Listener Feedback #240

Description: Father Robert and I discuss an "update" on Microsoft's GWX remover; an encouraging direction for the Windows 10 Edge browser; HP in the doghouse; "Oh, yeah, that's what I meant to say about how to upgrade a site's password hashing"; a really terrific Dynamic DNS hack; another update on Windows Update; a distressing heads-up about how some unseen behavior of our web browsers is fatiguing our SSDs; a bit of errata and miscellany; and then a discussion of feedback from our terrific listeners.

High quality (64 kbps) mp3 audio file URL: http://media.GRC.com/sn/SN-580.mp3
Quarter size (16 kbps) mp3 audio file URL: http://media.GRC.com/sn/sn-580-lq.mp3

SHOW TEASE: It's time for Security Now!. Steve Gibson is here to give you the skinny on the Windows nag remover; HP is in the doghouse; browsing the web might kill your SSD; and your questions, Steve's answers. Security Now! is next.

FR. ROBERT BALLECER: This is Security Now! with Steve Gibson, Episode 580, recorded September 28th for October 4th, 2016: Question & Answer #240.

It's time for Security Now!. It's a safe harbor in the increasingly turbulent sea that is security in the age of the Internet. We're here for a two-hour tour with our skipper, Explainer in Chief Mr. Steve Gibson - of course, Steve Gibson of GRC.com, creator of ShieldsUP!, SpinRite, and SQRL, the security tools that we love, that we know. Steve, my friend, one last time before Leo comes back I get the honor of working second chair with you.

Steve Gibson: Well, and I love it, Padre, because you never reuse your analogies for what this - now we have the safe harbor in the stormy seas. So yes, indeed.

FR. ROBERT: Well, if you're Skipper, I get to be Gilligan.

Steve: Yeah, okay. Oh, no, you could be - wasn't there some scientist?

FR. ROBERT: Oh, the Professor.

Steve: The Professor, of course.

FR. ROBERT: I didn't want to [crosstalk] Professor. I might be Thurston Howell III.

Steve: Thurston Howell III, yes.

FR. ROBERT: Yeah, my kind of character. But, sir, we've got a packed episode because of course we've got the Q&A which we were going to get to last episode, but we didn't. But it was a fun episode. It was full.

Steve: It was, you know, two hours of quality content is actually our goal. So we have more things to talk about, even though it was only yesterday because we're recording this Wednesday night, the day after, because you're going to be a traveling man for the next couple weeks. And Leo has been gone, of course. So we're doing this now. So if our listeners are confused, if anything like major security catastrophe happens, like in what they're seeing as "last week," and wonder why I'm not talking about it, it's because I didn't know about it in the future because, as I mentioned yesterday, our time machine is broken. Actually, there's a part that I need, the [wegelstegen], which is only available downstream, down the time stream. And so - since when you break in the past you're kind of stuck there.

FR. ROBERT: It's a time Catch-22. You need a time machine to get the part for your time machine.

Steve: Precisely that.

FR. ROBERT: That's always a problem. Yeah, it's right next to the flux capacitor and the heart of the Tardis. So there we go. And actually I just wanted to say I did check out the time travel show that you mentioned.

Steve: "Timeless," yeah.

FR. ROBERT: "Timeless." And it does look actually pretty good.

Steve: It looks, the special effects - I'm kind of a special effects junkie. And I should mention that last night, after we talked about the original Trek series, I thought, you know, I think Amazon has all of those. And so I watched one. Not the "Gate to Tomorrow" or whatever that one was.

FR. ROBERT: Yeah, "City on the Edge of Forever," yeah.

Steve: But I watched the one where they mistakenly jump back in time, and a pilot who's flying a jet sees them and attempts to intercept. They beam him out. The tractor beam shakes the plane apart. Now they've got a problem because now he's got information about the future, blah blah blah.

FR. ROBERT: Right, right.

Steve: Anyway, so I just thought, okay, I'm just going to watch this. And I get why now it's, I mean, it's 50 years ago; right? Because we're in the 50th Anniversary of Star Trek. And even 50 years later, while, yes, the sets are made of cardboard, and it's a little hokey, there's a charm to it that explains why it worked back then. So that, I mean, even now, just there's like - it's so full of good writing and little touches that I just thought, okay, this, you know, it's charming. It's not edge-of-the-seat gripping. But of course, because we've all memorized all those plots by heart. But it was fun. So we've got a bunch of other stuff to talk about.

FR. ROBERT: Yes, we do. Yes, we do.

Steve: We've got - I mentioned - I keep wanting to say, okay, should I say "yesterday" or "last week"? I'm not sure. Oh, well, last podcast.

FR. ROBERT: Last episode, there we go, yes.

Steve: On the last episode...

FR. ROBERT: Last time on Security Now!.

Steve: Last time I talked about this new update which I would provide links for on the GRC Link Farm page to Microsoft's official GWX remover. I've got news about that. After all, it's been a whole episode, and I've got news. An encouraging direction, but a little bit of misdirection also, for the development of Windows 10 Edge browser. HP has landed themselves in the doghouse, and they're trying to behave now. And then I titled this one, "Oh, yeah, that's what I meant to say about how to upgrade a site's password hashing." That's actually a topic.

A really terrific dynamic DNS hack that a listener suggested that I just love. Another update on Windows Update. A distressing heads-up about some unseen behavior of our web browsers which has the consequence of dramatically accelerating SSD fatigue. A little bit of errata. Some miscellany. And, given that we don't completely occupy two hours with that stuff, we'll take as many questions and comments and thoughts and discuss those things from our listeners as we have time for.

FR. ROBERT: Who knows? It might actually be when Leo gets back that we finally get to the Q&A. Hey, look, if you've got good topics, you have to talk about those topics. That's just how it works.

Steve: Of course. And really that's what the show's about anyway. It's called Security Now!, and I'm happy to have feedback from our listeners. It makes great material when we need some.

FR. ROBERT: I would like the record to show that all the times that I've been able to sub for Leo and talk to you on Security Now!, I think this might be the first time where your rundown at the beginning of the show actually contains a few stories that sound positive about security. I'm blown away. I mean, normally it just scares my pants off. So this is fantastic. This is not just my last episode with you in this particular run, but it's a momentous occasion where Steve is actually going to give us some good news about security.

Steve: It's nice to know that, after all, even the word "security" is meant to imply safety and what it says. So every so often it actually does.

FR. ROBERT: Indeed. Okay, Steve. So last time, and by "last time" I mean last episode, which we really mean yesterday, we talked about how Microsoft was finally pushing out an update that would kill the much hated Get Windows 10 nag screen.

Steve: Yes. And it wasn't clear, well, actually, not only that, but there's like eight or nine different widgets that they put onto people's machines at one time or another. Remember that it was very controversial, at one point they began, they, like, changed IE so their banner was coming up in your browser saying, oh, you know, you should upgrade to Windows 10. It's like, where did this come from? It turns out it was on your own system. It wasn't from a website you were visiting. So they famously pushed this thing like crazy.

So last time we talked about KB3184143. That's the magic number for getting rid of this, KB3184143. I did not know then whether it would be necessary for people to go to that Knowledge Base article, which I have a link to in GRC's Link Farm, or whether Microsoft was going to proactively push this out in Windows Update and sort of voluntarily do the right thing and remove all of that debris.

Well, kind of yes and no. I mentioned, I think last episode, that I had seen that Windows Update had some new goodies for me. I had not at that time looked at them. I looked at the Optional updates, and there it was, unchecked but available. So it is not necessary for people to go find the link, but it will be necessary in this instance to say, yes, I do want 3184143. So you'll find that probably henceforth sitting there in the optional category. And if you install it, it'll pull all the GWX stuff out. So it's better than it not being pushed at all. Maybe they'll promote it to Recommended at some point. Who knows? But it's not now. And it's not important, either.

FR. ROBERT: At some point I'm sure someone's going to do a breakdown. But I'd like to see if this actually really does pull out all of the cruft that has been added to Windows 7 and 8 machines over the last year and a half because...

Steve: Well, remember, though, there is also new telemetry that they've been back porting. And I don't think this pulls that out. That's all part of their long-term strategy. I think this is just GWX that it removes.

FR. ROBERT: That's not - I'm not real happy about that.

Steve: I agree. Okay. So, many people tweeted this little news item. And I don't know whether they read the entire, like all the fine print, because it sounds very encouraging until you get down to the last bits. And it's like, oops. So the good news is, and the headline that grabbed people, was that Microsoft was moving Windows 10's Edge browser into its own VM. This was picked up by so many of our listeners. And you and I talked about it. I remember you were talking about your solution for basically somehow creating browser isolation.

FR. ROBERT: Containers.

Steve: Yes. And I've talked about thinking that the only way to do this safely is, one way or another, and there are of course many ways to skin that cat, but not to have the browser running in the machine where you also have all your other valuables, but somehow isolate it. Well, this news is, as the headline says, that Microsoft has announced that the next major update to Windows 10 will run its Edge browser in a lightweight virtual machine to make exploiting the browser to attack the underlying operating system or compromising the user's data significantly more challenging. They've got a name for it. They call this Windows Defender Application Guard for Microsoft Edge. Doesn't exactly run off the tongue, but clearly explains what it's about.

Now, Edge is a great browser. We've talked about the fact that Microsoft bit the bullet and truly did a rewrite from scratch, taking everything they learned, and we hope they learned a lot, from Internet Explorer because it was getting a lot of arrows in its back for years, and started again. The architecture, the design of Edge is state of the art. And I'm impressed with it. I would like to be able to run it on earlier operating systems. But no, that's just another reason to drag you up to Windows 10. It is already constrained in a sandbox where they're deliberately attempting to isolate the browser, recognizing, as we all do, that it's just too difficult to guarantee containment of something as complex as a contemporary browser.

So it needs its own firewall, essentially. So putting it into a VM leverages some technology that we got earlier this year, basically Hyper-V being in there underneath things. And it does sound like they're doing the right thing. The way they're implementing this, they're not putting an entire second copy of the OS into a VM, but they're using virtual machine technology to very tightly constrain what the browser can do. All of that is good news. All of that got everybody excited. Only one problem. It's only available on the Enterprise edition of Windows 10.

FR. ROBERT: Oh, naturally. I mean, who else would want that? That's a completely useless feature to the rest of us.

Steve: It's like, "Oh, Microsoft." So, okay, now, that's today. Maybe they're doing it - and it's controllable using the whole Windows Group Policy system. There is this notion of untrusted and trusted sites. And it's not clear to me that this is such a great idea, why there needs to be a delineation between trusted and untrusted. Part of the problem, though, apparently, is that the VM does what it should do and, for example, doesn't allow permanent alterations to the system. So apparently you want to run as many things as possible in the trusted site outside of this VM and as few things as necessary, but the dangerous ones, in the internal one.

So the more I looked at it, the less wonderful it seemed, not only because most of us won't be running the Enterprise edition of Windows 10. But this could also be sort of some incrementalism, where they're going to get it out there, develop some experience with it, let corporate IT manage their networks. Maybe it's a little bit of a tease for getting corporate America or corporate global to consider Windows 10 because that's kind of a hard sell, even now. But anyway, for what it's worth, in the future, I mean, I like the direction they're going in. And so with any luck in the future they may be bringing this out for everybody. But at this point, Enterprise only.

FR. ROBERT: I did an interview with a company at the Intel Developer Forum that had a version of this. They actually used Intel Skylake or better processor. So they used that plus the TPM that was built into a lot of Enterprise machines. And they're calling it micro VM. It's essentially a container, but it acts like a VM. Like you said, it's a VM that's been stripped down to the core, actually below the core. It's a VM that really won't work unless it's in a full operating system.

Steve: Right, it's more sort of like an API VM.

FR. ROBERT: Exactly, exactly. But the important thing was that it could launch in 15 milliseconds. And that's always been the issue with running VMs in Enterprise. We understood it's better to do that. But did you really want to stop and wait another 30 seconds for a VM to spin up? In this instance you don't have to. I mean, it's essentially the same thing as if you just clicked the icon for the browser itself. But Steve, don't you feel as if we're heading in this direction? I mean, yes, it might happen with Enterprise customers first. But eventually we should get to the point where all native Microsoft apps will run in some sort of micro VM. They should not touch the actual operating system at all.

Steve: Yes. And the coverage for this did talk about, first of all, that one of the reasons the deployment was somewhat constrained is that there were some hardware requirements. You mentioned Skylake processors. There are some hardware requirements that all consumer processors may not meet. So it won't run on some hardware platforms. And the coverage also did talk about other Microsoft applications which could similarly benefit from this kind of containment. And of course one of the

other things that many of our listeners have aimed me at is the Qubes OS. And I've talked about it, I've responded to people saying, hey, why not that? And I haven't run it myself, but I've watched it being run. And to me, that's a heavyweight VM. I mean, there, from my watching it in use, it seemed like the notion of compartmentalization dominated the user's experience.

And from my standpoint, I don't want to think about compartmentalization. I just want it to work. I want it to be not seen and not heard, just there. Whereas the whole point with Qubes is, I mean, to me it felt like something an NSA workstation would use, where you really are willing to focus on what goes in which VM and who has to talk to each other and which things need isolation and all that. It's like, no. No, I just want to run my apps and have them unable to damage my system. So it needs to be, for me, way more transparent than that.

And our listeners know that I built a machine with a Haswell chipset, back when there was the threat that Microsoft was going to stop supporting Windows 7 and only support the later OSes on the newer hardware. And I said, whoa, whoa, whoa, whoa, whoa. I built one with either 64GB or 128GB, the reason being I just want to have lots of VMS. And they'll be spun up, and the browser will run in one, and other things will run somewhere else. And to me, for me, if I'm going to make the move to a 64-bit OS, I might as well take advantage of the fact that memory is comparatively inexpensive now.

FR. ROBERT: Right, right. And it's interesting, this is also a strategic move for Microsoft because they are heavily invested on the VM side. They've got their own hypervisor. Their Enterprise customers are really pushed to VM. And I think they're not too happy with the incredible, incredibly explosive popularity of containers. Containers don't get them the same profit margin that they would off of VMs. So they understood they needed to create something that was lighter weight. They needed to...

Steve: Oh, you mean like from a licensing standpoint?

FR. ROBERT: From a licensing standpoint.

Steve: Oh.

FR. ROBERT: Absolutely. And we've seen this both from Microsoft and from VMware, which are two companies incredibly invested in this. VMware has their own version of this, not really available. The one they have right now is not great, but they've got one coming up that is also incredibly lightweight. It's essentially a container, but it runs inside of VM management. And their selling point, and this is the same thing for Microsoft, their big sell is it's like a container, it works like a container, you set it up like a container, but it doesn't have the inherent security problems that containers have because right now containers, you don't get 100% isolation, and you can actually break out of a container and infect the hardware below. So I don't know. Maybe this is a good play.

Steve: The EFF wrote a open letter to the President and CEO of Hewlett Packard, 1501 Page Mill Road. And I'll just sort of read the beginning because this explains what the letter's about. And this was to Dion Weisler. "Dear Mr. Weisler, I write to you today on behalf of the Electronic Frontier Foundation, a nonprofit devoted to defending technological freedom, human rights, and privacy in courtrooms, legislatures, and online. Like many others, we are alarmed by reports that HP has activated a dormant feature in Officejet Pro printers, and possibly other models, so that the printers now automatically verify whether its ink cartridges are official HP ink and not competitors' products or even refilled HP cartridges. If these printers detect third-party ink, printing stops. This activation was disguised as a security update.

"You must be aware," they write, "that this decision has shocked and angered your customers. Below, I have set out our concerns and the steps HP must take to begin to repair the damage it has done to its reputation and the public's trust." And I'll just - there were a number of bullet points that they elaborate on. But, for example, the first three were "HP deprived its customers of a useful, legitimate feature; HP abused its security update mechanism to trick its customers; HP's time-delayed anti-feature is a bait-and-switch." And so this was a few days ago. Their page was recently updated with the result of a major outcry. The EFF in their follow-on story said: "Over 10,000 of you have joined EFF in calling on HP to make amends for its self-destructing printers in the past few days. Looks like we got the company's attention. Today HP posted a response on its blog. Apparently recognizing that its customers are more likely to see an update that limits interoperability as a bug than as a feature, HP says that it will issue an optional firmware update rolling back the changes that it had made. We're very glad to see HP making this step."

So, and we've covered Microsoft's misbehavior. I'm trying to think if Apple has been caught doing the same things, you know, slipping some features in, calling it a "security update," and in fact doing things to people that they don't like. And very controversial, we know that Microsoft is in the process of, I think it's next month, actually, they're going to be switching over the way they handle Windows updates. But we'll cover that here in a minute. So, yeah. The good news is HP did this, and, wow. We've often talked about how expensive ink and toner are for printers. Basically it's the razor handle and razor blades model, where a lot of money is being made on the consumables.

FR. ROBERT: Right, right. And, okay. So the official policy line here is that they're trying to protect their consumers from third-party ink that might damage the printer. And that actually is true, that you can destroy a printer if you use really, really bad cartridges. That doesn't happen to most people. I've used reusable ink cartridges just fine with many of my printers. I did have one that leaked on me. I had to clean it up. And so it's this heavy-handed thing that you just don't trust that a company did it for your benefit.

Steve: Yeah, well...

FR. ROBERT: Even if they did, even if there is a clear benefit, you just don't trust them. A very good example, there's a 3D printer that I kind of enjoyed, the Da Vinci Jr. I've moved past it. But it used DRM filament. Now, I was able to bypass it, and what I found out was that third-party filament did destroy it at one point. So they were right. They were right in that they were trying to protect you from that happening. But it didn't matter. All you saw was the fact that they wouldn't let you do what you wanted to do with something that you bought. I mean, Apple it's the whole upgrading your OS and not being able to push it back. That's another thing. Yeah, it's better for the entire ecosystem. It's better for the company. But I think we as users, consumers, we get really upset any time a company says this is for your own good.

Steve: Well, and so, right. It was a security update inasmuch as it was for their financial security.

FR. ROBERT: Right.

Steve: And I think, I know that when I was up, I think it was New Year's before last, I fell in love with the Keurig machine that you guys have in the kitchen. And the point was made, though, that you wanted to make sure you got the right one because the newer ones were enforcing their Keurig's IP and not allowing non-Keurig K-Cups to be used with it. And so, but there it was caveat emptor. You knew upfront which one you were getting.

I think the extra controversial thing here was that - and this is one of the things that's got some people upset about Windows 10, also. Microsoft has reserved the right to change what it is, you know, unilaterally. And especially with the change coming in Windows Update. Users no longer have the opportunity to pick and choose what things they want to accept. It's like, no, here's the operating system. And it's like, but wait a minute, that's not the one I want. The one I wanted was last month and not this one.

And so what HP of course did was they, after the sale, they changed the terms of what their printer would do. And I think that's just not okay. If new printers refused anything other than HP cartridges, and refused to be refilled or to have those cartridges refilled, even if they were HP, it's like, okay. Then let's see how many HP sells compared to their competitors? But coming in afterwards and, I mean, I'm just glad HP backed off of this. I'm hoping other manufacturers saw that and take a lesson from it because it's not okay.

FR. ROBERT: That's actually a great analog, Steve, because when you talk about Keurig and their CRM, their Coffee Rights Management, that was one of those things where I know it sounded like a great idea in the board room. They're like, oh, okay, well, our patent's going to run out. Hey, how about this? We're going to make it so that our machines will only work with our coffee cups. So they have to buy them from us. And, well, of course people weren't going to buy a new machine that gave them less options.

Steve: Right.

FR. ROBERT: And so they saw their sales tank completely, not just sales of the machines, but the sales of their K-Cups, as well, because people were properly outraged that they had bought this machine, and it gave them less functionality than the machine they had bought prior. And so they returned them. I think you're right. If HP had two models of the same printer, and one sold for $199 and the other sold for $250, and the 199 one they told you, oh, well, you could only use HP cartridges with this, and the 250 one you can use with any cartridge you want, people are going to buy the 250.

Steve: Yeah.

FR. ROBERT: All right.

Steve: I mean, and if you don't know that yet, just look at the cost of cartridges. Oh, I mean, the printers are free almost. I look at this thing that you can get for 150 bucks, and it's like, how can they make this for $150? Well, maybe they don't. Or maybe they're not profiting from that at all. They're just counting on people buying their branded consumables.

Okay. So I just loved this. A number of our sharp-eared listeners were confused by something that I said last podcast, when I went into great detail about the problem that Yahoo! probably has with, we presume, an older hashing algorithm and the need to upgrade to a newer hashing algorithm. And I proposed a means by which that change could be made.

Well, the reason I confused people is that, some time ago, a couple years ago, I laid out the right way to do that, which I didn't do in the last podcast. In the last podcast I said, oh, well, they would have to hold onto the old hash until the user logs in with it, and then take the same plaintext and rehash it with the updated hash, and that would allow them to sort of ratchet the user forward. And years before I had said, oh, this is an easy problem to solve. You simply rehash the old hash with the new hash. So you take your existing database of old hashes. You hash all of those with the new hash.

Now you've got the strength of the new one, and you've transparently updated the database so that it now essentially uses a chained hash. But there's nothing wrong with that. And probably the old one wasn't very time consuming or there wouldn't have been a problem using it. So chaining them isn't going to hurt you very much. And so a number of users said, Steve, you know, you really - I thought you solved that problem a few years ago. It's like, ah, you're right. That's what I meant to say.

FR. ROBERT: Okay. As always, he meant to say exactly what he said when he said it, and now you understand.

Steve: Exactly, and now we're all confused. So, okay. This is the coolest DNS hack. We've been talking about dynamic DNS and the idea, for example, in the case of operating your own OpenVPN server, rather than using a third-party service, because it's got the problems of we're not sure they're not subjected to a government security letter. Any place that their tunneled traffic egresses onto the public Internet, it would be a high-interest location for any nation states or law enforcement. So the idea of distributing VPNs by not concentrating them on a single service, but rather now these days with the Ubiquiti router, with pfSense, with a number of other routers, you can run an OpenVPN server yourself on your home network.

The problem is most people use DHCP from their ISP to get their network's IP - which, while relatively static, can change from time to time. So then you want some sort of DynDNS, dynamic DNS, so that, no matter where you are, you're always able to find the IP address of your home connection, even if you're away traveling, and while gone your IP changes. Dynamic DNS is the famous means for doing that. Then we talked about a number of DNS services that offer that. And we've sort of been talking about this for a couple weeks now.

So, for example, my favorite registrar, which is now Hover - and, by the way, I should mention I think they're just becoming an advertiser of the TWiT network. So we're supposed to give a disclaimer, I guess, of that; but I loved them before, and I'll love them afterwards. They don't offer dynamic DNS. Namecheap does, Google's Domains offering does, and so forth. This hack is wonderful because we talked also about, I think it was week before last with you, Padre, about the service afraid.org, where you can create any subdomain of your own at afraid.org. So stevespiffydynip.afraid.org would be a domain name.

Now, the problem is, you may not like that domain name. It's like, well, okay, you know, it works, but I'm not sure I want to, you know, I mean, afraid.org? Really? So here's the hack. There's a record in DNS called a CNAME, stands for "canonical name." And what it essentially is, is an alias for a different name. I'm using that at GRC. I have a couple CNAME records. My blogs are steve.grc.com and blog.grc.com, but those are CNAME records over to wordpress.com domains that happen to be agilesynapse.wordpress.com and grcblog.wordpress.com. But users don't see those. They see and can easily remember steve.grc.com or blog.grc.com.

So anyone can do this with their existing non-DynDNS provider, like, for example, my favorite one, Hover. That is, if you already have a domain somewhere, then you can put a CNAME, a canonical name record, there which points to randomgibberish.afraid.org. And the DNS lookup process will fetch your formal public-facing DNS record, which is a nice, you know, like your own domain, and you can create a subdomain or whatever you want to do. And then, when the DNS resolver fetches that and obtains this CNAME record, it goes, oh, that's an alias for the real name. And then it goes and fetches this, wherever you pointed that CNAME record to, which has to be another DNS name. It's got to be a DNS record. It can't be, for example, an IP address.

That points it then to something.afraid.org, which does support dynamic DNS. So your device that you want to track as its IP changes, it keeps afraid.org updated. And then your actual DNS provider has a CNAME record pointing to afraid.org. Beautiful hack. So that was Angelo, and I can't pronounce his last name, S-I-J-P-T. So Angelo, he does this, and it works for him. So I just wanted to pass that on. I thought that was really a cool solution.

FR. ROBERT: And that is actually the proper way to do it. It will stay working throughout the time that you've got the CNAME set up properly. I will say, though, Steve, I kind of miss the days where everything was IPv4, and you could just memorize the IP. I don't know what it is. And you can't memorize an IPv6 address. That's just - it's too, I mean, that you actually do want to use a DNS or a CNAME entry in order to do that properly. But when you used to be able to know like the 50 IP addresses you need for all of your hardware, there was never a question of whether or not this was going to get to the right piece of gear. That was the only piece of gear that had that number.

Steve: Yup.

FR. ROBERT: Those days are gone.

Steve: Yup.

FR. ROBERT: Or mostly gone. You probably still - you have a couple of pieces of gear that are still IPv4-only; yes? Or have you dual-stacked everything?

Steve: No, my whole site is still IPv4. I mean, I'm still running on XP. So of course I'm still using IPv4.

FR. ROBERT: Okay, all right. I don't have anything still running XP, but I do have a few IPv4 addresses.

Steve: Well, there's the demand from people for ShieldsUP! to support IPv6. I actually have the equipment that I need. I just don't have the time. And everybody knows that I'm behind the eight ball, finishing getting SQRL wrapped up and then getting SpinRite 6.1. I'm not letting myself get diverted again onto any other major projects. And rewriting, see, because I wrote my own full IP stack for ShieldsUP!. I called it NanoProbe. And, I mean, I'm using WinPcap kernel filter, and I wrote everything on top of it. So it was fun, and I would love to do IPv6 support. Again, it's just a matter of time. So I'll get around to it once other projects are behind me.

FR. ROBERT: Indeed. And perhaps at some point I'll sell off the last of the IPv4 address space that I'm holding onto. I've got a little...

Steve: Wait, wait, wait, wait, how much? How much? What size?

FR. ROBERT: We had a /8 at one point.

Steve: Ooh.

FR. ROBERT: I know.

Steve: Ooh.

FR. ROBERT: I know, that's incredible. That's beachfront property. We don't have it anymore. We were good citizens.

Steve: Who? You and the Vatican? Is that what we're talking about here?

FR. ROBERT: Something. It was legitimate research. It was legitimate research. And we do have active traffic on a good 30% of those addresses. But, yeah, we turned back over a lot of it. We kept two Class B's and a Class C. But, yeah, we turned over the Class A.

Steve: The challenge is to find the right point in time for selling it because the curve is going to look like a hump; right? It's going to increase, IPv4 will increase in value up till the threshold of pain. And at some point IPv6 will just become more worth doing than the going price for an IPv4 block, in which case its price will start to fall. You'll be able to get less and less money for it because the world ultimately will switch over, and I'll be the only one still using IPv4.

FR. ROBERT: And actually it will be very noticeable. If you were to chart it out, there will be a major spike. And at that spike, people will realize it's just not worth it anymore, and then they'll switch, and immediately it will just plummet. Once half of the world decides, okay, I'm just doing IPv6...

Steve: We give up.

FR. ROBERT: We give up.

Steve: Yeah.

FR. ROBERT: Although I will say you're going to be making a security scanner for IPv6. What are the challenges of scanning ranges in IPv6 versus IPv4? Because, I mean...

Steve: Well, not a scanner because it's always targeting the client that is requesting the scan. And so it's scanning ports rather than scanning IPs.

FR. ROBERT: Ah, got it. That makes sense.

Steve: Yeah. And believe me, I mean, you can't - I know security researchers scan. But the point you're bringing up is absolutely salient because right now security researchers routinely scan the entire IPv4 space because it's very dense, most of it is in use, and compared to IPv6, IPv4, as we know, is a tiny little space. It's 32 bits rather than 128. And so your point is you can't actually scan all of IPv6. And of course I don't think people will. I think what will evolve is probably taking the BGP tables, they'll determine what blocks of this essentially infinite address space are actually in use, and they'll be scanning those.

FR. ROBERT: Right. And actually we ran into a problem not too long ago at one of the shows that we were doing. There was an enterprise-class router from a manufacturer in China, relatively big. You've heard about them. I don't want to use their name, though. But this was supposed to be their top-of-the-line enterprise router. They didn't put enough memory to dual stack the routes.

Steve: Ooh.

FR. ROBERT: So you could do IPv6, or you could do IPv4. But if you tried to do both, it couldn't handle enough routes for even its own internal network, much less routing BGP.

So we'll see. Some hardware will get a new life when suddenly it dumps its IPv4 stack.

Steve: Yeah, yeah.

FR. ROBERT: Now, Steve, this next one actually has me tickled a little bit because there is a member of the chatroom whom I very much value. So he's a good guy. He's an important person, intelligent person, sorry, who - he asked for a way to fix his Windows 7 machine because his updates weren't working properly. And what I told him was I had found a fix that involved you doing a single manual update. You couldn't just go to the Windows Updates program. You had to download something from the Microsoft site. You would do a manual update. And then it started working again. And he said that's absolutely bunk. It doesn't work. You have a way to fix this.

Steve: So, yes. We talked about this, I think three weeks ago. And this is a link that I've also got in GRC's Link Farm page, which is to a - it's in the Microsoft Community Forum. And they've been keeping this page current as the situation, which is a little bit fluid, is evolving. So there is an absolutely guaranteed solution for getting a new Windows 7 install updated. The reason I'm bringing this up again is that there's sort of some ominous wording in the most recent update to this page.

So the beginning of it reads: "Windows Update has become quite problematic for Windows 7 users. For the past year or so, we've been working to find a solution that will work for you. We have found one that works very well indeed for most. We know for certain this works well for September 2nd Tuesday, known as Patch Tuesday, which is today" - they were writing back then - "and onwards till October 7th. You want to ensure you get all your Windows updating done before the next Patch Tuesday on October 8th. From that point onwards, there will be a very significant change in the way Windows Update works. Some may like it a lot. Some may dislike it intensely."

Now, we've talked about what that change is, and that's that Microsoft is doing away with the granularity of updates. Essentially, they will be creating a monthly rollup that moves the entire OS forward. And their plan is, in the future, that rollup will go further and further back in time. And at some point it will be like the ultimate service pack which you apply as a single monolithic blob to the original install image, so that you would - and we don't know when they'll get there. But if you were to set up a new Windows 7 machine at that point in the future, you would start with your install image and then apply this one blob. Which, I mean, it makes sense when you think about it.

And I can empathize with Microsoft because it's so much easier to just have a single coherent update to all of the system files, essentially, all the things that they've changed over time. Frankly, as a coder, I have no idea how they actually implement selective updating because many of these things are making changes to the same modules, to the same files. And it's like, how do you figure out, how do you arrange to give the user the kind of choice we are spoiled with having up till next week?

FR. ROBERT: Well, a famous case of that was the .NET components, which were absolutely famous for failing out, and it would make Windows Update not work properly. And it was a convoluted process, and a lot of people didn't believe it worked. But I would always - I had this step-by-step thing which was essentially, get rid of .NET, and then install these patches in this order. And if you don't do it in this order before you do your next update, you'll have to start over. You'll have to erase everything again and start over. And I think that was actually - that's exactly what you're talking about. That's what you run into. When you give people granular control of updates, then sometimes they choose some, and sometimes they don't choose others, and now they're interacting on the OS level, and things don't work.

Steve: Yeah. And I don't know how they could. It's like, I just don't understand how it ever could have worked. Now, there are a lot of people who are upset. For example, during this whole Get Windows 10 fiasco, people were deliberately not installing that one GWX Update. And every month they'd say no, and every time it would come back, no, and so forth. And so they managed to get to now never having put it in because Microsoft gave them the option. And there have been other things, I mean, we've seen instances where a specific update will crash specific hardware. And so the user rolls back and then doesn't do that again because something about that update and their implementation or the state of their system, whatever, just is incompatible. Well, that's a freedom and a flexibility that we're going to lose.

So moving forward, I mean, again, I get it. I understand Microsoft wanting to do this. We don't have a choice. Moving forward, starting with 7 and 8.1, it's going to be more like it is with 10, where it's like you don't have a choice. These are mandatory. Otherwise you get left behind. You can certainly turn them off, but then no one wants to do that because they want the security improvements that are being made.

FR. ROBERT: Right. And I'm mixed about that. I mean, I actually do like the system of everything rolled up because I, like many other people, have had that experience of installing the fresh Windows 7 machine from a root image and then having to do literally days' worth of updates, pack after pack after pack, in order, in order to bring it up to snuff. This kind of simplifies that. At the same time, I have a really bad taste in my mouth right now with Windows 10 doing updates when I absolutely need it not to do an update.

Steve: And you mentioned last episode that you've rolled back from Windows 10 because it refused to respect when you wanted to give it permission to update. And talking about this monolithic-ness, we were just talking about the telemetry additions which Microsoft is moving back from Windows 10 into 8.1 and 7. Well, right now people know what those individual updates are, and they're rejecting them. They don't want that added to 7. They're going to lose the ability to do that.

FR. ROBERT: Right. I actually do have an experiment up right now. I have a Windows 7 machine, it's a root install off of an old Dell laptop I have. It's behind a firewall, so it can't talk to anything on my network. But it's just sitting there waiting for its Windows update. And when that updates, I'll know, okay, they've reached all the way back through all the versions, and they're now - this is every machine. So when that machine updates, it means every Windows machine that's eligible for this will now have it. And I'll let you know when it triggers.

Steve: So, okay. This is important. And I was feeling a little bummed because it seemed to be really bad news about Firefox, which is still my chosen browser. I know everyone knows Google's Chrome is now the winner at the moment in the browser wars. So here's the story that appeared to only affect Firefox. And I'm phrasing it that way because it turns out Chrome's no better. But we'll get to that in a second.

And paraphrasing from the report, Firefox users running on SSD mass storage should consider this must-change setting. Today's modern multicore processor systems and higher quantities of RAM allow users to open multiple Firefox tabs in Windows simultaneously. I've got two windows. I have something like 227 tabs open at the moment. I just sort of use them as infinite placeholders. I have tabs from the work I was doing on SpinRite 6.1 that are still open from when I stopped working on 6.1 to turn my attention to SQRL, until I get back to 6.1. So that gives you a sense of they're a little dusty. But they're there. They're my placeholders.

So these guys write: "This can have an unintended effect for those SSDs, as session store data is being written constantly to NAND memory, thus fatiguing it. Purely by chance, the author of this report fired up a free copy of SSDLife on two consecutive days on a workstation used only for email and browsing. For those unfamiliar with this tool, it reports estimated lifetime for an attached SSD, and it also shows the amount of data read and written.

In this case, SSDLife notified the user that 12GB was written to the SSD in one day. Since he didn't recall downloading any huge files over the previous day or visiting any new sites that could have resulted in bringing down a lot of new content to cache, this puzzled him. He monitored these stats over the next couple of weeks, and this behavior stayed consistent. Even if the workstation was left idle, with nothing running on it but a few browser windows, it would invariably write at least 10GB per day to the SSD."

FR. ROBERT: Wow.

Steve: Uh-huh. "Using Sysinternals Resource Monitor's disk utilization immediately revealed the culprit:

Firefox. It was continuously writing between 300K and 2MB per second to a file named 'recovery.js.' This is Firefox's session backup file, which is used to restore browser sessions in case of a browser or an OS crash or hang."

Now, there's a setting in Firefox you can change. If you go to, in the URL, you put about:config to bring up this daunting array, I mean, just a blizzard of settings. In fact, there are so many, you have to have a search bar, and so there is one. Then you search for "browser.sessionstore.interval." Even when I put in browser.sessionstore, I thought, oh, it probably narrows it down enough. No, there's still, like 25 different things. Anyway, ".interval." It defaults to 15,000, and that interval is in milliseconds. So that's every 15 seconds the default is Firefox will store its state. And if that state is a lot of large pages, it is redundantly, and I guess not very intelligently, not incrementally, it's just dumping its entire state constantly.

So this experimenter, in his case - what I did was I added a zero. So I dropped it from very 15 seconds to every 150 seconds. This guy set it to 30 minutes. Now, I have to say I appreciate that crash recovery. With the latest Firefox, where they've created a separate thread for the UI that was supposed to increase stability, for me, I'm having more trouble with it. It's not as stable as earlier Firefoxes were. So I'm not that happy with it. And so I'm having to restart it sometimes. And I'm happy that it recovers all those tabs. But, boy. And I'm using an all-spinning drive RAID for this workstation.

But this next machine that I've talked about, this Windows 7 machine with all that RAM, I'm going to seriously look at revisiting a RAM disk and see if it isn't possible to redirect Firefox's state store, its crash snapshotting, over to a RAM disk because an SSD is my primary drive on that system, a nice one. But still, I don't want it just sitting there chewing up the SSD.

So anyway, so this guy says: "Bottom line is that, if you have a lower capacity consumer-level SSD in some of your machines, you may want to check and tweak your Firefox config. Those drives can be rated for about 20GB of writes per day, and Firefox alone might be using more than half of that." And, like, for no reason. I mean, doing an insane amount of redundant writing, essentially. I'm hoping that this is going to put some pressure on the architects to make this incremental because it's ridiculous that, leaving it alone on an idle workstation, every 15 seconds it saves the entire browser state nonincrementally. It's just poor engineering.

They finished saying: "This is especially true if you're like [this guy] and have several browser windows open at all times." I haven't looked at mine. I shudder to think how much my system is writing redundantly to hard drives. "Changing this parameter may even help with normal hard disks. Your machine will feel faster if it doesn't have to constantly write this session info. Users have observed that content open in the browser does have a major impact on writes" - so not just how many tabs, but how big the tabs are, that is, in terms of how large the page is that the tab is holding - "as does the number of open windows and tabs. If you are using Firefox and a lower write endurance SSD, you should check this immediately."

And then, in an update to this, they said: "We are now testing other browsers. Currently in the middle of a Chrome v52.0.blah blah blah blah blah test. We have been able to see a pace of over 24GB per day of writes on this machine." So they're seeing a gigabyte per hour under Chrome. So it's not just Firefox. It's both of those browsers that are not intelligently saving their state. And it's interesting because I have noticed that, when I do a restart, sometimes - in fact, I have an add-on, a Firefox add-on that's like a little green curly arrow which is a demand restart. And sometimes its use of memory just sort of grows.

And so it's like, I'd say, okay, and I push the button, and it does a Firefox restart. It takes it out, I get back three quarters of my machine's memory, and then it creeps up again. The point is that I've noticed that, if I then touch, if I open a tab or switch to a tab that I haven't looked at since it restarted, it brings up the old page's content. And I thought, oh, isn't that interesting. Where is that? Well, now we know. It's sitting in this monster file which it's been redundantly writing. But it didn't load it back into the browser until I actually viewed it. But anyway, so I think that's a really valuable tip, especially for people who are on SSD-based machines. A lot of laptops these days...

FR. ROBERT: Oh, yeah, we're all on SSDs now.

Steve: Yes, yes.

FR. ROBERT: The scariest part about this is how many of us have desktops that are on all day, and we leave our browser open? I do it all the time. And now I need to go in and find out what it's doing.

Steve: Yes. It's your portal to the Internet. I mean, I've got two browser windows open statically. It's now part of my environment is the browser. And it's sitting here just chewing up the hard disk. So again, Sysinternals Resource Monitor. And there is a disk utilization feature there which shows you by process how much what processes are using your disk. Anyone who's interested who is worried or curious about this, spin that up and see what your browser is doing. The good news is, for Firefox, there wasn't any follow-up yet for Chrome. But for Firefox we can slow that way down.

FR. ROBERT: And I will say that, if you want to see what damage may have been done already, all of the major SSD manufacturers - Samsung, Intel, Kingston, I think even Toshiba maybe, they've got utilities that you can run that will actually show you how many writes have been made to the cells, on average. Assuming, of course, that they're doing trim properly. And I know that most modern drives will safely do between two and 3,000 writes per cell. Some of the Enterprise drives will up to 10,000 writes per cell. And that should give you a good clue as to how much damage has been done. I really want to check that now because I've literally had a desktop on for about 18 months with browser tabs open at all times. I don't know.

Steve: Yeah. Yeah.

FR. ROBERT: Steve, we've got a little bit of errata that we need to take care of.

Steve: Well, last episode we were talking about corpuses. And I said, wait a minute. Should that be corpi? Yes. And I got two responses. Someone tweeted me "corpora," and then a friend of mine who is bit of a linguist, he just sent "corpora, because third..."

FR. ROBERT: Declensions.

Steve: "...third declension neuter." It's like, okay, John, yes, of course. Who doesn't know that?

FR. ROBERT: You are bringing me back to really dark days of learning Latin during my master's in divinity [crosstalk].

Steve: Third declension neuter is why it's corpora. So, okay. Corpora.

FR. ROBERT: I prefer "corpo." It sounds friendlier.

Steve: That does sound very corpo. It sounds like something you'd want to order a large size of.

FR. ROBERT: Sir, can I get the large corpa? And a beer. Let's do that, yeah.

Steve: So a couple little bits of miscellany. Our listeners know that I've been working on a Healthy Sleep Formula for a few months now. Actually, I began working on it almost a year ago. It was the end of October of last year. So we're coming up, so it's 11 months. And a lot of people have been using it with great success. One of the problems has been that immediately upon putting up the ingredients that it uses, they have been selling out, those that are not heavily sourced on the Internet. I wanted to make sure everyone knew that the niacinamide that has been impossible to get for the last few months, as promised, by the end of this month is coming back. Both iHerb and Swanson Vitamins are now restocked with that niacinamide. And I found two others that are lower dose and smaller pills. Some people were having also problems with the size of that. So just an FYI.

I'm going to - I'll find some time to update the page. I haven't for a while, although I have been keeping it up with that news. My plan is to wait until, like, all the way through October, and then in early November ask people explicitly to send me their feedback about their experiences with it and dosages and so forth. It's only now that people have been able to purchase all the ingredients reliably. So I want to give people some time, finally, now that they're able to get it, to use it. And then I will incorporate their feedback into a page where we begin to generate some user experience and guidelines. But it's been a hit, so I'm really pleased for that.

Everybody also knows that I love puzzle-style toys on iOS and Android. One of our favorites, which disappointed us only because it only had 50 levels, was called Hook. And then another puzzle succeeded it, different, called Klocki, K-L-O-C-K-I. They've just both been made free. So if anybody didn't get them before, I don't think they were ever very expensive, like $0.99 or something, but they are now, both Hook and Klocki, are free for iOS and Android. So I wanted to give people a heads-up.

And also, unfortunately, this podcast airs on the 4th, and HBO will have premiered their

new "Westworld" series on Sunday the 2nd, two days before this podcast. But they'll be - I'm sure they'll be reairing it throughout the week. So I just wanted to, again, give people a heads-up. The previews look wonderful. I don't know how it could have 100% on Rotten Tomatoes when it hasn't yet even aired, but somehow it has. And there are reviews available over at IMDB and elsewhere. So it's a remake of the original, well, it's based on the original Michael Crichton novel, "Westworld," which was Yul Brynner, famously, the sort of defective robot gunslinger. And it looks really interesting. So for those of us who enjoy science fiction and have access to HBO, it looks like that's going to be fun.

And I just made a little note here about SpinRite. We actually did a podcast yesterday, and I didn't dig around for any other feedback from users. So everybody knows about SpinRite. So we'll just move on.

FR. ROBERT: Well, before we move on about that, I will say that last night I ran SpinRite on my OnePlus One, as was suggested, because I can use my phone. There's a setting for me to make it a mass storage device.

Steve: Right.

FR. ROBERT: And it worked. Now, I wasn't able to complete it because I needed to come in to work today. But now that I know that it works, now that I know that it can access it, that's something I'm going to kick off once I land in Rome because I can't wait to see if it will get me the speed back that I've lost over the years.

Steve: Well, and I did hear from some other people who watched us live in the last podcast who said that they're seeing SpinRite speeding up dodgy SSDs all the time now.

FR. ROBERT: Right.

Steve: So it's interesting that, I mean, apparently SSDs get slow for retry reads or something, like very much the same way that spinning hard drives do. And SpinRite does successfully speed them back up again. So, yay!

FR. ROBERT: Now, I know a little of the mechanism behind that, Steve. I understand that, first of all, there's the garbage collection.

Steve: Right.

FR. ROBERT: And anytime you do garbage collection, you're helping the SSD perform better when it's actually doing active transfers. But the other part, and this is the wonderful physics of this, every cell is actually, it's a capacitor. It's a little battery that stores a charge that lets the controller know if it's on or off. And over time that charge will diminish, and it gets more difficult to tell the difference between an on and an off. Level 2, does it actually go to those cells that could be on or off and recharge them and force them, so it makes sure that the on are really on and the off are really off? Or what's the mechanism behind that?

Steve: Okay, so no it doesn't because that would be writing. And we know that that would be the fatiguing the drive.

FR. ROBERT: That would be bad.

Steve: Yeah. So what it does is it reads it in a way that - because there's lot of extra

commands down in the ATAPI API. So it's able to put it into a maintenance mode, which makes it less - it makes it more finicky. And then when it reads it, the SSD looks at the amount of error correction needed. And only if the cells are becoming weak, so that more error correction is necessary, then selectively the SSD will rewrite those problematical areas.

FR. ROBERT: I love that.

Steve: So it does, it just, like, it does what you want. It won't rewrite it unless it needs to in order to fix it on the fly.

FR. ROBERT: Now, what mechanism did you write in so that it knows when the cell is in trouble?

Steve: Well, that's built into the SSD. So both hard drives and SSDs rely heavily on error correction code. And there's something, the error correction process, when it realizes that it got a bad checksum, which is the fast thing to do, then it falls into error correction. It uses the math of ECC to locate the region which is in trouble, and it looks to see how bad it is. The actual term in error correction logic is the syndrome. The syndrome is an XOR mask of the problem area. And the number of bits from the first one to the last one of the mask tells it how large the error is. And if it starts getting too large, then it says, okay, I'd better fix this before it grows out of my ability to correct it mathematically.

And so it actually has, it's like the drives are deliberately tolerating some persistent correction and then making sort of a value judgment about, okay, is this bad enough that it's worth rewriting this in order to fix it? And so basically - but here's the key. The drive doesn't go and police itself. It only knows there's a problem when you ask it to read a sector. And if that sector - you were talking about the capacitors, the charges stranded on little electrostatic islands. If it's been a long time since you've visited that sector, then that can have become weak to a point that it will no longer read. So what SpinRite does is just, very methodically and in a linear fashion, so it's as efficient as possible, just move through the entire addressable area of the medium, whether it's a hard drive or a nonvolatile NAND storage, and just give the NAND controller or the hard drive a look at its own storage. And it fixes what it needs to.

FR. ROBERT: Fantastic. All right, Steve. I would love to talk more about the tech because that's who I am, and I love when you explain things. But if we don't get into Q&A we're going to have yet another episode where we have to kick the Q&A down the street. This first one, it's a little interesting. Tell me about how am I going to handle DNSSEC on Windows 10?

Steve: So, yeah. I got an interesting question because we were talking up DNSSEC in the last couple episodes, and you and I are both bullish on the promise of having secure DNS. Normally DNSSEC is not something that the client does. It's typically something that the web, I'm sorry, that the DNS server the client refers to uses to verify that the records it has received are accurate. However, it can be pushed down to the client. And, for example, Windows 10 does support DNSSEC. That is, Windows 10 can be instructed to itself verify the signatures on records that it retrieves. It's not the default setting.

And the problem, of course, and this is in general the problem with DNSSEC, it's sort of an all-or-nothing proposition. That is, either we tell our client to not trust any non-DNSSEC record, or to go ahead and trust them, I mean, like not to check. And but the problem is, not everybody is signed. I'm not. GRC, as I said a couple weeks ago, I have not signed my site because the nature of it is I'm constantly changing things and

tweaking it. And the tools that are available, just they make that a little problematical at this point.

So the way I would phrase it is, it's good that Windows 10 supports DNSSEC. There are other clients for Windows that you can add which do support DNSSEC. But very much like IPv6, where the infrastructure is still being laid down, the DNS roots are being signed. And we talked about how they're being rekeyed from 1024 to 2048 bits. And but we need then the other DNS servers to get signed, and for it to be supported universally. Until it's supported universally, then we're not going to be able to insist on it. And we see this in many instances.

We were talking about OCSP last podcast, the Online (SSL) Certificate Status Protocol, where because it's not universally available, it's a soft fail if you can't get it. Well, we'd like to say hard fail, but then you end up with problems because it's a little spotty still. So we're still sort of in the same place. We're moving forward, but making fundamental changes to the entire infrastructure. It's just a slow process, like with IPv6.

FR. ROBERT: I think the comparison with IPv6 is dead-on because this is one of these technologies where it's good to have it on. It's good to have your machines understand IPv6 and understand DNSSEC. But until everyone is willing to turn the key, it's not going to be that much use. Like right now, your router is probably dual-stacked, and you're probably not using the IPv6 portion of it.

Steve: Right.

FR. ROBERT: All right. We've got one here from Wildkarde. Oh, and by the way, the last one was from Lockpicker, @AspiringLockpick on Twitter. This one's from WildkardeUK, I guess. I love the spellings here. He wants to know the difference between HTTPS and HSTS.

Steve: So a little bit of acronym soup there. I think I - he says he just finished watching Security Now!. And I referred to HSTS. And he was saying, wait a minute, what's that versus HTTPS? So HSTS is HTTP Strict Transport Security. So it's kind of a superset, although they're different things. HTTPS, of course, is HTTP Secured, HTTP SSL or TLS, where you exchange a certificate with a server to get both authentication and encryption privacy of your communications.

So there are [audio dropout], though, with various ways attackers have of intercepting pages which the site, the website intends to be secure. But if somebody, a so-called man in the middle were to intercept the communication, for example, and in all the URLs on the page, if they removed the s: from the end of the https: so that they essentially turn them into http's, the browser would just assume, oh, okay, this site doesn't use HTTPS. That is, it's possible, even if the server wants to enforce security, because of the way this evolved, the browser might not know that.

So Strict Transport Security, just like its name says, HSTS, is a later add-on to the HTTP specification, to the web spec, which allows a site in the response headers so that, when the web browser asks for a page, the response headers contain an HSTS header which explicitly says, for the following X number of seconds, this server will always be HTTPS. That is, it allows a server to assert its intention in the future to be strictly HTTPS. And that information the browser stores.

So the idea is, if you ever visit a website, and it's HSTS, then it's able to send you a header to educate your browser so that the site says we are always going to be secure.

And if you ever receive a nonsecure URL, you have permission to upgrade it to HTTPS without even asking. If you see any HTTP URLs from this site, promote them to HTTPS. So that solves a really nice problem. It then means that subsequent visits cannot be usefully downgraded by a man in the middle.

Now, smart people recognized, oops, there's still a little loophole here. And that is, what about the first time? Again, this is the sort of problem you get when you add things on after the fact. So there is that first visit problem because it's on the first visit that you get that HSTS header that the browser can then store permanently.

So the various browsers have gone a step further. Years ago, when GRC committed to being HTTPS, I submitted both www.grc.com and grc.com to Google so that it would be built into the browser. And there's a shared list that Firefox also uses which now wires a growing number of website domains as preset to HSTS so that that even solves the first visit problem. Anybody, for years now, who uses Chrome, if they'd never been to GRC ever, and some bad guy tried to demote the URLs, Chrome would ignore that and know that GRC was HTTPS and have the flexibility to upgrade any query right off the bat. So a very cool sort of incrementalism that we're seeing as we move forward, increasing the security of the web this way.

FR. ROBERT: Fantastic. All right. We've got another one here from Ian Anderson Gray. Can I just read it out? Because it's actually pretty insightful.

Steve: Yeah, cool.

FR. ROBERT: He says: Hi, Steve. I wrote an article on how to broadcast to Facebook Live from your desktop: iag.me/socialmedia/br. It's become very popular with over 600 comments. Most people are able to connect without any issue, but it seems that some people can't. I have a feeling that some people's ISPs are blocking the connection. Facebook Live uses RTMP, and I think this is port 1935. In order to help less techie people, what's the easiest way to check if their ISP is blocking port 1935, and what can they do to get their ISP to unblock it? Could there be another issue? Thanks so much for your time and Security Now! I never miss it. Ian.

Steve: So that was a great question. And he and I actually exchanged DMs back and forth for a while because I was sort of curious to try to come up with a solution. And I had a couple of ideas. So we've talked about how ISPs are preemptively blocking packets for known troublesome ports. Many ISPs block port 25, which is the SMTP port, because there was a - and the problem's sort of gone away, maybe because they've just gotten so good at blocking them. But spam used to come from users' infected machines, and port 25 was the SMTP port, so they would block that. And in some cases ISPs just want you to use their own SMTP server so they don't let you get outside their network. You have to use their server so that they have some control.

And then, of course, the famous Windows file and printer sharing ports, 137-139 and 445, also generally blocked by ISPs. And that's a service they provide, sort of, because if anybody had their filesharing open on the Internet - and Windows machines are often, historically have been. Now they've got firewalls that are turned on by default, so things are better. But this was what the original impetus for ShieldsUP! was back in the day. I scanned the region around my business's IP when we first got a DSL line. And here were all these C: drives, like just open. And I thought, okay, I have to make the world aware of this. So I created ShieldsUP! specifically to allow people to check to make sure that their C: drive wasn't available to everyone on the Internet because for a while they were. Those were the Wild West days.

FR. ROBERT: Those were the days, where you could send people something, and the Microsoft Messenger protocol, and just have a message pop up on their screen.

Steve: Yes.

FR. ROBERT: That was actually - that was fun. For pranking, that was actually a lot of fun.

Steve: So now ISPs are proactively blocking the more troublesome ports. So there's at least the possibility that this RTMP protocol, which he was right, it is port 1935, and it is over TCP by default. There is a UDP variant, but it's not very popular. So there's a chance that some ISPs might be blocking it. The question is, how does he know? And this is sort of a nice broader question for anyone, like how could you detect what ports your ISP is blocking, if it's doing so? And so there are a couple ways. One is, if you have a configurable router, which you can tell not to drop packets, but to reply with an ICMP message, the formal proper behavior for an IP stack device is, if a packet comes in trying to open a connection, and there is nothing there to accept a connection, the proper behavior is to politely say, sorry, there's no service running on the port you just requested. So it sends back an ICMP saying, eh, no.

Now, the problem with that is it makes you probeable. That is, scanners will know you're there. So one of the other things that ShieldsUP! I think really popularized, as far as I know I coined the term "stealth." Of course I used that from ShieldsUP! and Star Trek and so forth. Because in stealth mode the router doesn't reply. It just drops the packet. So somebody scanning across an IP region would not know there was anybody at that IP. If you can change your router to disable stealth temporarily, you can then use ShieldsUP! to scan all of the service port range from port 1 through - I actually go a little bit above 1024, and you'll get this beautiful grid.

Now, normally the grid is green, meaning all stealth, because your router is dropping the packets. If you disable stealth on your router, it switches to blue, which is the port is closed, except where your ISP is dropping the packets, before they have a chance to get to your router. So if you switch your router to respond - and you can immediately tell if it works by going to ShieldsUP! and scanning yourself. Rather than this normal field of green, you'll get a field of blue with some green spots. Those are the ports which your ISP is blocking for you before they got to your router and gave it the chance to say, yeah, I'm here, but there's no port open at that port. And 1935 is above that range.

But ShieldsUP! also has custom port scans, so you can scan custom ranges or even individual single ports. So once you confirmed that your router was responding by getting a field of blue with some green spots, then you could check, you could have ShieldsUP! scan 1935 and see if it comes up blue or green. If it's blue, that means that that scan from the public Internet, from me, from GRC, got all the way to your router and then bounced off of it and came back showing that your ISP is not blocking. If it shows as green on port 1935, then your router never received the packet, and something between the public Internet and your router, probably your ISP, arranged to drop that. And so that allows you to check to see whether incoming packets are being blocked.

There is a chance, though, because the way RTMP operates, it is an outbound connection. That is, your RTMP client is initiating a connection to port 1935 on some remote RTMP server. So how do we check for that? Well, I have another little piece of freeware called ID Serve - I-D S-E-R-V-E. I wrote it years ago as a simple way to identify, to ID a remote service because many services in a response give you a whole bunch of information in their reply headers. So it's a very simple, it's super light, small - of course I wrote it in assembly language - cute little Windows app and also runs under

Wine, which allows you to put in a domain name or an IP, and it will issue a connection to that and show you what it got back.

You can override the port that it goes to. Normally it goes to 80. So if you wanted to check a TLS connection, you'd put in :443. If you want to check RTMP, you'd put in the domain name or the IP :1935. It will then attempt to initiate a connection to that port. And its response will tell you if it was able to connect. And if it did, what the reply looked like. So there are some simple tools here that, with a little bit of jiggering around, you can actually answer the question of is traffic on any given port able to actually get to you. And are you able to initiate outbound connections to any given port.

FR. ROBERT: I used to have a server back in the day that I kept hidden away from the public. And I could turn it on, so it had an out-of-band management, so I could turn it on and off as necessary. But it had all ports open. And that was just so that I could run NMAP on my side and ping through and just see what was being dropped by my carrier.

Steve: Ah.

FR. ROBERT: It was important because I spent a lot of time in convention centers where they always had some weird rules, and you never knew who was responsible for blocking a port. Of course, you'd never do that today. You would never put a server on the Internet with ports open like that. But back in the day it worked.

Steve: And of course we've also seen situations, for example, where you have captive portals, like you're in a Starbucks WiFi, and they're doing things like blocking VPN ports and so forth. So it can be interesting to get a snapshot of what traffic is able to get out and in from, as you said, Father, from a location that you're not familiar with.

FR. ROBERT: Right, right. Right, let's move on. We've got another one from, I'm going to say it's either Stijn or Stijn Crevits. We'll see. He says: Hey, Steve, good news. cPanel does in fact support Let's Encrypt now. But they need to fix a bug where only admins can configure it. One issue with LE's auto-renewal, however, is that you can't use HPKP, which is, again, we're getting into alphabet soup here.

Steve: Yes.

FR. ROBERT: What is HPKP?

Steve: So we've never talked about it. It's yet another incremental extension. It stands for HTTP Public Key Pinning. So this is cool. And so this is like - this is sort of off-topic for the question, but I just wanted to mention it because it is a cool feature. We know that certificate pinning provides clients with a means of affirmatively verifying, sort of a priori, the identify of a certificate. I have the GRC Fingerprinting page, which allows users to check the fingerprint of certificates and compare them with the ones that GRC receives and see if they're getting the same certificate that GRC is getting in order to potentially detect a decrypting proxy in line where they'd be getting the proxy certificate rather than the actual server certificate.

So another way to do this - and of course Chrome does this for Google's certificates. This is the way Chrome is always able to catch anybody who screws around with Google's certs, is that Chrome knows the fingerprint, the actual hash of all of Google's currently-in-use certificates. And that's called "pinning," where instead of relying on the trust in the CA system, which is sort of one level of indirection, and we've talked about the various ways that can fail, instead it's, okay, we know what certificates are valid for Google.com.

Here's the list of the hashes of those certificates. And so that's called a "pin."

Well, HPKP, in the same way that HSTS allows a site to assert that it will always be using HTTPS security, HPKP is a means in the protocol for a site to give a browser a pin, that is, to pin its own certificate. Which, as with HSTS, the browser caches for a length of time that the site has said this pin will be valid. So it's another very cool, sort of we're creeping along, making the 'Net more secure, looking for things that are still hackable, and coming up with ways around it. And so this is a way for a site to provide browsers with the precomputed signatures of the certificates it will be giving them.

And so, for example, if you had visited a site that had given you a pin for its certificate, then you went somewhere else, you went to work maybe, where you are behind a proxy, or a proxy was brought up without your knowledge. Your browser would immediately complain, saying, wait a minute. This certificate that I have received, even though it looks otherwise valid because, for example, someone had snuck a CA for a dynamically issued certificate into your machine store - which can be done through group policies, for example, for administration. If that happened, then the browser would say, wait a minute, this certificate's hash does not match the one that I previously received from the site, and the pin is still within its valid, non-expired time. So alert, warning. And so you get advised.

So anyway, I did want to - I wanted to note that cPanel, which is the widely used control panel for hosting providers, is now supporting Let's Encrypt. And more things are supporting it moving forward, which is just great news.

FR. ROBERT: Any time my service provider wants to give me more security, I'm down with that.

Steve: Yeah.

FR. ROBERT: All right. We're on a role. Let's keep it going. Let's see if we can actually get through a good number of the questions here. We've got Chris A., who says, Steve, love Security Now! and SpinRite. I would love to follow your advice and offer TLS on my personal website, but there's a problem. Let's Encrypt is hosted by the Akamai CDN, and I trust Akamai about as far as I can throw a datacenter. And it says, "More on that if you wish." That much lauded DigiCert doesn't appear to have joined the free cert party yet, and other free cert issuers have had numerous problems, as you've recently detailed on the podcast. Is there a free cert issuer other than Let's Encrypt that you could recommend? Thank you for all you do. Chris.

Steve: So this was an interesting question because I took it from a different angle. I said to myself, okay, he doesn't trust Akamai. What evil, what wrong could Akamai as the carrier of Let's Encrypt do? And I can't see any. The protocol protects you from the delivery system's misbehavior. So Let's Encrypt is hosting in order to scale, that is, they're hosting on a big CDN in order to scale. And so they've got servers which Akamai's network finds.

So Akamai, we're used to thinking of them as a caching system. But in this case they're not doing caching because Let's Encrypt is not a caching-based solution. It is a dynamic client-and-server protocol where the client and server have a well-designed state-of-the-art handshake that allows the client to assert and prove its control over a domain, and for the server to then issue a certificate which has been signed by a CA that the client trusts.

So I'm going to throw this back to Chris and say, you know, sorry you don't like them, but I think it's okay. All that Akamai is doing is helping the Let's Encrypt system scale. We talked about anycast routing last episode, about the idea that a single IP can appear to be widely geographically distributed. So that's what's happening here. And due to the nature of it, you don't have to trust Akamai. You're actually trusting Let's Encrypt's servers. And no man in the middle, I mean, this is MITM-proof. It was designed to be robust against that attack, specifically because it needs to be. And so Akamai is the man in the middle, as the deliverer, as the conduit of the content. And there's nothing they can do to screw it up.

FR. ROBERT: But I understand his sentiment here. If he doesn't trust a carrier, a service provider, and if I don't trust a carrier, a service provider, I don't want any of my services to touch anything that might even be remotely connected. It might not just be a security thing, too. It could also be I don't want to support this company in any way, shape, or form.

Steve: Okay. That's a good point. I would say consider all the things we don't trust between here and sites that we do. Like a lot of people don't like Comcast, you know, their ISP. Do they trust them? No. But they don't have to because they're just a carrier of the traffic which is being authenticated and encrypted outside of their ISP's control. The exact same is true with Akamai. So I appreciate what Chris may be saying, and I think you make a good point, Padre. But from a technical standpoint, there just isn't anything to worry about. Well, okay, with the caveat that there's always something to worry about.

FR. ROBERT: Yeah. We always worry about something. But you know what, there are other things that are more important to worry about than that.

Steve: Yes.

FR. ROBERT: Right, right. We've got Ben Moore here, who says: Steve, thank you for all that you do. Of course I think we all have that sentiment. In 2015 I experienced a compromise of my eBay account. Turns out eBay has a "feature" where you can have a password reset code sent to your mobile phone by text message, and within the dialog box you can select "Try a different number" if you no longer have access to the phone number associated with your account. After the compromise, I got everything sorted out. However, now in 2016, my account was compromised again - oddly enough, by the same phone number as last year.

So after resetting all my passwords - easy enough - I discovered that somehow the phone number used to reset the password is still associated with my account, but cannot be corrected or deleted. After a few very long calls to support to correct, the final one ending in me demanding a return call with a resolution - still waiting after a month for a call back - the phone number is still associated with my account.

Here's why I'm writing: Have you, in your experience, found a way to correct this sort of problem? Is my account, which has been open since 2001, to be forfeited? Can I better protect myself? I'm hoping your experience can assist, or your exposure might be able to highlight the absolute failure on the part of eBay to be "able" to alter account information for security purposes or, frankly, willingness to help keep customer accounts secure.

Steve: That was a great question. And I replied to Ben already, saying I can't think of anything you can do except to forfeit your account, even adding multifactor. What eBay is doing is saying the phone number is your ultimate solution for proving your account control. And it's crazy that it isn't possible to revoke previously issued phone numbers

that you no longer wish to have associated with your account. The fact that he can't through the UI, and the fact that he can't through many conversations with them, says they've got a problem with their system.

But if Ben wants security, and he clearly does, I think you just have to drop that account and create another one. And I can't think of, I mean, like in fact you'd have to drop it in a way that somebody who then compromised it could not do any ill under your name. So, like, change all the information, remove credit card information, remove your first pet's name and all the other nonsense that anyone who got access to it could abuse. I think you just have to abandon it. And I did want to give this a little bit of light because it's wrong that this is the way eBay's management is set up for this kind of account recovery. It's crazy.

FR. ROBERT: And, oh, I want to get your thoughts on this. Like you, I believe eBay is wrong. I believe the system is backwards. There needs to be some way to update the way that you reconfirm an account or you get an account back. However, barring that, so barring the fact that they've messed up on the implementation of security, it is something to be lauded for a company not to surrender credentials to someone who they think might be the person who owns the account.

Steve: Yeah, that is a good point. I mean, if a bad guy was able to remove the good phone number, then you lose your ability to recover your own account.

FR. ROBERT: Yeah, exactly, exactly. I mean, this would be another story if it was someone who said, "Someone called up eBay, and they pretended to be me, and eBay gave all my information to them." That would be horrendous, as well. So there's always that fine line of a company sticking by its security principles and willing to work with customers. Although in this particular case, as you mentioned, the fact that he can't update that phone number, that's a fail. That's a big fail.

Steve: Yeah, that's the problem, yeah. Padre, we're at two hours.

FR. ROBERT: I think so. We've actually hit the two hours. This will be the first time I've ever done a show with you where we actually hit two hours and didn't go over it grossly. So go figure. Steve, it's been a pleasure. I am, of course, I'm happy to have Leo back. I love working with Leo. I love having him in the office. But it will be a little sad that I don't get to do Security Now! with you anymore.

Steve: I know that our listeners feel the same way. I always see a lot of tweets saying, hey, you and Padre have a great chemistry, love listening to you work together. And I know that we both feel that way about each other. So thank you very much for the last three weeks.

FR. ROBERT: Indeed, thank you. Thank you for working with me. And of course Steve is the skipper of this two-hour tour. Steve Gibson can be found at GRC.com. He is the author of ShieldsUP!, SpinRite, SQRL. He is the man that we turn to whenever we have a question about security, about best practices. And of course he's got audio versions and transcripts of every show at GRC.com. So head over there if you need to get your fix. Of course you can also get, at our show page, TWiT.tv/sn, you'll find all of our episodes plus a place to subscribe, as well as being able to download the podcast whenever and wherever fine audio podcasts are aggregated.

We do Security Now! here on the TWiT.tv network every Tuesday at 13:30 Pacific, of course unless we're prerecording them. And you can watch us live at live.twit.tv. As long

as you're watching live, jump into our chatroom at irc.twit.tv. I feel that it's actually a big part of the show. I love seeing people respond, comment, ask questions in real time. Again, that's live.twit.tv and irc.twit.tv. Until next time, I'm Father Robert Ballecer in for Leo Laporte, saying get out there and get some Security Now!.

Steve: Thanks, Padre. Always a pleasure.