Cobalt Strike

Description: This week we examine a devastating and still ongoing DDoS attack against the latest in a series of VoIP service providers. We check out the once again mixed blessing of last Tuesday's Microsoft patches, and we examine a welcome feature of Android 11 that's being backported through Android 6. We catch up with Chrome's patching of two more new zero-day vulnerabilities and attacks. Then we look at a "pwnage" email I received from Troy Hunt's Have I Been Pwned site. Was GRC pwned? I then have a quick sci-fi reminder for the end of the week, a SpinRite update, and a fun related YouTube posting. Then we'll wrap up by introducing the latest weapon in the malign perpetrator's arsenal, the powerful commercial tool known as Cobalt Strike.

High quality (64 kbps) mp3 audio file URL: http://media.GRC.com/sn/SN-837.mp3
Quarter size (16 kbps) mp3 audio file URL: http://media.GRC.com/sn/sn-837-lq.mp3

SHOW TEASE: It's time for Security Now!. Steve Gibson is here. We'll talk about one of the longest DDoS attacks ever - it's been going on for five days - who's behind it and what they are doing. We'll also talk about Patch Tuesday's mixed blessing, the 10th, count them, 10th zero-day patch for Chrome this year alone. And then a look at how a very powerful pentesting tool is being adapted by hackers to attack you. It's all coming up next on Security Now!.

Leo Laporte: This is Security Now! with Steve Gibson, Episode 837, recorded Tuesday, September 21st, 2021: Cobalt Strike.

It's time for Security Now!, the show where we cover your security, your privacy, your health, welfare, and wellbeing online with Steve Gibson of GRC.com. Hi, Steve.

Steve Gibson: Hey, Leo. Great to be with you again. This is our Tuesday after the second Tuesday, so we've got some news about how that went. What's really interesting is that right now, as we're recording this, we're in the fifth day of a continuous, devastating, still ongoing DDoS attack against the latest in a series of several VoIP service providers. So we're going to talk about that. As I mentioned, we're going to also check in on how last Tuesday's Patch Tuesday went. We've got some welcome feature of Android 11 that Google has announced they're backporting through to all the way back to Android 6. But I'm a little unsure about exactly how that works. We'll talk about that.

We catch up with Chrome's patching of two more new zero-day vulnerabilities and the attacks on those, being zero-days. We look at the pwnage email I received from Troy Hunt's Have I Been Pwned site. Was GRC pwned? I then have a quick sci-fi reminder for the end of the week, a SpinRite update, and a fun related YouTube posting. Then we're going to wrap up by introducing the latest weapon in the malign perpetrator's arsenal. It

is a powerful commercial tool known as Cobalt Strike, which unfortunately has been taken up rapidly by the underworld. And since I'm seeing increasing mentions of it, I thought we ought to put Cobalt Strike on all of our listeners' radar so that when we touch on it in the future, as unfortunately I'm afraid we're going to be, everyone will be nodding sagely. Ah, yes, Cobalt Strike. We know about that from Episode 837.

Leo: Picture of the Week time, Steve.

Steve: Indeed. Okay. So I figured after I saw this we had to share it, it was only fair, as a counterpoint to all of those horrifying and in many cases confounding wiring closet pictures that we were showing there for a while. This is as far at the opposite end of that spectrum as you could find. I titled this "Give that person a raise." I can't describe...

Leo: It's like a laced-up shoe. There's bundles of - by the way, excellent choice in color.

Steve: Yes, very nice.

Leo: They're kind of teal cables.

Steve: Yes. And white plastic tie wraps. They bundled...

Leo: Perfectly spaced.

Steve: And circular, so the bundles are round.

Leo: Yes.

Steve: And it's just - first one comes in on the top from the left. Second one comes in, drops below it, goes to the right. Third one comes in from the left again, slipping behind the second and the fourth.

Leo: It's like lacing up a sneaker. It's incredible.

Steve: It's just, like, first of all, this is not - whoever did this, this is not their first wrap-up of these things.

Leo: No. No.

Steve: I did note that it looks like it's coax as opposed to a multi-stranded Ethernet cable. Looked at it really closely. I think I'm seeing coax, like video terminations. So I don't know what this is, but still. You could certainly do this with standard 10Base-style Ethernet wiring, although I think this isn't. But oh, just a beautiful piece of work.

Leo: Yeah, this looks like a switcher, like the kind we use.

Steve: Yes.

Leo: And, you know, I have to say, Russell does a pretty good job. But this guy is definitely flexing.

Steve: It's world-class. And, okay. And notice that you don't even see the plastic bumps of the - well, okay. There's one.

Leo: They're over there. But you don't - but he's made it behind...

Steve: There are generally, yes, he clearly deliberately rotated them so that they're behind the main vertical bundle. It's just a piece of work.

Leo: It looks like a competition, frankly. It's pretty impressive, yeah.

Steve: It does. It's like, yeah. Maybe, yeah, I was going to say maybe you don't want to give the guy a raise because it may have taken about a year and a half.

Leo: Uh-huh.

Steve: To put this thing together. And it's like his boss is like, are you done with that thing yet? No, sir. I had to tear it all down and start over because we added a wire. It's like...

Leo: Oh, god, yeah.

Steve: And that's, see, that's the lesson I learned is it's always fun to tie all this stuff down. But it makes the installation brittle because you do need to move things around sometimes.

Leo: Right, right.

Steve: And this is a little hostile to that. So let's hope somehow they don't have to do that.

Leo: I always use Velcro for that reason.

Steve: There you go. Okay. So the DDoS attack on VoIP.ms, that's their domain name. After last week's podcast discussing the history, evolution, technology, and

countermeasures of the Internet's denial of service attacks, many of our listeners tweeted to let me know of - this was later in the week, it began on Thursday - of an ongoing multiday attack against the company VoIP.ms. And they are, as their name suggests, a Voice over IP provider whose website states has about 80,000 customers. Okay, now, that was last Thursday. Sadly, I don't know how many they have still.

Several of our listeners tweeted are among their customers. I had some DM traffic going back and back saying, you know, who were telling me about this. And I said, you know - and they were speaking in the DMs as if they had no phone service. And indeed that is the case. They were dark with their phone service. So until last Thursday, VoIP's Twitter stream had occasionally posted an advertisement to discuss and promote the various benefits of Voice over IP services. For example, on Monday, at the beginning of the week, on the 13th, they happily tweeted the rhetorical question: "Are you looking for a proven growth lever for your business? Are you making full use of cloud communications? Read this article and discover all the advantages offered by this technology." And then in their tweet they link to something.

So that was on the Monday. On the 16th they tweeted, quote, and this is at the leading edge of this attack: "We're aware" - they weren't aware yet of what. They tweeted: "We're aware of an issue that's keeping our customers to properly reach our website. There's a team actively working on the issue as we speak. And we expect it is fixed shortly. Thanks for your patience." So did somebody trip over a wire? We don't know.

A bit later: "We continue to work with our provider as our top priority, and we'll be ready to provide a post-mortem of this event once the service is reestablished. Thanks for your kind understanding, and please stay tuned!" And then: "After further investigation, our service provider confirmed they are currently facing a network attack resulting in a denial of services. For all practical purposes this is making our domain name, https://voip.ms, unreachable at the moment." And then: "For those of you who kindly switched from hostname to IP address and are within the U.S., we have made changes in the configuration with our upstream carriers to mitigate the issue for incoming calls, as well."

So they were beginning to play one of the games that people who - and as we know, I have too much experience with being on the receiving end of these things myself. You know, one of the things you do is - this was back in the Verio days when they were my ISP. I had a great relationship with them. And so we changed GRC's IP address, and then changed DNS, which would propagate. And for a while the attack would go away because it was still attacking the old IP. But inevitably it would catch up with the change. So, you know, changing IP addresses was one of the games back then.

They said in another tweet: "If you need to access your customer portal, you may be able to do so by adding a DNS entry to your localhost." And what they meant was your hosts file. They said: "Our public website and customer portal IP address is 173.231.187.61. See detailed steps below on how to do this for a Windows Computer." And then they provided instructions.

Okay. So this attack continued without letup from Thursday throughout the weekend. Then Monday, yesterday morning, they tweeted: "We want to assure you that all our energy and resources are being put into fighting this ransom DDoS attack." Which is the first time they made any reference to ransom. A little after 11:30 a.m. yesterday, BleepingComputer, that was tracking this, posted some news. They wrote: "Threat actors are targeting voice-over-Internet Provider VoIP.ms with a DDoS attack and extorting the company to stop the assault that's severely disrupting the company's operation. VoIP.ms is an Internet phone service company that provides affordable voice-over-IP service to businesses around the world. On September 16th, 2021, VoIP.ms became the victim of a distributed denial-of-service attack targeting their infrastructure, including DNS name servers.

"As customers configured their VoIP equipment to connect to the company's domain name, the DDoS attack disrupted telephony services, preventing them from receiving or making phone calls. As DNS was no longer working, the company advised customers to modify their hosts file to point the domain at their IP address to bypass DNS resolution. However, this just led the threat actors to perform DDoS attacks directly at the IP address, as well. To mitigate the attacks, VoIP.ms moved their website and DNS servers to Cloudflare. And while they reported some success, the company's site and VoIP infrastructure still have issues due to the continued denial-of-service attack.

"An announcement posted to the VoIP.ms website says: 'A Distributed Denial of Service attack continues to be targeted at our websites and POP servers.'" POP is Point of Presence, meaning the servers and services where they're delivering their VoIP. They said: "'Our team is deploying continuous efforts to stop this. However the service is being intermittently affected. We apologize for all the inconveniences.' At the time of this writing, the site is bouncing back and forth between being accessible and displaying a 500 Internal Server Error. Today, customers continue to experience issues with their telephone service, including loss of service, dropped calls, poor performance, and the inability to forward lines.

"On September 18th," BleepingComputer writes, "a threat actor using the name 'REvil' claimed responsibility for the attack and posted a link to a ransom note posted to Pastebin. This ransom note has since been removed from Pastebin, but BleepingComputer was told it asked for one bitcoin, or approximately at the time 45,000 USD, to stop the DDoS attacks. Soon after their original tweet, the threat actors raised their extortion demand to 100 bitcoins..."

Leo: Oh, boy.

Steve: "...or approximately $4.3 million at the time."

Leo: They realized what they had here. Wow.

Steve: Uh-huh. "The customers' responses to the attack against VoIP.ms have been mixed. Some feel that VoIP.ms should pay the ransom to restore services before they themselves do not lose customers." Meaning the customers of the companies who are using VoIP.ms's services. "At the same time, other VoIP.ms customers are vowing to stick with them and telling the company not to give in to the ransom demand." There's been lots of dialogue over on Reddit about this, with a lot of customers saying, you know, they give up. They waited out the weekend. They're jumping ship because, like, what's in it for them to stick around? BleepingComputer has contacted VoIP.ms with questions regarding the attack, but has not received a reply. And you can imagine the VoIP.ms folks have their hands full.

It strongly appears also that this is only the most recent component of a larger campaign. Earlier this month multiple VoIP providers in the U.K. were very similarly targeted. The site Unified Communications, www.uctoday.com, posted the news of the DDoS attacks in the U.K. two weeks ago. And I'm going to share what they wrote because everyone will note the similarities.

They wrote: "At least three U.K. VoIP providers have been hit by a DDoS attack, according to the Cloud Communications Alliance (CCA). In an email sent this morning, CCA said it has learned of a 'sophisticated, specific, and ongoing attack' believed to be from Russian cybercriminal organization REvil." We'll talk about why I think this is not the

case in a minute. "Two providers revealed they were victims of the attack last week, but the third company was not named by CCA. Poole-based Voip Unlimited said the attack on its core network started on August 31st and was continuous for 75 hours. An update from this firm this morning said it did not observe any further attacks over the weekend. London-based Voipfone reported its attack at the same time and said this morning that its services are fully operational, with traffic being closely monitored. Both firms saw their services disrupted over a three-day period.

"CCA said that the culprits were demanding ransom, starting at one bitcoin, but quickly increasing. The attack involves hammering a company's network with traffic of between 100 and 450 gigabits per second, often for up to 24 hours on multiple occasions. It starts with an attack on IP addresses used for SIP ingress and egress, but then migrates to other services. CCA said the attack is capable of evading some typical DDoS prevention measures.

"Voip Unlimited Managing Director Mark Pillow told The Register: 'At 2:00 p.m. 31st August, Voip Unlimited's network was the victim of an alarmingly large and sophisticated DDoS attack attached to a colossal ransom demand. UK Comms Council have communicated to us that other U.K. SIP'" - SIP of course is Session Initiation Protocol, that's the VoIP protocol - "'providers are affected and identified them as a criminal hacking organization called REvil who appear to be undertaking planned and organized DDoS attacks against VoIP companies in the U.K.'"

Okay. So here's what's interesting about this. The RPS (Requests Per Second) attacks and their mitigations which we covered last week, which Cloudflare and others are able to provide, that is, mitigations to requests per second attacks, are extremely specific to filtering web requests being made by web browser clients. In fact, anybody - sometimes when you go to a site behind Cloudflare, and I'm trying to think of several, but none jumps to mind. Brian Krebs' site, for example. You'll sometimes get like an intercept page which will say, you know, "One moment, verifying your browser." And then you proceed to the site.

Leo: Right. We've all seen that a lot. I just went to VoIP.ms's site, and you have to do a CAPTCHA, and it says hold on a second, blah blah blah. But it's still up.

Steve: Okay.

Leo: It's weird that they would attack the site and the POPs. Those wouldn't be the same address.

Steve: Probably not. Because the POPs - SIP is an HTTP-carried protocol, but not web. It just uses the HTTP protocol.

Leo: Right. Right.

Steve: And what we did read was that the attacks do tend to hit their VoIP infrastructure first, and then I guess just to sort of annoy people. Now, one reason of bringing down the web interface is that the customer portals, which is what they would use to migrate their service away, are web hosted. So by attacking the provider's web interface, that would be preventing people from getting to their own client portals and thus moving themselves somewhere else.

Anyway, so the problem is that the protections that we have for the web-based request per second attacks, and you just talked about them, you know, having that odd intercept page which is no doubt running JavaScript on the browser in order to positively confirm that it is a web client that wants to connect, and not somebody just making bogus queries. And especially if you had a CAPTCHA. You know, there's yet again another mitigation up in front before you're able to proceed. That would all be there to block a serious incoming attempt. There is no provider of large-pipe VoIP protocol DDoS protection. And that's the key.

So it has apparently dawned upon some miscreants somewhere that attacking the non-web servers of the Internet's global VoIP providers would be a new revenue source for extortion demands. They're probably not wrong, unfortunately. I agree with everyone that nothing about this sounds like the true REvil gang. The REvil gang conducted themselves with some professionalism. And first of all, you don't post an extortion demand, or REvil doesn't post it on Pastebin. That's not the way REvil operates. They also don't escalate from one bitcoin to 100. That, again, this just - it sounds like...

Leo: Amateurish, yeah.

Steve: Yeah. It sounds like a group who are wanting to ride on REvil's name without in any other way being REvil. And they've clearly got a big botnet. There's no doubt about that. They've got a significant botnet. And unfortunately we are in a world now, as we talked about last week, where big botnets are not uncommon. And what's occurred to them is that the anti-DDS defenses are not yet defending VoIP. And you can imagine that there's lots of other Internet-based services that may be mission-critical for which there are not yet any specific attack preventions.

So I was curious about where we are today, this morning. So I checked in with VoIP.ms's Twitter feed, and I found the following. They said: We want to assure you that all our energy and resources are being put into fighting this ransom DDoS attack. We do feel grateful to count on your patience and understanding on these unfortunate events. Please stay tuned on our social media feeds for further updates. Thank you. They said: "Our entire team are working 24/7 on implementing all the required measures in order to have the service back up and running. We definitely understand your current level of frustration; but please, due to the current circumstances, we can't disclose our action plan at this time. Don't doubt we are definitely using all internal and external resources that can be used in situations like these; and that as soon as this event is over, your service will be just like you used it before the attacks."

And then they added and finished: "All the team at https://VoIP.ms continues to work hard on recovering all services as soon as possible. With the help of internal and external specialists, all efforts and" - okay, blah blah blah. They said: "Multiple changes have been applied in order to improve connectivity and reliability, and we have observed positive changes in certain areas of the service. The following services have been recovered and are fully functional at the moment of this post: SMS, MMS, Recordings, Call Recordings, Conference Recordings. Please continue following our social media channels and issue tracker to not miss any important updates."

Okay. So it sounds as though actual VoIP calling remains offline due to this attack. They have messaging and playback of previous recorded audio working, but apparently not real-time interactive calling, five days downstream. And of course

the great danger to them is that VoIP services don't come with a great deal of customer lock-in. So some percentage of their previous 80,000 customers, all of whom will have been without mission-critical VoIP services since last Thursday, may be unable to tolerate

this outage and will port their numbers to another VoIP carrier. I know that because it's all in the public transaction log. There's a lot of conversation about this with people talking about who they went to and what they had to do to get there and blah blah blah. So, I mean, they're losing customers as this attack goes on.

Leo: It's exacerbated a little bit because it looks like they don't - they tout it anyway on their web page, no contract. So if you're month-to-month, it's very easy to move on.

Steve: Yup.

Leo: But I would guess for that reason it's - I don't know, but I would guess this is not a big enterprise VoIP solution. This is probably used by a variety of people for something simpler.

Steve: Right. Patched Tuesday's mixed blessing. This being the third Tuesday of the month, as I mentioned at the top of the show, we're able to get some perspective on what last Tuesday's monthly Patch Tuesday wrought. And "wrought" likely describes the mental state of at least a few of the industry's IT professionals following last Tuesday. It occurred to me that the well-worn term "side effects" is apropos for labeling things that go wrong after one of Microsoft's monthly patch batches are installed. Like a powerful pharmaceutical drug that a doctor prescribes, the application of the month's patch updates attempts to fix things that really do need to be fixed. But also like so many powerful pharmaceuticals, there can often be unwanted side effects to make the patient, or the IT administrator, think twice before swallowing the pill.

On the "you really do need to take this pill" side, last week's batch of updates fixed a total of 86 vulnerabilities, two of them being zero-days being actively exploited ITW, which you may recall is our recently coined abbreviation - and we didn't coin it, the industry has - In The Wild. The demographic breakdown of those was one denial of service vulnerability, two security feature bypass vulnerabilities, eight spoofing vulnerabilities, 11 information disclosure vulnerabilities, 16 remote code execution vulnerabilities, and 27 elevation of privilege vulnerabilities.

The biggest news was that, as we hoped would be true, Microsoft was indeed able to get that nasty MSHTML zero-day patched. That was the one that the hacker forums were having a field day with, the one we talked about last week where an ActiveX control was embedded in an Office doc that could cause IE's sort of disconnected-but-still-present engine core to be invoked to download a malicious site's content and then compromise the system. There was no terrific workaround because the weak workarounds that had been proposed had all been worked around.

So my advice and hope - I know. My god. My advice and hope was that Microsoft would be able - and I'm praying here, for those - I was off-camera. I'll bring my praying hands up higher - was that Microsoft would be able to deal with that one swiftly. And indeed they did. Since then, we learned that the flaw was being used to download and execute a malicious DLL that in turn would install a - wait for it - Cobalt Strike Beacon, and we'll know by the end of this podcast exactly what that means, onto the victim's computer.

The Cobalt Strike Beacon allows bad guys to obtain remote access to the machine, allowing them to exfiltrate files and also to spread laterally throughout the network. So in other words, this zero-day which was being exploited in the wild was being used to download the Cobalt Strike Beacon onto victims' machines. Someone who just, like

Pandora's inbox, said it really can't do any harm to just open one little Office document, can it?

Okay. So all that's the good news. And it provides sufficient reason for swallowing last week's update pill. But as I noted, this month's treatment was not without its side effects, which are causing, believe it or not, many network printer users to gag. The newly introduced trouble appears to be the result of Microsoft's attempt to resolve the last remaining outstanding, that is to say, still outstanding known vulnerability with their very troubled network printing management. The so-called PrintNightmare vulnerability is/was, and perhaps is still, being tracked as CVE-2021-36958. After applying last week's patches, which included a fix for 36958, Windows admins began reporting wide-scale network printing problems. For example, from Microsoft's own forum under the subject "Print server and Print Nightmare update," there was a posting last week. This just hit us this morning, too, 9/15/2021. No one can print to the network printers. I removed KB5005613 from our server and rebooted the server, and that fixed it. Had to do that at all eight of our branch offices, too. Microsoft updates...

Leo: Oh, geez.

Steve: I know. I know. And this guy said: "Microsoft updates seem to be more like hackers. Not professional." Somebody else wrote: "Today, 16 September 2021, I got the same problem, cannot print to printer on the server. Fortunately, I read this article and then I can assume that what happened to me is caused by BAD [all caps] Windows update. Then I check Updates History and find one update installed on 15 September (security update KB5005565). So I uninstall it and reboot and, YES [all caps], the printer works normally." And there's others I have in the show notes. I won't drag everyone across them. But anyway, causing widespread problems.

So the trouble is rather widespread, and uninstalling that security update KB5005565 appears to be the only way - and I should mention that's a function of which specific version of Windows you've got. The numbers vary depending upon Windows version. So your mileage will vary. That's the only way, uninstalling the relevant update, the only way to bring network printing back online. But uninstalling the fix restores the vulnerability that was allowing bad guys to perform remote code execution and gain local system privileges. The ransomware gangs Vice Society, Magniber, Conti are all known to have jumped onto this flaw and are now using it as long as it lasts - and apparently it's got another month, believe it or not - to obtain elevated privileges on compromised machines.

Which begs the question, how long will it last? There's apparently no telling. Remember that this remaining PrintNightmare vulnerability was originally reported privately to Microsoft back in December of 2020 by Victor Mata of FusionX, Accenture Security. One hopes that we might see a permanent and working fix for this serious flaw before we get to this December, by which time Microsoft will have known of it for a full year. A bitter pill to swallow, indeed. And I keep coming back to the still-unanswered question: Given Microsoft's virtually unlimited resources, how is it that they allow this to go on month after month after month? I don't know.

Okay. One last before we take a break and I get some water and catch my breath. Android. This is really welcome news. But I'm a little puzzled by it. Android to auto-reset app permissions on many more devices. Last Friday, Google announced that later this year support for Android 11's privacy protection feature - which debuted a little over a year ago. It was on September 8th of 2020. Okay. That feature proactively resets unneeded app permissions for applications that haven't been used in months. That

feature would be made available to billions - okay, potentially - of devices running older Android versions. Or at least to any of those which are able to update their OS.

And this is great news. Very much like having old and unneeded database information purge itself, which we wish database information did, but it doesn't, but auto-removing unneeded, and that is to say unused, permissions is obviously terrific security. It's unfortunate that so many older Android devices may never be able to obtain this. Google boasts of three billion devices running Android. But the timeline demographics of their versions suggest that older versions are not rushing to update.

I grabbed a beautiful chart which is in the show notes, and the chart makes the trend clear. Android 11's adoption is growing almost linearly, but maybe slowing down slightly as we'd expect. But nearly all of its growth is coming from Android 10. That is to say this chart, for those who don't see it - thanks, Leo, for putting it up - Android 11's adoption going up almost exactly mirrors Android 10's coming down. Which tells us that Android 10 devices are being updated with Android 11. The lines for Android 8 and 9, being the next two most recent versions, are drooping a little. So they're being upgraded, presumably to 11. But the older Androids are holding pretty firm. And this is what we'd expect; right?

Now, in time the batteries of those increasingly decrepit Android devices will fail. Or their radios will become obsolete as cellular technologies advance, and they're no longer able to find a cell tower that still supports 1G. So they'll eventually be tossed into the recycle bin, and their components and precious metals will hopefully be reprocessed to make new gadgets. But for those devices which can upgrade, this new and very useful feature will become available on all devices with Google Play services running Android from 6.0, which is API level 23, through Android 10. That is, you know, 11 already has it. So they're catching up: 6, 7, 8, 9, and 10.

In their posting, Google said: "Starting in December 2021, we are expanding this to billions more devices. This feature will automatically be enabled on devices with Google Play services that are running Android 6.0, API level 23 or higher. On these devices, users can now go to the auto-reset settings page and enable/disable auto-reset for specific apps. The system will start to automatically reset the permissions of unused apps a few weeks after the feature launches on a device."

Okay. Now, I studied Google's announcement, and I came away unsure whether this would be enabled by default on older devices. As we know, thanks to the tyranny of the default, the only thing that matters is whether this is enabled by default. The reason this is confusing is that Google's announcement says, and I'm quoting them verbatim: "This feature will automatically be enabled on devices with Google Play services that are running Android 6.0, API level 23 or higher. The feature will be enabled by default" - okay. So enabled on devices running Android 6.0 API level 23 or higher. "The feature will be enabled by default for apps targeting Android 11, API level 30 or higher. However, users can enable permission auto-reset manually for apps targeting API levels 23 through 29." Which is to say prior to the API level prior to Android 11.

So this suggests that even when older devices are updated, their older apps, which might remain unaware of the new support offered by Android 11 at API level 30, would not automatically be enabled. Presumably Google was unwilling to change this behavior without the user being aware. But if I'm understanding this correctly, I think that was a mistake. Just ask once for all older apps. It certainly isn't burdensome for the user to say no, no, I want to leave things as they are. That would not be unduly burdensome. And it's the older devices that are most in need of having their security shored up.

So if I understand it right, I do wish Google had decided to be a little more intrusive. But props to them in any event for adding this feature to Android. As I said, eventually

devices not running Android 11 will just disappear from the face of the Earth. They'll just, you know, they won't be usable any longer. And that'll be good because this idea of automatically downgrading the permission that you gave an app a long time ago that you haven't used, yay. That's the way the world should work.

Leo: Yeah. I should point out that a lot of the older Android versions are running on things like coffee makers. My toaster oven uses Android. A lot of my appliances run on Android.

Steve: Good point. Good point.

Leo: I doubt they'll ever be updated. And then I don't care for app privacy because I'm not running any apps.

Steve: Right, right.

Leo: It's more security I worry about. And again, they're not using a browser. They're just, you know, it's just...

Steve: They're just being a little OS.

Leo: Yeah, just a little OS. Because it's free. And so a lot of people do that. Doesn't even have Google services on it, I'm sure. We're going to take a break. But before we do, I want to mention a story that broke this afternoon. You haven't had time to see it. The FBI apparently knew the REvil ransomware key for three weeks before telling anybody. "The FBI refrained" - this is from The Washington Post, broke about noon today our time - "refrained for almost three weeks from helping to unlock the computers of hundreds of businesses and institutions hobbled by a major ransomware attack this summer."

Steve: Kaseya.

Leo: "The Bureau had secretly obtained the digital key needed to do so, according to several current and former U.S. officials. The key was obtained through access to the servers of the Russia-based criminal gang behind the July attack. Deploying it immediately could have helped the victims avoid what analysts estimate was millions of dollars in recovery costs. But the FBI held onto the key with the agreement of other agencies, in part because it was planning to carry out an operation to disrupt the hackers, a group known as REvil, and the Bureau did not want to tip them off. Also a government assessment found the harm was not as severe as initially feared." Tell that to the folks who paid the ransom.

Steve: Ooh.

Leo: "The planned takedown never occurred because before the FBI could go after them, the hackers disappeared and shut down their server." So you can't give FBI

credit for doing it, but you can mention the fact that they knew for three weeks what the decryption key was.

Steve: Ouch.

Leo: Yeah. I'm sure you'll...

Steve: You know, there are those kinds of tough calls, right, where you learn something from a channel, but you have to protect the channel's identity. So you can't make public what you've learned without endangering a resource that could potentially be even more vital in the future.

Leo: The classic case is Enigma.

Steve: Yes.

Leo: The Allies had cracked the Enigma machine. But if they had used the information coming through Nazi communications to save submarines and vessels sailing across the Atlantic, it would have given away the fact that they'd cracked the code.

Steve: Well, and I love the story, right, that they contrived to make the discovery of the U-boat missions accidental. Like, you know, having some weather craft fly over and, like, spot them. And then, of course, and thus be able to launch an attack. So, wow.

Leo: British spy agencies were quite inventive in that era. Just fascinating stories from that time. Anyway, I thought I'd pass that along to you.

Steve: Thank you.

Leo: The Kaseya story. They eventually did give it to Kaseya. But it took them three weeks to get it to them.

Steve: Wow.

Leo: All right. Back to Steve.

Steve: Speaking of keeping the bad guys out. Google patched the ninth and 10th in-the-wild zero-days in Chrome this year.

Leo: Oh, my god.

Steve: I know.

Leo: I mean, the browser is the vector for most attacks because it's what goes on the Internet. I don't think that Chrome is more insecure than anything else.

Steve: Nope. It is the big target.

Leo: Right.

Steve: And as we've said, browsers are just crazy complicated now. I mean, they're elevating themselves to the complexity of an operating system. So the good news is everyone running Chrome desktop for Windows, Mac, and Linux will now find themselves running 93.0.4577.82. So first number 93, last one that matters 82. And that's good news since the bad guys are scouring Chrome for any way in. And so far they've been discovering and exploiting those ways all this year at a rate slightly better than one per month, since we're in month nine, and this is zero-days nine and 10. Chrome's update to 93 blah blah dot 82 fixes a total of 11 security vulnerabilities, two of them being these zero-days exploited in the wild. One is an out-of-bounds write in the V8 JavaScript engine, and the other is a use-after-free bug in the Indexed DB API. So they're both memory bugs. The release notes commented that: "Google is aware that exploits for CVE-2021-30632 and 30633 exist in the wild."

The two zero-day vulnerabilities were disclosed to Google - get this, Leo - on September 8th. So Chromium gets fixed in less than a week. Five days, actually. This makes me even more glad that Microsoft decided that they were incapable of managing the ongoing development and maintenance of a state-of-the-art web browser. They adopted the Chromium core, thank goodness. As we know, our web browsers are today's primary attack surface, as you were just saying. We push them out onto the Internet and actively solicit unknown remote web servers to download and run their code on our machine.

Leo: Well, when you put it that way, it's amazing it works at all.

Steve: I know. This is where we all say in chorus: What could possible go wrong? It sounds insane. And it really is. So Lord knows that we cannot be waiting nine months for Microsoft to get around to fixing critical problems that they've been informed of. Google takes five days. Thank you. Happy to be behind a Chromium-based browser.

Leo: Well. And that's the other thing. Everybody else benefits, too, because of all the Chromium-based browsers.

Steve: Yeah.

Leo: It is the one good argument for having everybody run Chromium. Including Edge.

Steve: Because you get to multiply - exactly, yeah, yeah. Okay. So was GRC pwned, Leo?

Leo: What?

Steve: The last time we talked about Troy Hunt's excellent Have I Been Pwned web service, I noted that not only is it possible for individuals to subscribe to the site's autonomous notifications via email in the event of that email address appearing in any new breach, which Have I Been Pwned adds to their massive and growing breach database, but anyone who owns their own email domain, like TWiT.tv, can similarly submit their entire domain for notification. And I did that quite a while ago.

So I was interested to receive a notification on Sunday, day before yesterday, from HIBP (Have I Been Pwned), informing me that one or more email accounts belonging to GRC.com had just popped up in a new data breach. And sure enough, while gathering the cyber news of the past week to share with everyone here, I encountered the mention of that site which had been mentioned by Have I Been Pwned as having been breached. The site was Epik, E-P-I-K, dot com, which I had no memory of ever giving any email credentials to. I didn't know of it at all. And Wikipedia, however, knows about them, saying: "Epik is an American domain registrar and web hosting company known for providing services to websites that host far-right, neo-Nazi, and other extremist content." Uh, what? And a breach of that site had leaked some GRC credentials?

So the security industry coverage of this breach notes that Epik is the host of sites including Gab, Parler, and "The Donald." And the reputable site The Record, which was tipped off of the breach on Monday, first received a small subset of samples which was later followed by a full copy of the entire leak from an individual who claimed to be loosely associated with the Anonymous group the group which was proudly claiming responsibility for the breach and this extensive exfiltration. When The Record then reached out for comment last Tuesday, Epik denied the breach and the hackers' claims in an email reply. Epik wrote: "We are not aware of any breach. We take the security of our clients' data extremely seriously, and we are investigating the allegation." Signed, Epik spokesperson.

Okay. So despite Epik's denial, the data which The Record received in full and reviewed confirms the hacker's - the Anonymous group hacker's - claims. In a 32GB torrent file hosted through the DDoSecrets portal, the hackers included several SQL database dumps containing gigabytes of sensitive information such as domain ownership details, domain transactions, account details, and troves of personal data points.

Okay. So presumably a subset of this massive trove of information was then provided to Troy and was added to his Have I Been Pwned database, whereupon any of those who had previously signed up for notification would receive an email, just as I had. But I still had no idea how GRC could possibly have been part of the breach of such a domain registrar and host provider.

Leo: Especially since it didn't happen.

Steve: Well, exactly. Very good point. What are you talking about?

Leo: There was no breach.

Steve: We didn't notice any breach. Did you see it, Margaret? No. Okay. So I went over to HIBP to see what was up. It's possible to query Troy's terrific site for any matches on

a specific email address or, if one owns an entire domain, any matches against that domain. So I did that, and I found the two email addresses that had apparently been, and now I have air quotes, "leaked" by the breach of Epik.com. They were network-solutions-public-whois@grc.com, and whois2011-1@grc.com. Whew. So it was neither I nor Sue nor Greg nor any corporate email. Returning to the emailed notification I had received from Have I Been Pwned, I discovered that the notification that Troy's site sent had provided an interesting description of the breached site, along with the notification.

Have I Been Pwned said: "In September 2021," that is, this - "the domain registrar and web host Epik suffered a significant data breach, allegedly in retaliation for hosting alt-right websites. The breach exposed a huge volume of data, not just of Epik customers, but also scraped WHOIS records belonging to individuals and organizations who were not Epik customers. The data included over 15 million unique email addresses, including anonymized versions for domain privacy, names, phone numbers, physical addresses, purchases, and passwords stored in various formats." So, big whew. That all fits.

Leo: Was Epik scraping WHOIS?

Steve: Yes. Probably to spam for commercial purposes. Those two GRC.com email addresses would have indeed been scraped from some of GRC's public ICANN domain registrations. And who knows why this distasteful-seeming domain registrar might have had them? As I wrote in the show notes, perhaps to use as spamming for commercial purposes, "come on over here" registration solicitations. Because the date of registration shows, so they would know to send email addresses when your registration is close to being renewed.

Leo: Right, or to snipe your domain, maybe.

Steve: Could also be. Good point.

Leo: Who knows what their business was, yeah.

Steve: If it doesn't get renewed, then grab it, yup. Anyway, in any event, I thought it was very cool that Have I Been Pwned's proactive service works. It's the first time it's notified me since this was - you know, it had never occurred to me. And I wanted to make sure to take this opportunity to remind our listeners of Troy's useful and 100% free service. So thank you, Troy. And, whew, fortunately that was just a weird misfire.

Leo: Yeah.

Steve: Wow. I wanted to remind our listeners that the first three installments of Apple's 10-part miniseries based on Isaac Asimov's Foundation trilogy become available this Friday the 24th. As it happens, Lorrie and I had some dinner party plans which were moved from Friday to Thursday.

Leo: Whew.

Steve: Yes, which worked for me since I'd love to be able to set this Friday night aside for those first three Foundation episodes. I don't know if they're an hour each. Are they two hours each? I don't know how long each one is. But I can't wait. It looks wonderful.

And just another note, we will be needing to wait one more month for the remake of "Dune," which will be appearing on HBO Max, as well as for Apple TV's release of another 10-part miniseries, "Invasion," which chronicles an invasion of Earth by hostile extraterrestrials. So it's looking like the sci-fi lovers among us will finally be having some fun and some new visuals. I think that's mostly what I love about these films, Leo, is that they're just, you know, I mean, the plot lines are rarely surprising; right?

Leo: Well, you read the book; right? You know.

Steve: Oh, yeah, yeah, yeah. I know all about...

Leo: Yeah, you know what's going to happen, yeah.

Steve: ...what's happening, yeah.

Leo: I can't wait. I'm going to - Friday. That's exciting.

Steve: Also a note that my work on SpinRite is progressing. As I last noted, I was becoming uncomfortable with having written so much completely untested code. It's all since been tested. I decided to stop writing, to exercise and debug everything I've written. That's done. SpinRite has always incorporated drive benchmarks. They in fact, for a reason we'll see in a minute, I was just reminded that even SpinRite 1 apparently did that. The benchmarks perform deliberately repetitive and non-repetitive reads to measure various aspects of a drive's physical data read and cached data performance.

The benchmarks also have the benefit of giving SpinRite's new - the current benchmarks that exist in SpinRite, what will be 6.1, giving SpinRite's new IO abstraction system a rather thorough workout, which is precisely what it needs. So I have updated the benchmarking code to work with the new system. I was going to have to do that anyway, decided to do it now. And I'm nearly ready to make another test release available to the SpinRite testing gang.

Once the dust has settled from that, I'll finish updating the data recovery code, which was where I was when I decided to pause and make sure everything I'd written was already working. Then I'm pretty sure that the logging system will need adjusting. I've already been in that code updating things for everything that changed about 6.1, but nothing was running back then, so I wasn't able to confirm that the various new formatting things I had written were working.

And while I'm sure that other stuff will come up, it really does feel as though we're sort of getting to the point where we can see the light at the end of this very long tunnel. As I had mentioned before, I've pretty much rewritten SpinRite. I underestimated the amount of work that would be necessary. But I'm glad I did because this is the new foundation for everything going forward.

And in the meantime, I have something very fun to share. A guy by the name of Adrian Black has a retro computing YouTube channel called "Adrian's Digital Basement," to

which a bunch of our listeners subscribe. I know that because I began receiving tweets informing me of Adrian's recent posting. A couple of Saturdays ago, Adrian was playing with a sort of amazing emulator for the very first MFM (Modified Frequency Modulation) hard drives of the sort that the first IBM XT could be equipped with. So this is an emulator which pretends to be an MFM drive. It acts exactly like an MFM drive. You know, 17 sectors per track and all that, that we had back then. And, you know, it's sort of in the way that like the machines that you see blinking behind me have a chip that emulates a PDP-8, and we've got PDP-11s, Leo, where there's actually a little Linux machine behind them doing, you know, reading the switches and blinking the lights, again as an emulator. So this thing pretends to be a hard drive.

So anyone who recalls the iconic golden-shelled 10MB Seagate ST-225 will see one sitting there on Adrian's workspace. You can see it there in the middle over on the right.

Leo: The question is why would you want to emulate this?

Steve: Uh-huh. Well, on the screen there to the left...

Leo: Because you can. Yeah?

Steve: On the screen there to the left...

Leo: Oh, yeah, I recognize that screen.

Steve: ...is an ancient version of SpinRite. This MFM hard drive emulator was not transferring data very quickly, and Adrian suspects that perhaps that's because the emulator was low-level formatted with a one-to-one sector interleave. Remember, it's a drive. It's pretending to be an MFM hard drive. And Leo, I do agree with you, like, okay, how much time did the person who created this thing have on their hands? Okay. So if the interleave is one-to-one, and that's too tight, that means there's no sector interleaving at all. But the controller he has the emulator hooked up to is not fast enough to run it one to one. It needs some interleaving. It's not quick enough to catch the immediate next sector after reading the previous one.

Okay. So without batting an eye in this video, which is this week's grc.sc shortcut, thus grc.sc/837, Adrian fires up an ancient copy of SpinRite and has SpinRite examine the current speed and interleave, then has SpinRite optimize the drive's interleave, just like in the old days.

Leo: Oh, I remember you could do that. That's right.

Steve: Yes, and you see it actually doing it. It successively tries each interleave in turn and measures the drive's data transfer performance...

Leo: I remember that.

Steve: ...at each interleave setting.

Leo: Holy cow.

Steve: And then builds a correspondence table. So this week's GRC shortcut, as I noted, it will jump you 41 minutes in, to just before Adrian begins to run his ancient copy of SpinRite on the emulator. Again, https://grc.sc/837. Anyway, I know that some of our listeners will get a kick out of it.

Leo: Could he do that with SpinRite 6? Or is that too ancient a technology?

Steve: No, I took out all the interleaving code. I think 3.1 still had it. I don't remember whether 5 did. There have been requests that we've had from people who have like an 8088 or an 8086 because my newer code started using some of the instructions from the 286 or the 386, just because they're just so good. I had to use a couple, like, well, for example, RDTSC is Read the Time Stamp, which provides high-resolution timing information. There was some that was okay back on the 8088 and 86, which is all I was, you know, all I could use is what I had back then. But the point is that we can provide a SpinRite 3.1, a working SpinRite 3.1, for people who have a need. That will perform interleaving. And, frankly, I don't recognize, it's been so long, I don't recognize the screen, like which version of SpinRite it is from looking at the screen. So I'm sure someone will tell us.

Leo: The good old days.

Steve: Ah, yes.

Leo: I realize the reason you might want an MFM drive is if you were trying to emulate something that thought it was going to be running on an MFM drive.

Steve: Yes, actually. And he makes reference to that. I think he has an actual PDP-8.

Leo: There you go.

Steve: That had a hard drive controller, and they're just, you know, all of them...

Leo: So he wants to use a modern drive, but he has to emulate - or modern storage of some kind. But he has to emulate it, yeah.

Steve: Yes.

Leo: Yeah, that makes sense.

Steve: Yes.

Leo: So there is a good reason to do that. I mean, here you are, sitting in front of a bunch of PDPs. I'm sitting in front, I mean, obviously we don't have any affection for the old kind of computing, not at all.

Steve: That's right.

Leo: Of course they're all running Arduinos and Raspberry Pis.

Steve: Right.

Leo: But it's a lot easier. I don't think I can run an MFM drive on this Altair 8800. I think I'd have to run a - I don't know what I'd - a paper tape reader. I don't know. Okay, Steve. Let's get to the meat of the matter today.

Steve: Okay. So we're going to learn about, sadly, a new tool which has fallen into the bad guys' hands. CobaltStrike.com proudly introduces newcomers to their quite pricey offering with the headline "Software for Adversary Simulations and Red Team Operations." So they go on to explain. This is a product that they are offering. "Adversary Simulations and Red Team Operations are security assessments that replicate the tactics and techniques of an advanced adversary in a network. While penetration tests focus on unpatched vulnerabilities and misconfigurations, these assessments benefit security operations and incident response.

"Cobalt Strike gives you a post-exploitation agent and covert channels to emulate a quiet long-term embedded actor in your customer's network. Malleable command-and-control lets you change your network indicators to look like different malware each time. These tools complement Cobalt Strike's solid social engineering process, its robust collaboration capability, and unique reports designed to aid blue team training.

"Raphael Mudge created Cobalt Strike in 2012 to enable threat-representative security tests. Cobalt Strike was one of the first public red team command and control frameworks. In 2020, HelpSystems acquired Cobalt Strike to add to its Core Security portfolio. Today, Cobalt Strike is the go-to red team platform for many U.S. government, large business, and consulting organizations." In other words, this is a top-level, state-of-the-art, high-end tool used, well, intended to be used by good guys to perform benign post-penetration red team operations. The bad news is it's so good that it has also rapidly become the go-to platform for non-simulated real world malware post-penetration network infiltration.

We've seen many examples over the years of well-designed and well-meaning utilities being commandeered and abused for malicious purposes. For example, Sysinternals' Mark Russinovich created PsExec. Its description at Microsoft, who as we know purchased Sysinternals back in the summer of 2006, the PsExec description says - and read this as something from the malicious perps' viewpoint. They said: "Utilities like Telnet, and remote control programs like Symantec's PC Anywhere, let you execute programs on remote systems; but they can be a pain to set up, and require that you install client software on the remote systems that you wish to access." Oh, how pesky. "PsExec is a lightweight Telnet replacement that lets you execute processes on other systems, complete with full interactivity for console applications, without having to manually install client software."

Leo: Don't you hate that?

Steve: What could possibly go wrong? Isn't this wonderful? How convenient. Unfortunately for everyone. They say: "PsExec's most powerful uses [uh-huh] include launching interactive command-prompts on remote systems and remote-enabling tools like IPConfig that otherwise do not have the ability to show information about remote systems."

Leo: Oh, man.

Steve: I know. At this point once again we say in unison, what could possibly go wrong?

Leo: What could possibly go wrong?

Steve: Oh, and it even said, it had a little footer: "Note: some anti-virus scanners report that one or more of the tools are infected with a 'remote admin' virus. None of the PsTools contain viruses, but they have been used by viruses, which is why they trigger virus notifications." Yeah. And of course another example real quick, I chose the lovely Remote Utilities solution for my own and for Lorrie's remote system control needs for the same reasons that it has also become the remote control solution of choice for nameless miscreants across the Internet. It's not its fault, of course. Just like a compiler compiles code, it can't help it if the code it's compiling is malware.

Okay. So today, Cobalt Strike deserves and receives our attention because it appears that we're just at the beginning of a wave of malicious exploitation which unfortunately is all being enabled by Cobalt Strike. Back in July, the security firm Proofpoint published, and then recently updated, a report titled: "Cobalt Strike: Favorite Tool from APT to Crimeware." Of course APT, Advanced Persistent Threat.

One of the terms we'll encounter today and in the future is "Beacon," or "Cobalt Strike Beacon." The Beacon is the bad bit that's infiltrated into an unwitting victim's machine. And unfortunately it was quite well designed. Okay. Listen to how Cobalt Strike themselves boast about Beacon's capabilities. They write: "Beacon is Cobalt Strike's payload to model advanced attackers. Use Beacon to egress a network over HTTP, HTTPS, or DNS." By the way, also SMB.

"You may also limit which hosts egress a network by controlling peer-to-peer Beacons over Windows-named pipes. Beacon is flexible and supports asynchronous and interactive communication. Asynchronous communication is low and slow. Beacon will phone home, download its tasks, and go to sleep. Interactive communication happens in real-time. Beacon's network indicators are malleable. Redefine Beacon's communication with Cobalt Strike's malleable C2 (command-and-control) language. This allows you to cloak Beacon activity to look like other malware, or blend in as legitimate traffic."

And then they go into some additional detail which should chill the blood of any CISO. They say: "Right-click on a Beacon session and select 'interact' to open that Beacon's console. The console is the main user interface for your Beacon session. The Beacon console allows you to see which tasks are issued to a Beacon and see when it downloads them. The Beacon console is also where command output and other information will appear." So, right, a complete happy point-and-click user interface for the bad guys to manage all the Beacons that they have managed to install in their victims' machines.

"Be aware that Beacon is an asynchronous payload. Commands do not execute right away. Each command goes into a queue. When the Beacon checks in, that is, connects to you, it will download these commands and execute them one by one. At this time, Beacon will also report any output it has for you. If you make a mistake, use the clear command to clear the command queue for the current Beacon.

"By default, Beacons check in every 60 seconds. You may change this with Beacon's sleep command. Use sleep followed by a time in seconds to specify how often Beacon should check in. You may also specify a second number between 0 and 99. This number is a jitter factor." Right? Because to introduce random check-in timings in order to further hide the Beacon. "Beacon will vary each of its check-in times by the random percentage you specify as a jitter factor." Ah, so it's a percentage. "For example, sleep 300 20 will force Beacon to sleep for 300 seconds with a 20% jitter percentage. This means Beacon will sleep for a random value between 240 to 300 seconds after each check-in." Ah, so it's unidirectional from the time you specify.

"To make a Beacon check in multiple times each second, try sleep 0. This is interactive mode. In this mode, commands will execute right away. You must make your Beacon interactive before you tunnel traffic through it." So it's also a tunneling technology. That's what they meant when they said it will export a network. It will literally export a network connection. "A few Beacon commands - browser pivot, desktop, et cetera - will automatically put Beacon into interactive mode at the next check in."

So running commands: Beacon's shell command will task a Beacon to execute a command via cmd.exe on the compromised host. When the command completes, Beacon will present the output to you. Use the run command to execute a command without cmd.exe. The run command will post output back to you. The execute command runs a program in the background and does not capture output. Use the powershell command to execute a command with PowerShell on the compromised host. Use the powerpick command to execute PowerShell cmdlets without powershell.exe. This command relies on the Unmanaged PowerShell technique developed by Lee Christensen. The powershell and powerpick commands will use your current security token. The psinject command will inject Unmanaged PowerShell into a specific process and run your cmdlet from that process.

The powershell-import command will import a PowerShell script into Beacon. Future uses of the powershell, powerpick, and psinject commands will have cmdlets from the imported script available to them. Beacon will only hold one PowerShell script at a time. Import an empty file to clear the imported script from Beacon.

The execute-assembly command will run a local .NET executable as a Beacon post-exploitation job. You may pass arguments to this assembly as if it were run from a Windows command-line interface. This command will also inherit your current token. If you want Beacon to execute commands from a specific directory, use the cd command in the Beacon console to switch the working directory of a Beacon's process. The pwd command will tell you which directory you're currently working from.

Last, but not least, Beacon can execute Beacon Object Files without creating a new process. Because won't that be handy. Beacon Object Files are compiled C programs, written to a specific convention, that run within a Beacon session. Use inline-execute followed by args to execute a Beacon Object File with the specified arguments. Use the spawn command to spawn a session for a listener. The spawn command accepts an architecture, for example x86 or x64, and a listener as arguments. By default, the spawn command will spawn a session in rundll32.exe. An alert administrator may find it strange that rundll32.exe is periodically making connections to the Internet. Find a better program, for example Internet Explorer,

and use the spawnto command to state which program Beacon should spawn sessions to.

The spawnto command requires you to specify an architecture (x86, x64) and a full path to a program to spawn, as needed. Type spawnto by itself and press enter to instruct Beacon to go back to its default behavior. Type inject followed by a process id and a listener name to inject a session into a specific process. Use ps to get a list of processes on the current system. Use inject [pid] x64 to inject a 64-bit Beacon into an x64 process. The spawn and inject commands both inject a payload stage into memory. If the payload stage is an HTTP, HTTPS, or DNS Beacon, and it can't reach you, you will not see a session. If the payload stage is a bind TCP or SMB Beacon, these commands will automatically try to link to and assume control of these payloads.

Use dllinject [pid] to inject a Reflective DLL into a specific process, by process ID. Use the shinject [pid] architecture and path command to inject shellcode from a local file into a process on the target. Use shspawn architecture and path to spawn the "spawn to" process and inject the specified shellcode file into the process. Use dllload process ID and path to load an on-disk DLL into another process of your choice. Use ppid and then pid to assign an alternate parent process for programs run by your Beacon session. This is a means to make your activity blend in with normal actions on the target. I mean, these guys could not have written a more potent horrifying tool for bad guys to have gotten a hold of. And I won't keep, I mean, it goes - there's the runu, the spawnu, and others.

This is a dream come true for the underworld. It was deliberately and consciously designed by advanced cybersecurity experts specifically to operate as covertly as possible using every advanced trick in the book. And it's now loose, being proactively used, not for red versus blue team training, but in the wild by threat actors. And its use is spreading, not just in depth, but dramatically in breadth.

Here's what Proofpoint describes and found. They said: "In December 2020 the world learned about an expansive and effective espionage campaign that successfully backdoored the popular network monitoring software" - wait for it - "SolarWinds. Investigators revealed tools used by the threat actors including Cobalt Strike Beacon." In other words, Cobalt Strike Beacon, the malicious use of it, was behind the now-infamous Solar Winds attacks.

"This campaign," they wrote, "was attributed to threat actors working for Russia's Foreign Intelligence Service, a group with Cobalt Strike in their toolbox since at least 2018. This high-profile activity was part of a clever attack chain enabling advanced threat actors to surreptitiously compromise a relatively small number of victims. The tool used and customized to fit their needs is almost" - meaning Cobalt Strike Beacon - "is almost a decade old, but increasingly popular."

Cobalt Strike debuted in 2012 in response to perceived gaps in an existing red team tool, the Metasploit Framework. In 2015, Cobalt Strike 3.0 launched as a standalone adversary emulation platform. By 2016, Proofpoint researchers began observing threat actors using Cobalt Strike. Historically, Cobalt Strike used in malicious operations was largely associated with well-resourced threat actors, including large cybercrime operators like TA3546, also known as FIN7, and APT groups such as TA423, also known as Leviathan or APT40.

Proofpoint researchers have attributed two-thirds of identified Cobalt Strike campaigns from 2016 through 2018 to well-resourced cybercrime organizations or APT groups. That ratio decreased dramatically the following years. Between 2019 and present, just 15%, down from 66, 15% of Cobalt Strike campaigns were attributable to known threat actors. In other words, the word got out that Cobalt Strike was the tool to use. And after running through that command reference, of course. And everyone began picking it up and using it.

Proofpoint notes that: "Threat actors can obtain Cobalt Strike in a variety of ways: purchasing it directly from the vendor's website, which requires verification; buying a version on the dark web via various hacking forums; or using cracked, illegitimate versions of the software. In March 2020, a cracked version of Cobalt Strike 4.0 was released and made available to threat actors."

And in making even more clear the appeal that Cobalt Strike offers, Proofpoint explains: "Cobalt Strike is used by a diverse array of threat actors. And while it is not unusual for cybercriminal and APT actors to leverage similar tooling in their campaigns, Cobalt Strike is unique in that its built-in capabilities enable it to be quickly deployed and operationalized, regardless of actor sophistication or access to human or financial resources." In other words, you don't need to be a genius to use it. You don't need a team, and you don't need any money.

"Cobalt Strike is also session-based, that is, if threat actors can access a host and complete an operation without needing to establish ongoing presence, there will not be remaining artifacts on the host after it is no longer running in memory." And remember that's what we saw with the earlier SolarWinds hacks; right? It was very hard to get attribution because these things were very careful to erase their footprints. Cobalt Strike, you know, it was designed to do that. So in essence, they said, they can hit it and forget it.

"Threat actors," they continue, "can also use the malleability of Cobalt Strike to create customized builds that add or remove features to achieve objectives or evade detection. For example, APT29 frequently uses custom Cobalt Strike Beacon loaders to blend in with legitimate traffic or evade analysis. For defenders, customized Cobalt Strike modules often require unique signatures, so threat detection engineers may be required to play catch-up with Cobalt Strike use in the wild.

"Cobalt Strike is also appealing to threat actors for its inherent obfuscation. Attribution gets more difficult if everyone is using the same tool. If an organization has a red team actively making use of it, it's possible malicious traffic could be mistaken as legitimate. The software's ease of use can improve the capabilities of less sophisticated actors. For sophisticated actors, why spend development cycles on something new when you already have a great tool for the job? Proofpoint data shows Cobalt Strike is a popular tool for everything from strategic compromises to noisy, widespread campaigns."

So we have a top-of-the-line tool which was designed to enable good guys to stealthily penetrate and inhabit their own networks for the purpose of testing and training anti-penetration and intrusion detection teams. It licenses for $3,500 per seat. On the other hand, it got loose, was cracked, is now on the dark web for free, and it's out of control from its publisher. Now this highly professional high-end tool which no script...

Leo: Script kitty, meow.

Steve: ...which no script kiddie - thank you, Leo - would have ever been able to create for themselves is freely available, being used in the wild on a much more than daily basis. There is little doubt that we'll be encountering the term "Cobalt Strike" in the future. Now at least we'll all know exactly what it means and entails when we talk about it. Wow.

Leo: David Redekop in our Discourse, our Club TWiT Discourse, says...

Steve: Oh, cool. Hi, David.

Leo: Yay. Hey, David. He says you can easily block Cobalt Strike C2 connections because CSC2 hosts never use domain names, only hard-coded IP addresses, which is I think what you'd expect with a pen test tool. You're not going to do DNS on it. You're going to have the IP address, so you just disallow traffic that wasn't first resolved by DNS.

Steve: Yeah, but that's tricky. That's going to require a special...

Leo: Yeah, I don't know how to do that.

Steve: ...filter, yeah.

Leo: Yeah.

Steve: David has a lot of experience with running local small DNS servers. And I would imagine, if he's saying that, he has a filter that's smart enough to do that as part of his offering.

Leo: Yeah, yeah. He says Pegasus did that, as well. It's nice to have smart people in our chat. You want to get a copy of this show, I'll tell you how you can do that, a couple of ways. You can go to Steve's site first. That's GRC.com. He has 16Kb audio for the bandwidth-impaired, 64Kb audio for those with ears. He also has transcripts for those with eyes, nicely written. You can use them to read along while you're listening, to read standalone, or to use for search. In fact, that's one of the best benefits of the transcripts. It allows searching for a specific part of any show, all 837, and going right to that part of the show. Thanks for doing that, Steve, we appreciate it, and Elaine Farris, who does those transcriptions.

Steve also has a copy waiting for you of something called SpinRite. Perhaps you've heard of it. Version 6 is current, the world's best mass storage maintenance and recovery utility. You can also, if you get 6.0 now, get a free upgrade to 6.1 and kind of follow along with the developments.

Steve: It was really cool listening to Adrian on that YouTube video. He just - he so matter-of-factly talked about it.

Leo: He was going to run SpinRite.

Steve: He was like, "And then I inhaled." And he said, you know, "And then I ran SpinRite." It's like, wow, it was just - it was like, so cool.

Leo: It's actually amazing, I guess it's because he works with this vintage equipment, that he remembered that, oh, yeah, I can fix the interleave. Just run SpinRite. It'll pick a better interleave level. So cool, yeah.

You can also catch the show on our website, that is, TWiT.tv/sn for Security Now!. There's a full-time YouTube channel with all the episodes. And you can also watch it live. I think a lot of people like to watch it happen live. You're more than welcome to. We have live audio and video streams at TWiT.tv/live. If you are watching live, get in the Discord with our Club TWiT members or our free IRC channel, irc.twit.tv. You can watch and chat at the same time. After the fact, the website, YouTube, you might have conversations with other TWiT members in our forums, that's TWiT.community, or our Mastodon instance, TWiT.social. There are lots of places you can interact with the family of TWiTs. Of course, Steve has his own forums at GRC.com.

We do the show of a Tuesday afternoon, right after MacBreak Weekly, usually somewhere between 1:30 and 2:00 p.m. Pacific. That's 4:30 Eastern; that's 20:30 UTC. And although I guess are some countries off summertime already? I think they might be. But we're still on summertime until after Halloween, till October. Yabba Dabba Do. Somebody listening says I need a copy of SpinRite right now.

Steve: Thank you.

Leo: It's awesome. Thank you, Steve. Have a wonderful week. Enjoy "Foundation."

Steve: Oh, we'll be talking about it next Tuesday.

Leo: Oh, that means I have to watch, too. You're going to watch all three episodes. All right. I will, too. All right. Thanks, Steve.

Steve: Okay, buddy. Bye.