SECURITY NOW!

Transcript of Episode #523

uBlock Origin

Description: Leo and I catch up with the week's major security events. We then examine the ecosystem of web page advertising by comparing it to other "opportunistic advertising" such as that appearing on public transportation, highway billboards, broadcast television commercials and other public venues - which consumers have no obligation to consume. I eschew the implication that visitors to a web page have an obligation to retrieve third-party content, over which the website has little or no control, which consumes bandwidth, reduces online privacy, hinders performance, and potentially exposes visitors to malicious exploitation. And I believe this remains true even when a visitor's retrieval of such despicable third-party content would generate much-needed revenue for the visited site. Finally, I review the many operational features of uBlock Origin, my chosen HTML firewall, which effectively returns control to web users.

High quality (64 kbps) mp3 audio file URL: http://media.GRC.com/sn/SN-523.mp3
Quarter size (16 kbps) mp3 audio file URL: http://media.GRC.com/sn/sn-523-lq.mp3

SHOW TEASE: It's time for Security Now!. Steve Gibson's here. This is the episode I've been waiting for all week. He's going to cover adblocking, why he thinks we need adblockers, and, I think, I'm guessing, his favorite ad blocker to date. Actually, I like it a lot, too. uBlock Origin, how it works, how to work it best, coming up next on Security Now!.

Leo Laporte: This is Security Now! with Steve Gibson, Episode 523, recorded Tuesday, September 1st, 2015: uBlock Origin.

It's time for Security Now!, the show where we protect you and your privacy and your security. And of course there's no better guy to do it than Steve Gibson. I say it with love: He is paranoid. And that's what you want in a security expert; right? Hello, Steve.

Steve Gibson: Hello, my friend. So we had another - we had a modest news week, so the industry is cooperating with us.

Leo: Yay.

Steve: Which means that we get to give really full coverage of one of the tools that I think, I know you have been loving it since we talked about it last week and let our listeners know this is where we were going to go. I would imagine that a bunch of them have adopted it, too. And that's uBlock Origin is the name of an add-on which is cross-platform, cross-browser. And today we do a deep dive into its history and background because there's a lot of confusion about that due to some recent changes in ownership.

And in fact, as I was putting notes together, I thought, how could I describe its author, whose name is Raymond Hill? And I thought, you know, I've read everything he's written that I've been able to find. And if you were to combine Richard Stallman and John Dvorak...

Leo: Oh, dear.

Steve: ...that's what you would get.

Leo: Oh, my. A combination of the two? Oh, my.

Steve: Oh, a cranky old geek and, yeah. But also a talented programmer.

Leo: Clearly. Clearly a very talented programmer, I would say.

Steve: Yeah.

Leo: Yeah.

Steve: So we've got news. We're going to talk about briefly my experience of downgrading my rights running Firefox, which I'm now doing successfully. If there are any other holdouts who are still using XP, then pay attention. A little bit of news about malvertising. News of Amazon and Google and Flash. More sad news about where Microsoft continues to take Windows. Dave Winer had an interesting blog posting last week, I thought, that I got a kick out of. He's another one, actually he's 60 years old, so he's my age, and we're both a little bit older than you are, Leo. But we've all been around from the dawn of...

Leo: Not much, yeah, yeah.

Steve: ...this industry. A little bit of miscellany. And then a deep dive into, basically, on an audio podcast I can't point to things, but I don't need to. What I want to do, the idea with talking about uBlock Origin is to give people a good background in what it is, where it came from, and what they can expect from it, and what's in there. Because, if anything, it sort of is UI challenged. You know, it's - anyway, we'll go over the whole thing, have a great podcast.

Leo: Yeah, because there's some deep advanced settings I'm very curious about.

Steve: Oh, boy. There's, like, what does this mean?

Leo: Yeah, yeah. Yeah, no kidding. All right. Let's get the security news from Mr. G.

Steve: So just to follow up to our many people who heard that we were talking about audio quality problems at the end of last podcast, I focused on it. And I'm sure the problem was that my system at this end is so bolted down that, although I had originally configured to allow your networks to have direct access to my Skype system, that had not held. And so when we looked, after the podcast, the relay count showed six. Right now it shows zero, and we have 0.0% packet loss. So what this means is essentially I now have a direct UDP data connection between my machine and your Skype machine, with no intermediaries. And it's almost certain that that has been the problem. So in fact it may have been that we didn't lose this for a while so that initially my configuration was holding, and then it drifted. Anyway, I got it...

Leo: Yeah, it's weird. I mean, this is kind of the art of the Internet and of Internet broadcasting. You always have to fiddle with it.

Steve: Well, for what it's worth, I know exactly what's going on. I mean, remember, I wrote IP stacks and ShieldsUP! and TCP protocols and all that.

Leo: No, I know, boy, if anybody does, you know it.

Steve: Yeah. So anyway, I found out what was wrong, and I believe that I fixed it. And, I mean, I've got Wireshark running and confirmed that the IP of your machine is where my packets are going directly.

Leo: Good.

Steve: So I think it's as good as it's going to get, at least with this level. So we'll see how we do.

The picture of the day is a screenshot of what you get - I actually hit this as I was putting the notes together because I went to a domain, adnxs.com. And, bang, I got a big yellow warning triangle - yeah, there it is - saying, oh, "uBlock Origin has prevented the following page from loading." The reason I went to that domain is it was a couple removed from one serving malware on MSN. And so I thought, I wonder if this is being blocked? So the lesson here instantly is no one using uBlock Origin could have been infected by the malware which MSN has been serving. So that's a perfect example. Oh, and by the way, the blocking filter is adnxs.com. So uBlock Origin knows specifically that this particular domain is one that our browsers don't - it's not in their best interest to go fetch.

Leo: And of course I went right there and got the same exact yellow triangle.

Steve: Yeah.

Leo: Good thing.

Steve: Yeah.

Leo: Yeah.

Steve: Okay. So I promised that I would follow up last week on this little utility that was of use in the XP era, but not at or after Windows 7. And that's called DropMyRights. DropMyRights was written by a Microsoft developer to leverage a feature in Windows which allows a process to be started with specific rights, never more than the current user, but conditionally less than. And so all this thing does, the source code is available, and in fact I mentioned last week, if you put DropMyRights into Google, our podcast, discussing it years ago, is the first link that comes up. And Elaine shot me a note when she was doing the transcribing. She does not use Google, and so she shot me a note saying, yeah, not only on Google are you first. So it's sort of fallen by the wayside except at GRC. We are proudly keeping it and XP alive. So...

Leo: Good job.

Steve: So if anybody is using XP, the short version of this is that I am now running Firefox in my login session as a non-admin normal user so that all of those exploits that sort of use the browser as their focus, and either have a font rendering problem in the browser, or maybe have a Flash defect, or some little glitch has been found in the JavaScript handling, whatever, all of these things are constrained to the rights of the process that they get loose in, and so - which is why we tell people it's really safer if you do not run as an admin. Run as a - sometimes it's called a "limited" user. In the XP world it was called a "normal" user as opposed to an admin.

Leo: Standard. Standard user.

Steve: Or standard.

Leo: Yes.

Steve: Yeah. And I'm saying "normal" because the actual header file says "safer level ID fully trusted," then there's "normal," there's "constrained," there's "untrusted," and there's "disallowed." And so you can choose among those. And Firefox won't run as anything other than either fully trusted, which is not as safe as it could be, or normal, which as we know, and as you say, Leo, you're right, in Windows they call it a "standard" user. So I get to be an admin, yet Firefox has reduced rights. So if anything ever happened, then there's much less that it's able to do. And as we're going to see in this episode, one of the coolest things about uBlock Origin is, I mean, way beyond the adblocking aspect is the malvertising, and also explicit malware blocking. So it does an awful lot for us in terms of preventing the abuse that is increasingly prevalent on the web.

So speaking of malvertising, the first note I had to talk about today was that the Malwarebytes guys that have really been focused on the so-called "Angler" exploit kit have detected it elsewhere. And as I just mentioned, they found it on MSN.com. It's the same ad network which uses AdSpirit.de as its launching point. And so the MSN.com site has links to lax1.ib.adnxs.com. And we've talked about, we've talked in the last couple weeks about the way these things work is they essentially chain together domains.

So on the MSN page is a script tag that - and unfortunately it's not, you know, in the old days it was a GIF or a JPG or a PNG, and you just got an ad. You know, you got an image that sat there. But now what everyone does is they embed script pages in their own web pages, and that allows somebody else's JavaScript to run. And the argument, of course, is that, oh, that's the way the ad rotator works. Well, that's nonsense because we had ad rotation back when they were just images. The idea is that the image would be some gibberish-y looking thing that would be a fixed asset on the requesting page, but then the ad server would perform the rotation.

What's actually happening is all kinds of extra-value tracking is going on. And in fact I looked at some of the - by using the logging feature in uBlock Origin, I looked at some of the .js files. And, I mean, it is a little chilling to imagine that, every time we go to pages, all of this stuff is running. So anyway, we'll get to that. But what happens is the first script then invokes a second script, in this case pub.adspirit.de. And then it goes, like, three or four more. From there it actually uses Red Hat's cloud storage as one of its intermediaries through a chain of about five different things, until you are finally delivered using whatever wedge they are trying to exploit in your browser end of things in order to install either ad fraud software or the CryptoLocker ransomware-style software.

So out of curiosity, because we were going to be talking about uBlock Origin, I gulped and put both of those domains in the URL address bar. Now, I wasn't going to get served anything. I thought, you know, maybe I'll see the company, in the worst case. In both cases, uBlock origin already knew about them and just said, no, this is not somewhere you want to go. This is actually a new feature that uBlock Origin added with 9.8 something, where it also filters the first-party domain.

Notice that I actually put those into the URL bar. That wasn't in there initially, and the earlier variations of this that we'll be talking about don't offer that feature. And specifically, for example, its sister, which is just called uBlock - on which development has all but stopped at this point, whereas Raymond is moving ahead - it doesn't do that. It only does third-party references get filtered, but everything in a first-party context doesn't. And Raymond being, as I said, kind of a hybrid between John Dvorak and Richard Stallman, based on everything that I've seen, he just thought, oh, let's filter the first party also, just keep people safe.

So, and for what it's worth, there were three different lists which each of those was blocked by because, when this block comes up, the blocking page also tells you why it was that this got blocked. And if you, for example, SourceForge is being blocked because, as we know, their behavior has been sort of questionable lately. And so one of the first things that happened to me was I followed a link to SourceForge, and up came the big yellow triangle. And I thought, eh, it's okay. So I think I hit "temporary," just because why not just keep things temporary. But so you're able to, right there on the page, say, okay, you're getting a little carried away here. Let's allow this. And then I was able to do my business at SourceForge, whatever it was. So we'll talk about all those features here in a moment.

Today is September 1st, 2015, the day on which two things happen, which is not as good as they could be, but not bad. One is that Amazon has decided to ban Flash ads starting today, September 1st. And it's interesting because it would be great if they were saying no more, we're no longer going to accept Flash ads because they're insecure. Instead, Amazon's posting said that this is driven by recent browser-setting updates from Google Chrome and existing browser settings from Mozilla Firefox and Apple Safari, for example. And what they're referring to there is the click-to-play settings, where it doesn't come up and just run. You have to click on it in order to make it go.

Amazon continues, saying: "...that limits Flash content displayed on web pages. This change ensures customers continue to have a positive, consistent experience across Amazon and its affiliates, and that ads displayed across the site function properly for optimal performance." So essentially what they're saying is we don't want to take ads that we're not going to be able to serve, so let's all stop doing this.

At the same time, and they did mention the changes coming to Chrome, which also land today, Google has had what has been an experimental setting on the content page. If you go to Chrome, in the URL bar of Chrome, chrome://settings/content, that brings up a sort of a dialogue window with a whole bunch of stuff. And if you scroll down a ways, you'll find a section, Plugins. And it used to be that the first one was selected, "Run all plugin content." And in screenshots that are still around the 'Net, it says "(recommended)," which is what Chrome does for the settings that it defaults to. Today, and from now on, it's choosing the second setting, which instead of "Run all plugin content," it's "Detect and run important plugin content." And this has got to be one of those things where they were like, they sat around for a while trying to think, okay, what do we call this?

Leo: They don't want to say "safe," although that's the implication; right?

Steve: Yeah. Actually what they really want to say, I think, is first-party. The idea is that things like games that the site themselves are offering, or...

Leo: Or our video, maybe. Well, actually ours is third party. Right? The Ustream video would be third party on a TWiT site.

Steve: Yeah. I'm not sure...

Leo: You know what, I should see.

Steve: ...how they make their determination. Yeah, you might want to, yeah, go see if it plays your stuff.

Leo: Yeah.

Steve: And then it says, in the original description, it says Chrome...

Leo: Yes, it does play it.

Steve: Ah, whew.

Leo: It's a little confusing because it plays it, and it's showing me setting the content settings from a few seconds ago. Should we choose the third one, "Let me choose when to run plugin content"? Would that be better?

Steve: See, that's - I've got that in my notes as SMG's recommendation because then, again, yes, for safety I would say, I mean, and for example I have Firefox set up the same way. It's called "click to play." And you do that in Firefox by going to your plugins page, and over on the right there's just a whole - all of the plugins listed have dropdown list boxes, as they're technically termed; and it always says, you know, "always run," always run. But you can change it to "click to run," actually, in which case it just comes up and shows you sort of a Lego block. And it says, you know, click here if you wish to run this. I just, because Flash has been such a security problem in the past, and for our audience, and also because we're seeing just content on web pages which is jumping up and down and screaming out and crying for our attention, it's like, eh, you know, I think, yeah, the third setting under Chrome is "Let me choose when to run plugin content."

And there is a Manage Exceptions option, then, after those three radio buttons where, if you're, for example, say that your own Intranet uses some content in ways that, you know, for like a stock ticker or who knows, something, something where you inherently believe it's not malicious, you would want to put, you might say, ask me if I want to run content, rather than running it without asking. But for the following sites, just go ahead and do it. So we get the best of all worlds that way.

RC4, the long-beleaguered stream cipher, one of the earliest ciphers, of course it famously did very poorly due to its implementation on WEP, the original WiFi encryption, where it was, you know, really, it's like run away as quickly as you can. Its implementation SSL for creating privacy, for encryption connections, that was done far better than the implementation for WEP. So it wasn't as bad. But we've been continuing chipping away, we the industry, continually chipping away at it, finding one problem after another. And we're to the point now that it's time just to say it's over. So that begins happening around February of next year.

Our friend Adam Langley at Google put up a posting just this morning. He said: "RC4 is a 28-year-old cipher" - 28 years old - "that has done remarkably well, but it is now the subject of several significant attacks." And he has three reference links after that. He says: "The IETF has decided that RC4 is sufficiently bad to warrant a statement that it must no longer be used. When Chrome makes an HTTPS connection, it has an implicit duty to do what it can to ensure that the connection is secure. At this point, the use of RC4 in an HTTPS connection is falling below that bar, and thus we plan to disable support for RC4 in a future Chrome release." And I'm confusing my release numbers because I was going to say 44. That might be right because I think I was thinking of Firefox, which

I think is at 44.0.3 this morning. I think Chrome's not quite there in terms of just release numbers.

Leo: For some reason I think 40. No, 44.0.2403.157 is the current version of Chrome - 44, you're right.

Steve: Well, okay. In that case we're at 44, which must be what I was thinking. But Adam's aiming at this, I think in, like, what I remember seeing was February. Oh, in fact the next sentence: "That release is likely to reach," writes Adam, "the stable channel around January or February 2016. At that time..."

Leo: Whoa, we're in 45. It just updated.

Steve: "At that time, HTTPS servers" - oh, and notice Adam is not giving a version number because he doesn't know. He's just sort of setting sort of a time in the future, so January/February. And he writes...

Leo: Do you feel like they're trying to keep in step with Firefox?

Steve: Yeah.

Leo: Like they both want to be kind of the same?

Steve: Well, there is, now there is sort of like version number envy, where it's like, you know...

Leo: Yeah.

Steve: It's like the uninformed user might think, wait, wait, wait, wait, wait, wait. And remember that Firefox used to be on version 4 when Chrome was at 28.

Leo: Right.

Steve: And we knew, we knew it didn't matter.

Leo: No.

Steve: But now, now it's going to be like, oh, wait a minute, Chrome's at version 45. Firefox is only at version 44. Does that mean Chrome is better? It's like, oh, no, here's a whole new problem. Anyway, Adam writes: "At that time, HTTPS servers that only support RC4 will stop working." Now, okay. That sounds scary, but probably not. He says: "Measurements show that only 0.13% of HTTPS connections made by Chrome users who have opted into statistics collection currently use RC4. Even then, affected server operators can very likely simply tweak their configuration to enable a better cipher suite in order to ensure continued operation."

And then he notes that "Chrome has long implemented 1/n-1 record splitting" - that's one of the mitigations for one of the RC4 problems - "and is thus protected against the BEAST attack" - which was the first of these problems - "even with CBC modes and TLS 1.0." So the point was nobody was at risk in Chrome. Same thing, same is true for Mozilla. They fixed it quickly. Anyway, Mozilla made an announcement, and Microsoft made an announcement - this was announcement morning - for the end of RC4. And in fact one of the things that Chrome does - oh, in fact he mentions it here. He says: "Current versions of Chrome don't advertise support" - that is, even right now - "for RC4 on an HTTPS connection unless the first connection attempt fails, so servers that already support a non-RC4 cipher suite will not see any change."

In other words, what Chrome is doing is it's pretending. Remember that the way this protocol negotiation works is that it's client, that is to say browser initiated. The browser sends a list of all the [dropout] of protocols and handshakes and versions and essentially a list of cipher suites among - and one component of each cipher suite is the encryption algorithm. RC4 would be an example of one. AES, of course, is the one that is now in favor. And so it sends a list of all the ones that it supports. Then the server selects from among them based on the priority it's been configured with.

Well, unfortunately there are some servers that have RC4 in first place because there have been problems with cipher block chaining that caused people to move those up to move RC4 to the head of the line. And so the point is that, if the browser doesn't appear to offer RC4, then the server can't choose that and will automatically choose something else. If a server only had RC4, and I can't think of what, I mean, you know, maybe some corkscrew or something that, you know, an early IOT device that somebody has and...

Leo: A thermostat.

Steve: Yeah, exactly, I mean, some really, you know, some windup shoe leather or something, I mean, what isn't going to support something else? But the problem was that RC4 really was, it was so lean and elegant for what it did. And but its time is obviously passing at this point. So anyway, the browsers are saying goodbye to it. And starting after a couple months of next year - oh, and by the way, Microsoft seems to be following Chrome's timing, as is Mozilla. They're all saying, okay, we're in agreement, the industry has sort of said, yep, beginning of 2016. I saw Mozilla's numbers also. I think they said, out of something like 214,000 domains, 834 - this is all just out of memory, so I'm sure I'm misquoting the numbers, but you get a sense for it. More than 200,000 domains, a little more than 800 were still using RC4. And one has to imagine, I mean, if that's someone you even want to connect to. It's like, is there anything of real value there? You know, why? What kind of a server would still be doing that if the percentage is 0.13? Unlikely.

And since we've been talking about Chrome, I just thought I would note something that I stumbled upon today. Google changes their logo. Biggest update in 16 years. And it's animated. So, oh, I love the page. Whoops, yeah.

Leo: This is the Google.com page. This is just a Google Doodle to demonstrate the new logo. I mean, it's not going to be animated forever, I don't think. It's pretty, though. I like it.

Steve: Actually, what's there, if you click the link in my show notes, FastCoDesign link - oh, no, there it goes. Now it's doing its animation. I think that phase, but without the hand.

Leo: You think that's going to be the little dot dot dot thing?

Steve: Well, they're saying it's animated. And if you click that FastCoDesign link in my show notes, that talks about the evolution in fonts and says that it's an animated logo. So who knows? We'll see.

Leo: Well, why not, you know? I hope they're using HTML5, not Flash.

Steve: Oh, I guarantee it. And you remember that crazy page that I created showing how magnetic storage recording operates?

Leo: Yeah.

Steve: I think it's GRC.com/animation.htm. And that's just all JavaScript, writing on the canvas. And, you know, it's a...

Leo: You don't need to...

Steve: No, it's only inertia. At this point it's just inertia.

Leo: Yeah. Well, there is one - we do streaming video, of course. And there's one argument for that. This was all done, you're saying this was all done with HTML5?

Steve: Yeah, yeah. There's no active content at all.

Leo: Yeah. It could be CSS or JavaScript. It doesn't actually even have to have JavaScript. But it certainly doesn't have to have Flash.

Steve: Well, for example...

Leo: Streaming video is a little more tricky because there isn't yet, I don't think, an approved non-Flash codec. So you could put a video tag in, but it's not clear what will happen.

Steve: Yeah, I don't need it on my site. I serve video. I use three, and you have to put them in the proper order, you know, and it's like...

Leo: What codec are you using?

Steve: I'm using, well, I'm using the built-in browser codecs that run on cross-platform. And then my final fallback, if I think it's WebM and Ogg and something else, if none of those three work, and they have to be in the proper order or iOS doesn't like it, then it finally falls back to using a Flash player of my own design. So we were able to do it; but it's like, it's a pain. So you're right. We're still there.

Leo: Well, we're - and people say, well, Leo, if you talk - you don't like Flash, but why are you still using Flash? Because we use Ustream and BitGravity. We don't do our own video. So we're, you know. But both BitGravity and - I know Ustream has an HTML stream. We pay for Flosoft, which is an HLS stream, because we have to for people who come in on iOS. We can't say goodbye. So, but, you know, that's expensive. So we're trying to use the free providers as long as we can.

Steve: Okay. So anyone who was smugly resisting the move to Windows 10 in the belief that 7 and 8 were going to keep them happy has been disabused of that this week. PCWorld writes: "Microsoft slips user-tracking tools into Windows 7 and 8 amidst Windows 10 privacy storm." Ars Technica, their headline: "Microsoft accused of adding spy features to Windows 7 and 8." ExtremeTech: "Microsoft backports privacy-invading Windows 10 features to Windows 7 and 8."

And ExtremeTech was great. They said: "Every time Microsoft releases a new version of an operating system, there's always a few users bitterly unhappy at the company's decision not to support new features on older products. Microsoft has finally listened to these diehard devotees of older operating systems. If you felt like Windows 7 and Windows 8 offered you a little too much privacy, rejoice: Microsoft is updating those operating systems with the same telemetry-gathering software it deployed on Windows 10. What? You wanted DirectX 12?"

So anyway, this news was broken by the guys at Ghacks, who discovered that since about April there are four different updates which Windows 7 and 8 systems have received, which installs the same controversial, unblockable, you can't turn them off, they avoid the hosts file, they're extremely resistant to anyone tampering, into Windows 7 and 8. So they connect, basically they...

Leo: Is there an opt-out? Is there a warning?

Steve: No. This has all been done just with no one noticing it. Microsoft doesn't allow you to turn it off. And in fact I liked Ars Technica's summary of this because I think it was very level-headed. Ars writes: "As with the other privacy concerns around Windows, our feeling is that the major issue at stake here is not that Windows is collecting data, but that it put the user in control. Collecting information about application errors and the way the operating system is used is reasonable. Having an accurate picture of how people use the operating system is likely to produce a better platform in the future; knowing which applications crash, and why, is obviously invaluable if those apps are to be fixed.

"But we continue to believe," writes Ars, "that people who do not wish to be a part of such data collection should have a clear and unambiguous way of opting out, and these opt-outs should be rigorous. Disabling CEIP" - that's that customer experience program "for example, should not only prevent systems from sending CEIP data, but should also prevent systems from retrieving configuration data from Microsoft's own systems. We would also argue that these settings should be made simpler. At the moment there are many individual controls, each governing a particular behavior. Some kind of global control to supplement these fine-tuning switches would be an improvement. We like cloud connectivity and online features, but these should be paired with clear user control."

And so what happened is that, secretly, and I only use that word because it wasn't explained, they came across as security updates, which everyone's like, oh, my god, yeah, I need those. And so, and they were the recommended updates. And they're listed. So I won't go into all the details. But I've got links...

Leo: Recommended or critical? Because that is a distinction that's important.

Steve: Yes, recommended.

Leo: Not critical, okay.

Steve: Correct. And so, as I was going to say, I've got links in the show notes. And on the Ghacks page they talk people who are annoyed by this through the process of, you know, they enumerate which four are the troublesome ones and that you can remove them. And then you need to tell Windows to hide them so that it will not automatically reinstall them. And what's also been controversial is that Microsoft has apparently gone to some lengths, now, maybe it's just to bypass malware. But, for example, vortex-win.data.microsoft.com and settings-win.data.microsoft.com are the two domains that these new services, these telemetry services, connect to. And they explicitly bypass the DNN resolution system, making it much more difficult to block them. You could do so outside of Windows. Some people said that even the local firewall wouldn't. Turns out those were some specious early reports that have not withstood scrutiny.

But anyway, I wanted to let our listeners know that Microsoft has moved some of this data gathering back into 7 and 8. But at this point it's certainly, you know, it's not like we're without control. But it was done without users being told, and there is no other mechanism for disabling it. Apparently there's some group policy edits that you can do. The group policy system is like a deeper, lower level settings system than the UI often surfaces. But it'd be nice if there was like a privacy panel, the way at least we do have in Windows 10, even though we know turning everything off in Windows 10 doesn't prevent it from continuing to do some of this stuff. So anyway, 7 and 8 have joined Windows 10.

Leo: I want to really emphasize to people that this is a recommended update, not a critical update. You should continue to install all critical updates on Windows.

Steve: Correct. Correct. So Dave Winer. We know Dave Winer, Leo.

Leo: I know him well. He was the creator of podcasting.

Steve: Yes. He's about our age, software developer. He was the early proponent of outlining. And I still think outlining, I mean, I do an outline, our listeners who have seen the show notes see an outline format.

Leo: Yeah.

Steve: It's because I'm using a desktop outliner to pull all this together during the week. And then during at least the morning, and sometimes I start in the evening before, I organize it and figure out what order it should go in. And anyway, I just think outlining is fabulous use of the flexibility that we have, thanks to computer software.

Leo: He wrote ThinkTank and then later MORE.

Steve: Oh, my god, yes.

Leo: And he has, by the way, I don't know if you've ever seen it, but he has online an HTML5 outliner called Little Outliner, LittleOutliner.com.

Steve: And is his site, is it Small Things? No, no, that's...

Leo: No, no.

Steve: I'm thinking of Joanna.

Leo: Oh, gosh. He has a funny - Scripting News is his site.

Steve: Scripting News, right.

Leo: Scripting News, yeah.

Steve: So former contributing editor at Wired magazine, research fellow at Harvard Law School, received his master's in computer science from U. of Wisconsin, bachelor's in math at Tulane. So, you know, one of the founders. Been here forever. And the title of his blog, as you said, I mean, he was also - he's considered the father of blogging.

Leo: Right.

Steve: The New York Times called him the "protoblogger." Anyway, last week's blog title was "Mac OS is spyware, too." Which I got a kick out of, coming from Dave.

Leo: Yeah, well, he would know, yeah.

Steve: He said: "All of a sudden my Mac is telling me whose birthday is tomorrow. People I don't even know that well. How did that happen? I don't like my computer randomly and unpredictably getting all social on me. It's a tool. Try to imagine a carpenter's hammer starting to nag about an upcoming bar mitzvah, a baseball player's bat starting to warn you about overdue bills. Who asked for this shit?" And I'm just quoting Dave. So thank you, Dave, industry veteran, comrade-in-arms. That is precisely how I feel. That's my issue is that, you know, I don't want a social networking hub. I want an operating system. And I saw one of your tweets a week or two ago where you were tweeting to somebody, and you mentioned @SGgrc. And you said, "What Steve wants is DOS."

Leo: Yeah.

Steve: I thought, well, you know, a file system...

Leo: Yeah. You said it. You want a file system and apps.

Steve: And runs my programs.

Leo: Runs software.

Steve: And otherwise...

Leo: Just shut up. That's DOS.

Steve: Yeah, stay away. That's right.

Leo: You shouldn't even install a GUI for that. That's just overhead you don't need.

Steve: Well, see, now, a GUI was useful progress. That, I mean, no one doubts that that was an innovation. That allowed us to create explorable apps where we could browse around in the menu. I mean, that's been a win. Everything since, downhill. Okay. So anyway, I got a kick out of Dave's...

Leo: We're all old codgers here. It's okay.

Steve: That's right.

Leo: I like it how it used to be.

Steve: So I just got - I saw something I wanted to mention via Twitter from Jeff Price. He said: "@SGgrc Surprised you liked auto update on Google's router. Remote access to router is a path to exploit." And I wrote back, I said: "That's a great point, Jeff. But there are many ways for it to be done right. It's the 'done wrong' cases that we are always talking about. The router doubtless reaches out to Google periodically to check for anything new and uses something strong such as certificate pinning and strongly signed firmware, et cetera. My point is it's easier to do it wrong, but quite possible to do it right." And that lengthy reply is courtesy of long DMs, which are now possible in Twitter.

And there's some guy, @nvtweak is his Twitter handle, @nvtweak. Neat guy. He's been hanging out on my feed. But he doesn't follow me. I am unable to DM him. Anyone can DM me. I changed that setting, and I have no problem with it. It allows me to have dialogues. But there have been many times when I've wanted to say something to this guy, but I cannot pollute my stream with a comment to one person when I've got 50,000 people following me. I just, I'm unwilling to do it, not when DM is there for that purpose. So Mr. Nvtweak, if you are hearing this, I'd love for you to follow me. You're in my feed all the time anyway. And then I can respond to some of the neat stuff that you post. I'd like to be able to.

And we were talking about the "Mr. Robot" finale last week, which wasn't because...

Leo: It was put off, wasn't it, yeah.

Steve: Yes. That Wednesday morning, the day after last week's podcast, was that awful shooting on camera of the two reporters. And apparently whatever we're going to see tomorrow night is so eerily similar on the "Mr. Robot" finale that USA Network decided to bump it a week. They said, you know, just we don't want to air what we were going to air because it would seem unfeeling of us. So anyway, that happens tomorrow, for anyone who doesn't already know that they got a rerun of the previous week's "Mr. Robot" last week.

And I did want to share a nice note that I found in the mailbag from a Mike Blommer, who's in Ann Arbor, Michigan. I talked last week, I sort of took a little time out from testimonials from users or owners of SpinRite who have used it to fix their system, to talk about what SpinRite does, and the way it does it. I forgot to mention that SpinRite doesn't care what's on the drive because it deals at the physical sector level. As I said last week, you know, it's looking at sector by sector. As a consequence, it's able to fix a hard drive, even if it's not an NTFS or a FAT format or a recognized format.

And we've talked about, for example, how it often is able to repair - there were a couple testimonials where it was helping DVR users. Well, this is another TiVo drive. Mike said: "Hi, Steve. Just wanted to let you know that SpinRite brought my TiVo back to life. My family has been addicted to TiVo for almost 10 years" - and I know that, Leo, you and I are among that crowd.

Leo: Oh, yeah. I came back to TiVo after missing it because I went to cable company DVRs. They're just not as good.

Steve: Oh, wow, yeah. Those are really poor imitations. I tried to go to whatever that was, Windows thing, the Media Center deal.

Leo: Yeah, now discontinued. Yeah, Windows Media Center, yeah.

Steve: Yeah. And it was, I mean, I fought with it, and struggled, and it's like, oh, my [crosstalk].

Leo: TiVo's the best. Still is.

Steve: Yup. So he says: "...addicted for 10 years, and I recently turned on my TV to see TiVo stuck on the "Welcome, Powering Up..." message. In searching the various TiVo forums, the most common response is that the hard drive is likely damaged and needs replacing." And actually, as someone whose own software product has saved his own TiVos on many occasions, I can vouch for that. That's the problem.

So he said: "I had purchased SpinRite, just in case, last year after becoming a fan of Security Now! and other TWiT podcasts. And I remembered you mentioning that SpinRite will work on TiVo, and Linux disks, too. So I figured I would give it a try. I hooked-up my TiVo drive to my PC running Windows 8, rebooted into my USB drive configured to boot SpinRite, and let it run overnight. There weren't any errors displayed, only processed blocks. But I think I remember," he writes, "you mentioning that SpinRite might still fix the drive, even if it doesn't display any errors. So I crossed my fingers that SpinRite fixed the boot part of the drive, and hooked it back up to the TiVo. And voila. The TiVo booted up just fine with all our recordings.

"SpinRite saved me," he writes, "from spending around $200 on a new TiVo-formatted drive, as well as losing many valuable recordings. Thanks from a relative newbie listener of Security Now! - two years. Love the show. I'm starting to get my 14-year-old son stuck on it, as well as The Tech Guy. We don't understand everything" - and he says, parens, "(insert picture of deer in the headlights here)," he says. "But we enjoy learning." So, gee. Thank you very much for the great note, Mike. And to your 14-year-old son.

Leo: The Tech Guy is the training wheels for Security Now!. You know, you start with The Tech Guy, and you work your way up to Security Now!.

Steve: Right.

Leo: Actually, Security Now! is the pinnacle. You might want to try Windows Weekly or MacBreak Weekly first. Then Security Now!. Work your way up.

Steve: I would argue you're doing ads right, Leo. I mean, I've been thinking a lot about this question of is there an implied contract or an implied agreement of any kind with a site we visit. And I have to say I would say no.

Leo: Really. Interesting, yeah.

Steve: I think, yeah, I really think not. I think, I mean, first of all, I get it from the standpoint of wanting to support sites that we like, like iMore. And I'm delighted that, when I go there, I'm seeing that Google has replaced their ads with filler because I'm paying something to iMore to have Google do that through Google Contributor. But I think about - I guess what I would call it is "opportunistic advertising." You know, when we ride public transportation, like a bus, you see ads above the windows because there's some empty space that allows, what, the city to generate revenue from advertisers by taking advantage of that space. Roadside billboards, same thing. There's some empty space, and, hey, put up a big ad, and some people will see it, and it'll happen. It's sort of opportunistic.

And of course the same has always been true of broadcast television, where the ads are interspersed with the content. People grumble about what seems to be the growing percentage of ads. And I notice, as a cable viewer, that some shows seem to have a preponderance of advertising and lower content, probably in order to pay the bills so that shows that are stronger aren't carrying that same burden. So, and then we have Twitter, which is famously still trying to figure out how to make money. You know, how do they monetize? And so then we come to web pages. And we sort of slid into this over time where, I mean, speaking of sliding into things, you often describe me as the discoverer of the first malware. In fact, that was adware from a company called Aureate.com.

Leo: Oh, yes. I remember them well.

Steve: Aureate. And then they changed their name to Radiate. And the idea was that they were pushing the idea of advertising-enabled freeware. So freeware that you would download would bring this Aureate adware with it and install it in your computer. And the concept was that, as you downloaded more Aureate-enabled software, you wouldn't get redundant copies of this. So only one sort of Aureate hub would live in your system. And then the various software that you used would use it.

The problem was people were being spied on. Because you were being shown ads, the system was, I mean, behind all of this there were always clever techies who were like, oh, oh, oh, we can do this, and we can do that. And so, similarly, it was we'll put ads in [dropout]. The more popular the software is, the more copies will be downloaded. But the more often it is used, the more time is spent in front of users. It even detected when it was being covered up. And so the idea was that the actual visibility of the ads, using technology, could be fed all the way back up the channel so that the freeware authors themselves would be reimbursed based on the popularity of the use of their software, which this application adware was making possible.

The only problem was to do that meant spying on users. And in the EULA that came with it, the freeware was supposed to mention this and disclose it. But of course nobody read that. So what happened when I discovered this entire ecosystem that was unknown at the time that I found it - oh, and I found it, by the way, thanks to ZoneAlarm, an application-based firewall running in my computer. I was an early beta tester of ZoneAlarm. And as I was using it, up popped an alert saying that something called aureate.dll wanted to communicate out of my computer. And I said, you know, as somebody who back then was - this was probably Windows 95. And so I knew everything that was in my computer and what it was doing. So here was something communicating that I didn't know what it was.

I dug in, found out what it was, unearthed the whole system, and then wrote OptOut, which allowed you to do this. Because the other insidious thing about this, because the concept was that it would be shared among apps, was Aureate specifically told the freeware authors, one of whom brought it into your system, and presumably more of whom would share it, not to remove it, even if their application was removed, because other shareware might still be using it. So this thing came in, and then it never left, and continued to run. It was also badly written. It was crashing IE and causing all kinds of problems. And oftentimes, if someone downloads something that then their system starts misbehaving, you remove it. Unfortunately, the problem didn't get removed, and the problem persisted.

Anyway, this was just a total disaster that many people who've been in the industry for a long time remember well. So now what we have, we have something eerily similar which has been evolving over time. The technology of the browser where the site we go to provides the HTML content, and that content can contain references to other content, either more content from the same site, for example, the icons and menu piece, you know, the various features of the site's own design, the browser goes and fetches. But it can also contain third-party content where the browser goes and gets something from somewhere else, not from that same site.

And so, once again, clever people said, hey, you know, we can use this in a way that nobody really thought of. We can tell a site to embed, like to reserve some physical real estate on the page and point to our ad server so that, when that page displays, the ad server provides the content. We can, thanks to the so-called "referer" header, we can get the domain that has asked for that ad to be displayed, which means that we can pay the domain to request advertising content to appear on its pages, and in an exactly analogous fashion. The more popular the page, the more popular the domain, the more visitors, the more ads we get to serve, the more revenue they get to generate. Which, you know, that's a cool idea.

And so we've sort of gone along like that for some length of time. Then scripting became popular. And rather than a link to an image, now it's put scripting tags on your page, and we will run some script because that will give us more flexibility. As I was talking about earlier, you know, ad serving. There's also this notion of, you know, the whole tracking thing came from many different sites around the Internet all using the same advertising server. Because when the browser asks for content from a third party, if the third party has ever given it a cookie, it'll return the cookie. And it'll return that cookie based on that third-party ad server.

So as you roam around the Internet, going to sites that are all sharing the same advertising network or advertising server, your browser keeps going back to that same server. So what has developed is the so-called "tracking" concept, the idea, which some people find creepy, that that third party that is not a site you have a relationship with directly, they're just serving ads. The idea is that they could be, and in fact we know that they do, they compile a dossier, essentially, anonymous most of the time, unless there's some way to get information. We know that they want to know everything they can about you. They want your zip code because that tells them a lot about your socioeconomic status. Everything that they can find out, they want to.

And in fact they used to be hosts of those contests where, remember, we used to have, like, web-based contests, where fill out this, and there's some chance that you will win a prize.

Leo: Some tiny, tiny, tiny chance.

Steve: Exactly. What wasn't immediately clear was that, when you were filling that out, you were giving them your name and address and phone number and whatever other information they could get from you from the ad server, which meant that your browser gave it back the same cookie that it could now associate with everywhere this browser had been over time. And of course, based on the profile of the websites, what kind of website this is, that information would happen. And then it got a little creepier still because search engines were putting the search queries in the URL. And in fact many still do. You will see, you know, search= and then the search terms you're using, separated with plus signs. That goes to the third party in the "referer" header. So now these anonymous entities that you have no relationship with can see what you're searching for and add that to this growing profile.

So this all sort of creeped people out. As the system has matured, this has all developed. So, and we've been talking for the last few weeks about the ethics and morals and the downside. The bad news is browsers have become one of the main ways for bad stuff, malware, to get into our system. And advertising networks are large, and they've become far more sophisticated, if I can use that word, than in the original, like, Google AdSense days, where it was just a few little things that would appear on your website and was very innocuous.

The other problem is I've referred to this as "perverse incentives." And by that I mean that there's a problem. And that is that the more obnoxious an ad is, the more it's going to attract attention. It may end up being clicked more. And so what that does is it incentivizes the designers of the ads to deliberately go out of their way to make them a bigger problem, you know, to attract the attention of the website visitors.

So this is a problem because we didn't go to this page for ads. We went to the page based on a link that we followed or the reputation of the site. And this is one of the problems is sites' reputations are being damaged because, as we've been covering in the last couple weeks, the fact of the way this system has evolved has meant that sites don't have control over the ads that they are causing to have delivered on their page, under their name. They just provide script or a reference tag, and hope for the best. And so not only does malware get delivered, but obnoxious ads get delivered. And the problem is the more obnoxious they are, the more revenue they may generate.

And then we have the other perverse incentive, which is the more ads that are on the page, the more revenue the site generates because it is, again, it's all metered based on the actual exposure of the ad to users. There are sites which are annoying, where they take a long story, and they chop it up needlessly into 10 small pieces, making you step through 10 pages in order, clearly, the only reason is to create more ad hits for them. So what's happened over time is that what started out to make sense has, because of these incentives which throughout the entire system sort of guarantee that it's going to go off the rails, we are now seeing it going off the rails.

And the good news is users are not without recourse because all of this uses technology. All of this is technology that we have control over, you know, in the same way one of the earliest concepts that we put forth on this podcast was the futility of encrypting a DVD. Because if a DVD was going to be viewed in someone's living room, then also in the living room was the decryption technology, necessarily. And so there wasn't any way to protect it. Similarly, if a website is delivering us a web page of content and references to other stuff, ultimately we are in control, in exactly the same kind of model.

So uBlock Origin is, as I described it last week in the introduction to this and promising to talk about it in detail, it is essentially an HTML firewall. Software firewalls that we've traditionally used have blocked based on port numbers and IP addresses. And, for example, they've - some of them are dynamic, where if packets leave off to a certain destination, then they're permitted to come back from the same destination. But any packets unsolicited, coming from somewhere else, get ignored. So you have static rules and dynamic rules. And we have the same thing with a sophisticated - that we have with a firewall with a sophisticated content blocker like uBlock Origin.

So my feeling, just to finish the issue, before we get into the technology, is that I like the idea of supporting people, supporting websites where it makes sense to. But I do think of this, I don't buy the idea that there's an implied responsibility for me to look up at the advertising above the windows on the bus or to slow my car down and read every billboard that I pass, or never use the fast-forward button on my TiVo. I mean, and Twitter has a problem because they still haven't figured out how to make any money. So it's not our fault for these things having sort of a tenuous model of survival. And you've mentioned, Leo, several times you've raised the point that, if we don't support free content, we're not going to have free content. And I agree with you. I mean, I absolutely would never dispute the fact that this model has some problems.

So I'm not sure how it's going to shake out. We are going through a transition. And users now know that they are being tracked. The benefit, we're told, is that the more the advertiser knows about us by an anonymous cookie, the more relevant ads can be served. Okay. If you ask most people - oh, no, I won't say "most." I can't make that judgment. But people ought to be given a option whether they would rather have irrelevant ads or being tracked. Now, the problem is, with ad relevancy it's believed that that increases the value of the ad, so the site which is serving more relevant ads is serving ads that are more valuable to the user. Who knows.

But I think there's no question that, as an industry, we're going through a transition. At this point people don't like the idea of being tracked. They don't like the idea of their battery being drained by their mobile device. And remember, the other sort of gotcha here, or got us, is that the site we're visiting provides us with this HTML content, which the browser then parses, and it launches off on the project of filling up this page. Using uBlock Origin yesterday, I went to Wired.com. With uBlock disabled for Wired.com, it took 45.83 seconds and 125 requests to load the page, for the page loading to stop, 45.83, almost 46 seconds. I enabled uBlock Origin. I got visually the same page, in 4.84 seconds, one tenth the time; 98 requests down from 125; but only 10 domains connected, rather than 33. So one third of the domains and 10% of the page loading time.

So what may happen, if blocking becomes prevalent, is that we may see sites that start detecting when an adblocker is present. And there are sites that do that now. In fact, PlayOnLinux.com, just probably PlayOnLinux.com. I ran across that link when I was digging into this stuff. And PlayOnLinux.com has a little statement over on the left that says, "Oh, you're using an ad blocker. This site depends upon revenue from ads for its existence. Please take that under consideration." And so we may see that kind of soft statement where an ad blocker is detected. And that's certainly possible to do.

There are anti-anti-adblocker, that is, adblocker detector defeating. But I think for my purposes that's going too far. If I went to a site that said "We are ad sponsored; you're running an ad blocker; in order to proceed you must disable it," now I have a decision to make. Is going there worth dropping my shields, essentially, and accepting all that content? I don't have a problem with that kind of decision, and uBlock Origin, because it supports sticky settings, allows us to support that kind of model.

So it may be that sites initially say, hey, you've got an adblocker, please consider dropping it because, if you want us to be able to continue offering this content, ads are the way we pay our bills and we pay our staff. Then people can decide how they feel about that. Or sites could get stronger about it and say you can't come in unless you allow your browser to pull ads from us. And then the third way is, and this would take some doing, but the first-party sites could also be proxies for ad networks, where the ads get pulled through them, and then they display as first-party content.

Leo: Yeah. A lot of - some sites do that. And they won't be affected by this; right?

Steve: Correct.

Leo: If we took the ad banners that we currently use Google to serve and just put them in as graphics, those wouldn't be blocked. They're just images.

Steve: Correct.

Leo: Yeah.

Steve: Correct. Correct. Okay. So what is uBlock Origin? uBlock Origin is first and foremost an easy-to-use, drop-in switchboard. An HTML firewall, I think, is the best way to put it. And in fact Raymond describes it that way himself. I mentioned that the author is a guy named Raymond Hill. He's a programmer living in Quebec, Canada. He goes by the handle Gorhill, G-O-R-H-I-L-L, and uses GitHub to host his projects. He first wrote something called HTTP Switchboard, and that was sort of where he began to develop the concepts of an HTML firewall. That then later evolved into something called uMatrix. And his UI sort of has a matrix-like effect, as users will see and as we'll describe.

Now, uMatrix became popular, and then it - but he wanted to make something that was more automatic. At the same time we've started to have curated lists of domains that are serving stuff that some users don't want. A huge category is ads. Another is malware. There are domains that are identified as being sources of malware. You never want your browser to go fetch anything from those domains. So that's really not the responsibility of the browsers, except in the extreme case. We've seen Google, a search engine, beginning to take responsibility for warning us of URLs that are associated with malware. And we know that it sometimes false-positives because for a while the TrueCrypt archives that I've been sourcing on GRC were flagged that way. So it's not perfect. But there are these lists which others are maintaining, and just on a volunteer basis, curating lists of these domains.

So what happened was uMatrix became uBlock. And that project has existed for it looks like about at least a year. What happened was, and part of the reason I described Raymond Hill sort of as what you'd get if you had mixed together John C. Dvorak and Richard Stallman, is...

Leo: That could be good or bad. I'm not sure.

Steve: Yeah. I mean, but it feels very much that he's that way. There were other contributors to uBlock, and there was a falling out among them. And so he decided that he did not want to be associated with those people any longer, and he forked the project to create uBlock Origin. This was back in May. So June, July, August, about three months ago. They split amicably. uBlock non-Origin, uBlock, and the icon for that is a "U" inside of a red or a dark red stop sign, as opposed to uBlock Origin, which is a "U" and an "O" that are sort of connected to each other. And that's in a little shield instead. Exactly, and you're showing it there onscreen, Leo.

So what's happened is uBlock.org the site was created as a home for the original uBlock extension, which - and all of this is multiplatform, multibrowser. Firefox, Chrome, Opera, Safari are all supported. And Microsoft is clearly feeling that they're missing out on this, so as we discussed last week, they'll be adopting what looks like it's going to become the web extensions API. And then soon there will just be an extensions ecosystem using an almost common code base that everyone will be able to use.

So about 90 days ago Raymond created uBlock Origin. And if you look at the charts of development, the development of uBlock has pretty much stopped. It looks like they're fixing problems, but the wind has just gone out of it. So for what it's worth, uBlock Origin, it's the one that was created by the originator. He, for whatever reason, became disaffected from the project. They're taking donations over on uBlock.org that do not help Raymond Hill at all. And what's interesting is he actively refuses donations. He doesn't want money. He doesn't want support. He doesn't want to pollute his coding environment with any sense of obligation toward his users. He writes at one point that, "As long as I feel like coding, that's what I want to do. And at any point that I don't feel like doing this anymore, I want to be able to walk away from it and not feel like I owe anybody anything."

So that's just who he is. He doesn't want money. He doesn't want support. I would love to give him some, but he won't take it. And so he's doing it because he wants to, and I just think we're lucky to have it. So he describes it himself, uBlock Origin, the one that will be supported by the author - and in fact, in the last 90 days he's been adding additional new features. He just added cloud support yesterday. It went to version 1.1, which allows you essentially to sort of do a copy and paste, using either Mozilla's or Chrome's existing sync system. So he didn't have to recreate his own servers. If you have sync, you're inter-browser sync-enabled in Firefox or Chrome, then using uBlock 1.1 you can copy your tuned configuration into the cloud, where you've done it, and then sort of paste it from the cloud back into a configuration on a different system. It's not automatic. It's not dynamic sync. And this is the first version of it that just appeared yesterday. So he's playing with that.

So my point is uBlock Origin is the ultimate product of a series of evolutions of this concept. In a review that I know you saw, Leo, last week, 10 different filtering systems were profiled, and uBlock Origin was faster than all of them, smaller than all of them, blocked more than all of the rest of them, and essentially is just the one we want. And I know, Leo, that I've heard you mention that this feeling of speedup is something you can really feel.

Leo: Oh, it's palpable, yeah.

Steve: Yes.

Leo: Absolutely, yeah.

Steve: Okay. So Ray describes it as a point-and-click firewall which can be configured on a per-site basis. This is one of the reasons I think it is very important. On the screen there that Leo is showing is a big blue power button. That is per-site sticky by default. So it is as easy, if you went to a site where you didn't want uBlock to be blocking stuff, you just - you click the little red shield. It opens the pop-up UI. And you click on that power button in order to disable blocking for this first-party domain, for this domain. If you control-click, then it blocks or unblocks it just for that one site.

Now, right now what Leo is showing, there are tooltips popping up. That tells us that it is not in advanced mode. If you click on the brown bar at the top, that will take you to uBlock's Settings page. And at the bottom, right there at the bottom of the first set of checkboxes, is the Advanced mode. So if you turn on Advanced mode, then go back to the uBlock shield, you will then discover that there are plus signs to the left of the words in the gray bar. You click the plus sign, and that opens up the Advanced Settings window, where you're able then to look at the domains that have been successfully read from, that have been permitted, and domains where some was permitted, or all has been blocked. You're also able to block, not only based on domain name, but on object.

So my goal here is to sort of give our listeners a background and a sense for what this can do, and then, without trying to make this an owner's manual, where I explain to them in detail how every aspect of it works, because you're just going to want to open it up, install it, and play with it. But I wanted you to know where you can get the extra goodies. And oddly, when you have it in expert mode, or Advanced mode, then he turns off the pop-ups. And it's odd because I thought only Chrome had pop-ups and Firefox didn't, but it's because I had enhanced mode enabled on Firefox, but not on Chrome. He believes that pop-ups annoy advanced users, so he disables pop-ups in...

Leo: Tooltips, we should maybe...

Steve: I'm sorry, tooltips, tooltips, yes.

Leo: Yeah, that's clearer what that is, yeah.

Steve: Okay. So the idea is that this thing is cross-browser. It in the background obtains lists, autonomously, without you doing anything, obtains these curated lists that third parties have put together. For example, one of them is the Adblock Plus list. And there's like an ad server list. There's a malware list. And you can, again, under Settings, you're able to look through the lists and check and uncheck the lists that uBlock amalgamated into its master blocking. So if you want it to block fewer things, sort of overall generically, you uncheck them. If you want more, you turn the checks on. When you're finished, you say Update up at the top. And essentially it recompiles these into a form that it's able to use at very high speed in order to do its work. And then you are using this Advanced mode. You can see what it has blocked and not blocked.

Over in the per-site settings, there are plus and minuses. That's log base-10 the number of things blocked or permitted. So one dash means between one and nine things were blocked, if it's a minus as blocked. Two means between 10 and 99. And three dashes means it blocked between a hundred or more. And then the reverse is true for plus signs, meaning that it permitted. Some objects will have both pluses and minuses, meaning that it selectively blocked and permitted things in some domains. And you are able to click on those. You can click in the green to say I want to permit these things, and they will be sticky; or you can click in the red to say I want to block them; or you can click in the middle to turn them off. And in the third column, where Leo is demonstrating in the video, those are per-site. In the second column, they are global.

So, for example, if you saw a domain that you absolutely never wanted to receive from, you could, in the second column that doesn't have the pluses and minuses, you would click in the red, and it would block either that object or that domain. Raymond suggests blocking all third-party scripts and all third-party frames. I tried that for a while, but it seemed overly restrictive. Things weren't working. It was like having NoScript with it blocking all scripts. Unfortunately, too many sites are dependent upon scripting.

Leo: You see what it blocks on my site. Everything.

Steve: Oh, yeah.

Leo: Including our nice typefaces, our New Relic monitoring for server errors. I mean, it really disables the site.

Steve: Yeah.

Leo: All the analytics is gone. And the ads. It does disable ads.

Steve: So what I wanted, essentially what I wanted to get through to people is this is what you want. You don't need to dig in. You can simply drop it in, and it's configured like for optimal usage. If you run across problems, you can disable it per site, and it will remember that. And if you are an advanced user, this thing allows you infinite level. Not only can you use that matrix in order to fine-tune per-site behavior, but those rules, you can have user-based rules and write your own. So this thing does everything you could want it to. We'll be coming back to it, I'm sure, in the future. But I wanted to give people a sense for its background and history and to say this is what I think we want to use going forward.

Leo: Nice job. I just leave it in default mode. I haven't modified it at all. But it does look like there's some nice features you could turn on or off, depending on your...

Steve: Oh, that's what I love. It's great for easy use for non-power users. But, I mean, everything you could ask for, for drilling down as far as you want to for a power user. And this basically is an HTML firewall that allows us to decide what we want our browser to do with the content we receive from a website. And we will now all be players in this advertising ecosystem as it evolves.

Leo: Something's going to change. I can tell you that right now.

Steve: And I loved your note. You mentioned to someone, maybe it was on TWiT yesterday, mentioning going to Financial Times, FT.com?

Leo: Mm-hmm.

Steve: I am so frustrated by that, just as you are, because I'm following a tasty-looking link; and, bang, I smack right into the paywall, and I go no further. So it's like, well, okay.

Leo: But you can't have it both ways.

Steve: Nope.

Leo: We have to figure out something. I'm sure people will. I hope they will. Steve's at GRC.com. That's all free, no ads at all.

Steve: Because of SpinRite. SpinRite is what makes it all possible.

Leo: All you have to do is buy SpinRite, and everything's good, golden, and magical. Go there, too, if you to leave Steve...

Steve: [Crosstalk]

Leo: Yes. GRC.com/feedback. That's where you'll find the show, 16Kb audio as well as 64Kb audio, and nicely written transcripts. But you can also go to our site, TWiT.tv/sn, or even just subscribe. It's all over the Internet. Ten years ago we started this. It seems sensible that by now, if you claim to have podcasts on your tool, that you ought to have this one. So do subscribe, though. You don't want to miss an episode of Security Now!. Steve, we'll be back here next week.

Steve: Q&A.

Leo: I won't be.

Steve: Oh, that's right, you're in New York.

Leo: Or will I be? Wait a minute.

Steve: Tuesday, when do you leave?

Leo: I don't remember.

Steve: Okay.

Leo: I think it's a redeye. I think I will - I don't know. If it's not me, somebody wonderful will be here. People like Robert better than me anyway, so...

Steve: The podcast will go on.

Leo: Steve rests for no man. 11:00 a.m. Pacific. I'm sorry, 1:30 p.m. Pacific, that's 4:30 p.m. Eastern time, 20:30 UTC on TWiT.tv/live. Or as I said, get it after the fact at TWiT.tv/sn or GRC.com. You think questions next week?

Steve: Yes, Q&A next week, assuming that the world doesn't melt between now and then. And the world's been very cooperative lately.

Leo: Hackers take August off. It may get busy again. I don't know.

Steve: It could, although I think they're still recovering from Black Hat and Defcon. So we have a little bit of quiet for a while. And you have a great trip, if I don't talk to you next week. And we'll talk soon.

Leo: See you next week on Security Now!. Thanks, Steve.

Steve: Thanks, Leo.