastern, 21:30 UTC for Security Now!.

Steve: Bye.