SpiKey

Description: This week we look at a new Chrome remote code execution flaw, some interesting news of three new ransomware victims, an emergency patch from Microsoft, the emergence of amateur RDP exploiters, the 15th birthday of the Zero Day Initiative, finally a good Windows 10 garbageware remover, recommendations of several of my most recommended remote networking utilities, then a bit of miscellany and SpinRite news. Then, finally, we examine a really terrific new high-tech hack against low-tech locks and their keys.

High quality (64 kbps) mp3 audio file URL: http://media.GRC.com/sn/SN-781.mp3
Quarter size (16 kbps) mp3 audio file URL: http://media.GRC.com/sn/sn-781-lq.mp3

SHOW TEASE: It's time for Security Now!. Steve Gibson is here. Coming up, what do the University of Utah, Jack Daniels, and Carnival Cruise Lines have in common? Steve has the answer. We'll also talk about the number one way ransomware gets on your system. It's not what you think. Steve has an explanation. And then we'll take a look at an amazing bit of research showing how you can pick a lock just by listening. It's all coming up next on Security Now!.

Leo Laporte: This is Security Now! with Steve Gibson, Episode 781, recorded Tuesday, August 25, 2020: SpiKey.

It's time for Security Now!, the show where we cover your safety, your privacy, your security online with our majordomo, the man in charge, Mr. Steve Gibson. Hello, Steve.

Steve Gibson: Yo, Leo.

Leo: Good to see you.

Steve: I did confirm that this is the launch of Year 16.

Leo: Wow.

Steve: It was August 19th, which was last Wednesday, of 2005 that was our maiden voyage on this journey that you proposed 15 years ago.

Leo: Thank god you didn't have a crystal ball.

Steve: No, this has been, Leo, you're one of my longest lasting relationships.

Leo: Same here. Same here. And I've never cheated on you, Steve, not once. So there.

Steve: Wait a minute. There have been others.

Leo: There have been others. Okay, there have been others, yes.

Steve: I happen to know there have been others.

Leo: I've spoken to Bruce Schneier.

Steve: But I've been able to share in those experiences, as well.

Leo: That's true. In fact, we've got another one...

Steve: Strange relationship. Strange relationship.

Leo: ...coming up, you and I. We're going to be doing an event in October to celebrate security month.

Steve: Yes, October 1st. October 1st.

Leo: Yeah, that'll be good. We'll have details of that.

Steve: Until then we have a little bit of a play on words. And it's not my play on words, it's theirs, SpiKey, S-p-i-capital K-e-y. We're going to talk about an incredibly cool bit of technology and an opportunity to sort of step back a little bit and look at the landscape of the whole, like, low-tech meets high-tech, essentially. But we'll get to that. First we're going to talk about a new Chrome remote code execution flaw which happily people will be patching when they move to Chrome 85. I forgot to look and see whether I'm on 85 today. It's supposed to be happening today.

We also have some interesting news of three new ransomware victims, some two of the three very well known; an emergency patch from Microsoft and a little bit of the back story around that; the emergence of amateur remote desktop protocol exploiters; and, weirdly coincident with this podcast's 15th birthday, which occurred - was it last week or this week? Anyway, I get - these zero-based or one-based issues are a constant source of programming bugs and also brain bugs for me. Anyway, the 15th birthday of the Zero Day Initiative. Actually, I guess it would have been last week because theirs happened one day after ours.

Leo: This is the world's first null-terminated podcast, so I understand your confusion, yeah.

Steve: That's perfect, yes, very good. We also have, I found, finally, a good Windows 10 garbageware remover that I'm going to talk about. I'm going to offer some recommendations of several of my most successful remote networking utilities. We've got a bit of miscellany, some SpinRite news, and then we're finally going to examine a really terrific new high-tech hack against low-tech locks and keys. And of course we've got a really good...

Leo: Funny, very funny Picture of the Week this week, yeah.

Steve: Yeah, yeah.

Leo: And also apropos. Great. I'm excited. Another great show in the offing. Picture of the Week?

Steve: So, yeah. It's a four-frame cartoon. The first one shows the very familiar Internet Explorer "e" with some weird black shrouded hand, skeleton hand, trying to pull it offscreen. And the hand gives up and sort of breaks free. And then we see in the third frame that we have the Grim Reaper. And Grim Reaper is saying - oh, and in the first frame it says "It's time to go." And the Internet Explorer resists, apparently. And so then the Grim Reaper in the third frame says, "Let's go." And then the fourth frame is "Internet Explorer is not responding."

Leo: Very familiar.

Steve: Which, yes, which is familiar to us all. And the Grim Reaper is puzzled that it's unable to remove it. And this was apropos of something that I meant to talk about last week, but didn't. Because just under a year from now, on August 17th of 2021, the use of IE11, our last IE, will no longer be supported for Microsoft's online services like Office 365, OneDrive, Outlook, and more. And of course this is significant. We often hear Paul and Mary Jo talking about how corporations have built IE in as a component of their infrastructure, with custom apps and things. It's all glued in. And they've got a year until it really stops being supported. And also Microsoft will be ending support for IE11 with Microsoft Teams web app later this year, and all support ending on November 30th.

So the clock is ticking, and corporations really need to be looking at Edge. Now, I don't know whether the IE11 compatibility mode built into Edge will continue or not. But for what it's worth, standalone IE, you know, sooner or later the Grim Reaper is going to succeed. And in fact that Grim Reaper may actually be named Microsoft.

Google Chrome users should today be moving to Chrome 85. Most users don't need to do anything, which is good. It just updates itself. But this will fix a potentially serious remote code execution vulnerability in Chrome's WebGL rendering engine. It's a use-after-free read flaw which was discovered by Cisco's Talos security group, and it was responsibly reported to Google more than three months ago, back on May 19th. Google quickly put the fix into their early-release cycle in the Dev and the Beta channels just a

couple weeks later, in early June. And the stable channel, which I and most of us use, is expected to be receiving that fix today when Chrome moves from 84 to 85.

I looked last night, I was on 84. I actually, I could look right now, come to think of it, because I've got Chrome right here. Let's see. Help > About and checking. Oh, updating Google Chrome, yes. I'm still on 84, but in a few moments I will be on 85. So you may need to go to Help > About to kind of give it a little kick. That sometimes is necessary.

Leo: I think if you close it, right, it will download it. And then if you close it, it will re-up. But most of us never close our browsers.

Steve: That's right. For me, Firefox is just open statically, and you need to go to Help > About, and then it goes, oh, yeah, thanks for asking, and then it updates itself and sometimes needs to do a - typically needs to close and then reopen with all the tabs surviving, fortunately, because as we saw last week I need all of those thousand tabs that I have open. No, it's not a thousand, but it has a scroll bar. So, yeah, wouldn't fit on one screen.

So what do the University of Utah, Jack Daniels Whiskey, and Carnival Cruise Lines all have in common? Well, Friday, last Friday, the University of Utah revealed that it had paid a ransomware gang $457.59.

Leo: No, 457,000.

Steve: Yes, sorry. I got confused at the comma.

Leo: That would have been a bargain.

Steve: $457,059.

Leo: Small difference.

Steve: Which sort of begs the question, where did they get that number, now that I finally got it out correctly: $457,059.

Leo: It's probably the bitcoin conversion. Yeah, like half a million bucks. It's like the bitcoin conversion thing. It probably was a certain...

Steve: Yeah, it had to have been that.

Leo: A hundred bitcoins or something.

Steve: Yes, some round number of bitcoin that turned out to be that. And what's interesting is that was not to obtain the decryption key for their files. They didn't need it because it turns out that very few of their files were encrypted. But rather - and Leo, I

know this goes to the thing where you just kind of like grit your teeth - to purchase the promise from the extortionists that the student information that had been exfiltrated beforehand...

Leo: Oh, boy. Oh, wow.

Steve: Yeah.

Leo: That's so bad.

Steve: Would not be publicly released.

Leo: You're going to see more and more of this one. This is big.

Steve: Yeah. So they're hoping that there is honor among thieves and that these guys will keep their word.

Leo: Incentive to keep your word is that, if you want others to pay you - right?

Steve: Yes, that's it exactly. Of course ransomware gangs are not all the same. And didn't we hear - oh, no, it wasn't. It was Canon that had some information leaked last week that we reported on. And so Lawrence over at BleepingComputer has said that they assumed, since Canon got themselves back up relatively quickly, that they had paid the ransom. But now, since the extortionists in that instance were leaking the information, maybe Canon had restored from backups and said, nah, we're not paying your stinking ransom, and the bad guys said okay, here comes your private corporate next decade plans for the future. How do you want that? How do you feel about that being leaked?

Anyway, so in this case the University of Utah explained that it had dodged a major ransomware incident, and that the attackers managed to encrypt only 0.02% of the data stored on their servers, and the university staff was easily able to restore that from backups. However, the ransomware group then threatened to release student-related data...

Leo: Oh, see, they can't let that happen, yeah, yeah.

Steve: ...they had obtained and exfiltrated. So the university said: "After careful consideration, the university decided to work with its cyber insurance provider to pay a fee to the ransomware attacker. This was done as a proactive and preventive step to ensure information was not released on the Internet." And again, to the extent that such can be ensured. "The university's cyber insurance policy paid part of the ransom, and the university covered the remainder. No tuition, grant, donation, state, or taxpayer funds were used to pay the ransom." I thought that was an interesting explicit statement that they made.

And they said the university disclosed that the attack took place a little over a month ago, on July 19th, 2020; and that the network belonging to the College of Social and

Behavioral Science was the victim. So apparently a subset of the entire larger university was where the break-in occurred, and there must have been some isolation there. So anyway, that is one of the three. And presumably they were able to negotiate a cheaper payment because the bad guys hadn't managed to get the bulk of the university's stuff. But they did pay for the promise to not share student data. And as you said, Leo, the reason that would be honored is, well, they got nearly half a million dollars, and they want to be able to use that.

Leo: They want to do it again.

Steve: Yeah, exactly.

Leo: You've got to build your credibility.

Steve: Exactly. And two other larger and notable recent ransomware victims were Brown-Forman, famous for their distillation of Jack Daniels Tennessee Whiskey, and Carnival Cruises. The Jack Daniels folks said: "Our quick actions upon discovering the attack prevented our systems from being encrypted. Unfortunately" - again - "we believe some information, including employee data, was impacted. We are working closely with law enforcement, as well as world-class third-party data security experts, to mitigate and resolve this situation as soon as possible. There are no active negotiations."

So it sort of sounds like - oh, in fact that statement from Brown-Forman came after Bloomberg News reported that it had received an anonymous tip of the ransomware attack. A site on the dark web claiming to be run by members of the REvil strain of ransomware says that it had obtained a terabyte of data from the Louisville, Kentucky-based Brown-Forman. The site said that stolen data included contracts, financial statements, credit histories, and internal correspondence of employees. Also included were screenshots of file structures and documents purportedly taken during the heist.

So it does look like the pattern we're seeing now is, because major companies that have the deep pockets also have the pocket depth to now proactively back up their servers, well, so it's possible for them, if the only thing done was encryption, a golden opportunity to extract a ransom could be thwarted if the good guys have backups. And so now what's being done is the data pre-encryption is being exfiltrated and stored somewhere. Then the data is encrypted. And so we have, you know, we're increasingly seeing this two-part attack: exfiltration, that the company desperately does not want to be made public, in case they have backups, in which case they would not otherwise need to pay the extortion. So it's not really ransomware as much as it is, okay, we've got copies of all your stuff. Shall we share it with the world?

Leo: It's plain old blackmail.

Steve: Yup. And as for Carnival Corporation, the operator of the world's biggest cruise lines, they disclosed that they were hit by a ransomware attack that provided unauthorized access to personal data of passengers and employees.

Leo: That could be bad because that could be passport information, as well as address, name, birth date. I don't know if they collect socials. But, boy, that's a lot of information they have. And credit card numbers, of course.

Steve: Do they have passport information because they're - they're taking you across country borders.

Leo: Yeah. When you sign up for a cruise - and it's very rare that a cruise is just within one nation. If you're going to another country, they collect your passport. So they have screenshots of it.

Steve: You mean they physically, like...

Leo: Yeah, yeah. They hold it.

Steve: ...we'll hold it while you're with us?

Leo: Yeah.

Steve: Wow. Yup.

Leo: And I've been on many...

Steve: So the company has not yet identified - huh?

Leo: I've been on many of their cruise lines because they own most of them.

Steve: Yeah, well, you were doing the tech cruises.

Leo: Holland America was one of them, yeah. That was Holland America. Seabourn was a recent cruise of mine. And of course Carnival's a big one of its own right. They own a lot of the cruise lines.

Steve: Well, in fact you and Lisa like the whole cruise modality; right?

Leo: Oh, we love cruises. We won't be going on one anytime soon.

Steve: Them were the days.

Leo: Yeah. I look back on that with nostalgia and affection. The good old days.

Steve: Yeah, I was actually talking to my buddy Mark Thompson, whom you know. He's launched a little project to provide air quality monitoring in real-time to health clubs.

Leo: Oh, he should. He must. That's great. Good for him.

Steve: Yeah. And in fact I don't know how much of this I can talk about. So I'm not going to say anything more.

Leo: Just leave it at that.

Steve: I just realized. But we were talking about the air quality in cruise lines as opposed to airliners. And it turns out that the air is fully, completely exchanged on an airplane very often.

Leo: Right.

Steve: But not so on a cruise line.

Leo: Oh, interesting.

Steve: Which deliberately maintains a closed cycle system because the external air is often not what passengers want to be breathing.

Leo: It's kind of muggy.

Steve: No control over humidity. It's subtropical or whatever.

Leo: Yeah, it's very muggy. Although the lines I go on, the small ships I go on, we always have a balcony and windows we can open. And we always eat outside. So I'm not so worried about that. But I would imagine, some of those big ships, you're not breathing outside air ever.

Steve: No, no. And so it is internally recycled and not of the highest quality. So anyway, Carnival said that they had not yet identified which of their many subsidiary lines was breached. But because they are publicly traded, they did need to disclose to the U.S. Securities & Exchange Commission the nature of the attack in their regulatory filing.

They said: "Based on its preliminary assessment and on the information currently known (in particular, that the incident occurred in a portion of a brand's information technology systems), the Company does not believe the incident will have a material impact on its business, operations, or financial results. Nonetheless, we expect that the security event included unauthorized access to personal data of guests and employees, which may result in potential claims from guests, employees, shareholders, or regulatory agencies. Although we believe that no other information technology systems of the other Company's brands have been impacted by this incident based upon our investigation to date, there can be no assurance that other information technology systems of the Company's brands will not be adversely affected." So a CYA statement for the regulatory requirements.

Which brings me to an interesting set of reports that were just released. Certainly we all likely agree that a ransomware attack is the last thing any company wants, given what we keep seeing. So the question is, how exactly are these occurring? The traditional answer has been phishing email, which hooks some well-meaning but unsuspecting insider. Well, while phishing email is indeed a popular entry point vector, these three recent reports from Coveware, Emsisoft - I keep tripping over that - Emsisoft, and Recorded Future clearly show that phishing actually takes a backseat to our old friend RDP.

And I have a chart on page 2 of the notes which is really interesting. This is Ransomware Attack Vectors over time. And it would be a pie chart except a pie chart can't show percentage change over time. So this is a line chart from fourth quarter of 2018 through the second quarter of 2020, so up to current, where it's showing the percent of cases of RDP compromise, email phishing, software vulnerability, or other. So it's like a pie chart varying over time. So consequently, for example, in the fourth quarter of 2018, by far and away the majority of the attacks, looks like maybe, just eyeballing it, maybe 85% were RDP. That came down as a percentage as...

Leo: This can't be right. This makes no - is RDP that big a problem?

Steve: Yes, it is, yes.

Leo: Holy cow.

Steve: And these guys provide the raw data to back that up.

Leo: And it's Windows RDP; right? It's Microsoft Windows.

Steve: Yes. Windows is 100% RDP. And so email phishing did come up as a percentage, which pushed the percentage of RDP down. But it's holding its own. And then of course what happened now in 2020, well, in the first and second quarters of 2020, is due to the COVID and the dramatic increase of hastily brought up RDP services in order to allow remote access, that essentially began fighting with email phishing as an entry point. So those are the two.

But email phishing never even reached parity with RDP. It's gone up and down, but RDP is holding its own, which I think is one of the things that I'm going to spend some time talking about here. In their report, Emsisoft explained what's happened this year. They said: "In recent months, organizations across every sector have come to rely heavily on Remote Desktop Protocol (RDP) to maintain business continuity while respecting social distancing."

And I'll back up a little bit, Leo, just to address your comment. Remember that there have been a series of really bad authentication problems with RDP. This is why I've been saying you just can't...

Leo: Well, I know, but who the hell uses RDP? Don't they use VPNs and other solutions? Who's using RDP? That's nuts.

Steve: No. No. There are 80,000, eight zero thousand, exposed RDP services on the Internet. It is absolutely crazy. But they're just assuming that, oh, yeah, you know, it must be secure because Microsoft says we can turn it on. Well, Microsoft said that once about Windows printer and file sharing, and we know how that went.

Leo: See, but Barracuda in their ad says 91% of all ransomware attacks come through phishing email, spear phishing emails. And every one I've heard about is a spear phishing email. I can't imagine Canon or Carnival or any of these people as using RDP. That's crazy.

Steve: Well, in terms of number I'm sure it's the smaller guys that aren't deploying the technology that they need to. Anyway, so Emsisoft said: "However, the rapid shift to remote working has also provided a unique opportunity for ransomware groups. Threat actors predicted that many organizations would not have the time or resources to securely implement RDP during the mass transition to working from home and, as a result, may be vulnerable to compromise. They were right. According to a McAfee report, the number of Internet-exposed RDP ports grew from approximately" - oh, boy, I low-balled it - from approximately, are you sitting down, Leo? - "3 million in January of 2020 to more than 4.5 million in March." So in less than three months, an additional 1.5 million additional RDP ports became publicly exposed.

Later in their report they note that, while the threat is not new, and of course as we know all too well on this podcast, the global shift to remote working has revealed that many organizations do not adequately secure RDP, and that the bad guys are taking advantage. According to a report by Kaspersky, at the start of March 2020 there were about 200,000 daily brute-force RDP attacks in the U.S. But by mid-April, just six weeks later, that number had grown from 200,000 to nearly 1.3 million brute-force attacks per day.

Now, today, RDP is regarded as the single biggest attack vector for ransomware. So all of these 4.5 million RDP ports are publicly exposed, and they are being actively attacked. And what is to me shocking is that, even now, Microsoft has not stepped up and offered better security. And so of course this obviously underscores a point I've been making, which is that RDP simply cannot be safely exposed to the public Internet. And there are two reasons. It's no longer sane to trust that Microsoft hasn't or won't make a mistake in their provision of the RDP service. They've done so over and over in the past. And as we'll be learning in a few minutes, they just released an emergency patch for two more privilege elevation flaws in Windows remote access service, which is what RDP is part of.

The second reason we cannot trust RDP is that its native authentication mechanisms are pathetic. I went looking for something that I thought I might not know about RDP authentication, thinking Microsoft must have fixed this. And I found a very recent, it was only a couple months old, best practices advisory from Microsoft on securing RDP authentication. It amounted to "Be sure to use a strong password." There is zero multifactor support for RDP, which is unconscionable.

There are multiple third parties who have responded to this need created by Microsoft by creating their own, much more secure RDP gateways, which are laden with authentication features. So if you don't feel like rolling your own solution, and you do have money to burn because none of these third-party solutions are inexpensive, you could simply throw some money at the problem and buy yourself this much-needed security. Or you could get a bit clever and roll your own.

If the clients you have connecting have fixed IPs, restricting access to the RDP port from only those IPs is an immediate, proven, fast, and secure solution. If the IPs are largely

fixed, as with the typical residential broadband service, the use of a DynDNS solution can allow for tracking their infrequent but possible changes. And typical changes may be so infrequent that updating the access firewall from DynDNS doesn't even need to be automated. I know that the IPs that I maintain in my two locations, they're essentially static. They haven't changed, like, in years. But if highly dynamic roaming access is needed, then no form of IP-based access restriction will suffice. This lifts the requirement for authentication from the network layer to the application layer.

Again, history teaches that what we must avoid is public access to an RDP endpoint. There's no safe way to protect it. We can't trust Microsoft, and we can't allow for this brute forcing of our authentication. You know, 1.3 million attacks per day is ongoing right now. So that requires the use of something in front of the RDP endpoint. I've talked about using a VPN offering and where the VPN offers some form of strong multifactor authentication, either a time-based one-time password or certificate-based.

But I wanted to add another option to the pot by noting that SSH is widely available, almost always offers the very strong authentication options that we're looking for, and it can be used to tunnel RDP. In fact, if you google "tunnel rdp over ssh," you'll be rewarded with all the suggestions you might need, and for all OS platforms. As I was thinking about this, the only theoretical downside is that, as a purist, both RDP and SSH use TCP. Long ago, when we were first covering the operation of VPNs in this podcast's deep history, I noted that there can be some tunnel confusion when TCP is tunneled inside TCP, since then you have two sets of TCP's error recovery and packet retransmission.

The theoretical optimal solution is for the VPN tunnel to use dumb old UDP packets for the tunnel protocol, and then RDP over TCP, which is carried by the UDP tunnel. So that way the RDP's TCP protocol handles any packet losses and retransmission, and those are carried over UDP. I did find some SSH UDP tunneling systems. But there was so much apparent success with simply tunneling RDP over standard SSH TCP tunnels that it appears my theoretical concerns might be nothing more than that, just theoretical. So anyway, I wanted to propose another option to the need for somehow hiding RDP endpoints from the outside world. It's clearly necessary to add some other layer of security in front of RDP.

Leo: Somebody in our chat whose company uses it says they use a proxy server in front of it, so it's not a publicly available IP address.

Steve: Right.

Leo: Yeah, which makes a lot of sense.

Steve: You just have to hide it.

Leo: Yeah.

Steve: And I'll mention one more Microsoft issue, and then we'll take our second break. Last Thursday, Microsoft issued an emergency out-of-cycle update for Windows 8.1, 8.1 RT, and Windows Server, the matching server instance of 8.1, which was Windows Server 2012 R2. The emergency update patches a pair of recently disclosed security

vulnerabilities. I have a link in the show notes because from my reading of this it doesn't look like this is going to be an automatic update. Maybe they'll roll them out next month.

What's interesting is they're both high-severity privilege escalation vulnerabilities residing in the Remote Access Service, which is obviously a particularly vulnerable area of a server. Interestingly, both of these vulnerabilities were patched as part of the previous week's August Patch Tuesday, but that was for Windows 10, Windows 7 for those on extended support, and Windows Server 2008, 2012, 2016, 2019, and Windows Server versions 1903, 1909, and 2004 systems. In other words, everything other than Windows 8.1 and its corresponding Server 2012 R2.

So as near as I can determine, it's just that these patches for those two operating systems just weren't ready in time to make whatever quality control, if any, Microsoft is applying to the monthly patch cycle. But at the same time they were too critical for Microsoft to leave them hanging since somebody examining the patches, and we now know that happens, somebody examining the patches for the other OSes could probably figure out what it was that was fixed and note that that had not been fixed in Windows Server 2012 R2 and then perhaps go attack it by reverse-engineering what had been fixed in all the other platforms. So as I said, it may be that they need to be manually patched and installed. It wasn't clear.

And assuming that they'll be part of next month's patch batch, I would say that end users probably don't need to worry because Windows 8.1 probably doesn't have any remote access services publicly exposed in the typical end user environment behind a NAT router. So unless you're actually running Windows Server 2012 R2, you probably don't need to worry. But if you are, I would certainly think it worth going to get that, or making sure that it isn't already updated by Microsoft.

Leo: This Coveware article, I think, tells you why it's so big. Because it's one particular kind of ransomware, Phobos, that focuses on RDP, and it's ransomware as a service. So any idiot can go, and all the credentials are available online for pennies. So any idiot can go either to a Shodan or get some credentials. And then you don't have to know what you're doing. Whereas any spear phishing attack, anything more sophisticated is going to take a lot more effort. So this is low-hanging fruit. You probably don't make a lot of money with it, you know, these are the $100, $200, $300 ransomwares. They're not getting a half million.

Steve: Yes. And in fact what you have just said is exactly where we're headed after our second break. And I even used the phrase "low-hanging fruit."

Leo: Yeah, yeah.

Steve: Yup.

Leo: Now I understand. Because it's counterintuitive, but of course it's not because lots of people who don't know what they're doing are putting RDP out in the public, which is so dumb.

Steve: Exactly, exactly.

Leo: So dumb. And there are so many better solutions out there. But, you know, it comes with Windows, you know, it's available. Why not?

Steve: Yeah, turn it on.

Leo: Yeah. Wow. And then they're using monkey123 as the password. I mean, it's got to be the same people; right?

Steve: Shhh, don't tell anybody, Leo. That's mine.

Leo: Secret. Of course you're getting bit.

Steve: How did you guess it on the first try?

Leo: It's probably not very lucrative.

Steve: Wow, you're good.

Leo: You know, if you really want the money, you're going to go out and find a company and target them and take some time. And then that's when you can exfiltrate data, do all those fun things. Make a lot more money, I would imagine. Wow. Wow. I can't believe that in this day and age.

Steve: So speaking of low-hanging fruit.

Leo: Yes.

Steve: Iranian script kiddies are using RDP to deploy the Dharma ransomware. This was some interesting and disturbing research by a group known as Group-IB. They detailed the collision of RDP and ransomware. They explain that apparently, like from all the forensic evidence, these look like low-skilled hackers - I'll explain why in a second - likely from Iran, have joined the ransomware business, targeting companies in Russia, India, China, and Japan. They're going after the new low-hanging fruit represented by casually or hastily deployed RDP servers using publicly available tools.

The group is deploying the Dharma ransomware. And based on the forensic artifacts of the attacks, it appears to be a non-sophisticated, purely financially - as opposed to for example politically or state-level - motivated group, which is new to cybercrime. Their extortion demands range - they're pretty modest - from one to five bitcoin, which puts it at around, what, $11,700 up to maybe $60,000. And they locate targets the old-fashioned way, by scanning IP address ranges for exposed remote desktop protocol, RDP endpoints.

Their tool of choice is a freely available open source port scanner called Masscan we've talked about before. Once they've located a potential target, which is to say they found port 3389 open, they launch a brute force authentication attack using another tool,

NLBrute, which is a utility that simply repeatedly attempts to authenticate against RDP using a list of username and passwords, attempting to find a combination that works. If they get in, they sometimes attempt to elevate their privileges by exploiting an old vulnerability which exists in Windows 7 through 10.

And researchers at this company Group-IB learned about the new group a couple of months ago, in June, during an incident response engagement at a company in Russia that had been attacked. Based on that forensic analysis and the artifacts from that, they determined the attacker to probably be a "Persian-speaking newbie." The conclusion is supported by clues from the next stages that they found of the attack, which appear to lack the confidence that you would expect from an actor who knows essentially what to do once they've gotten in. It's like, oh, we got in. Now what?

Leo: What do you recommend now?

Steve: Group-IB wrote: "Interestingly, the threat actors likely don't have a clear plan for what to do with the compromised networks. Once they've established the RDP connection, they decide which tools to deploy to move laterally. For instance, to disable built-in AV software, the attackers used Defender Control and Your Uninstaller." Which are tools available, sort of generic, in order to get done what they want. Nothing very sophisticated there. Further evidence that the operation is the work of a script kiddie from Iran comes from search queries in Persian to find other tools necessary for the attack. It's like, uh, okay, let's see, what should we search for now to do something?

Leo: It's probably a 12 year old.

Steve: Yeah. Anyway, those searches were turned up in Persian-language Telegram channels which provide those tools.

Leo: Oh, yeah. Of course, yeah.

Steve: The number of victims compromised so far by this attacker is not known, nor is the path that led the threat actor to the Dharma ransomware as a service operation (RaaS). But given that the Dharma operators provide a toolkit that makes it easy for anyone to become a cybercriminal, it should not come as a surprise that inexperienced individuals are deploying this file-encrypting malware. It's like, oh, oh yeah, wait, now we're supposed to launch the Dharma, once they get in.

So the senior analyst at Group-IB, a guy named Oleg Skulkin, said that the Dharma ransomware source code, which was leaked in March, likely explained the increasing use of this malware strain. Oleg indicated that: "It's surprising that Dharma landed in the hands of Iranian script kiddies, who are using it for financial gain, as Iran has traditionally been a land of state-sponsored attacks engaged in espionage and sabotage." So in other words, maybe it's not that surprising because it's sort of now available for everyone, not just state-level actors.

And of course we've talked about how this new ransomware as a service model is allowing many hackers who would never be in the ransomware game to become players. And given that we have ransomware as a service now, I don't see how this problem is ever going to go away. So again, no exposed open RDP ports; okay? Not for our listeners. Arrange to put something, anything, in front of RDP so that you or your

company are not open to exploitation. This is not as bad as Microsoft's original wide-open Windows file and printer sharing, which is what drove me to create GRC's ShieldsUp! service so many years ago. But it does definitely need attention. No exposed RDP ports.

The Zero Day Initiative (ZDI) that we've also referred to recently turned 15, as I mentioned at the top of the show. It turns out that there's a bit of a synchronized 15th birthday since last Thursday Pwn2Own's founding parent, the Zero Day Initiative, also turned 15, just as this podcast did. Our first podcast, as I mentioned, was August 15th, 2005, one day before the founding of the ZDI program.

Leo: That's interesting.

Steve: Yeah, one day. On last week's occasion of their 15th birthday, ZDI announced that more than $25 million in bounties had been paid to security researchers over the past decade and a half. Those monies went to more than 10,000 security researchers across more than 7,500 successful bug submissions. In explaining the genesis of ZDI they said: "Starting in 2005, 3Com" - remember them? - "3Com announced a new program called the Zero Day Initiative. The plan was to financially reward researchers who discover previously unknown software vulnerabilities and disclose them responsibly. The information about the vulnerability would be used to provide early protection to customers through TippingPoint's IPS (Intrusion Prevention System) filters, while the Zero Day Initiative then worked with the affected product's vendor to fix the vulnerability."

So that's an interesting angle here. The commercial TippingPoint IPS would benefit from providing immediate awareness of a vulnerability and could offer their proprietary customers, their commercial customers, unique early protection, thanks to the IPSes being immediately updated before any fix was available from the vendor. Which as we know could take, like, 90 days or more. Then of course the vendor would then fix the problem downstream of the IPS eventually. But even after the vendor's problem was fixed, there's certainly some value in knowing when attacks are being launched against one's IPS protection, even when there's no backend vulnerability any longer.

So anyway, that's how this happened is that 3Com said let's get in the business of collecting this information from hackers. That will allow us to mature our intrusion protection system in advance of any vendor's vulnerabilities being fixed. We offer our protected customers this window of safety for their own backend systems, and we're going to turn this into a commercial venture.

So they ended up saying that first year ZDI published a total of one advisory, pertaining to Symantec's Veritas NetBackup. "Fifteen years later," they said, "we've now published" - as I said at the top - "1,500 advisories as we evolved into the world's largest vendor-agnostic bug bounty program. To say it's been a journey is an understatement. It's certainly had some ups and downs, but the program is stronger than ever and on track for our largest year ever. As we begin our 16th year" - as they did last week, as we did last week - "let's take a look at some of the more notable happenings in the life of the ZDI program."

So I read through the entire posting, and it provided such a useful and synchronized perspective and walk through the 15 years of this podcast, I decided I wanted to share it with our listeners. So here's what they said. Through the years of 2005 through 2010, their first five years, they wrote: "Looking back at our activities through these years induces nostalgia as it reminds us of the bugs we bought in products and companies that are no longer with us. We can also see the rise of research into different products and technologies. For example, we bought only two Apple bugs in 2006. That number rose to

52 by 2010. Java bugs, particularly sandbox escapes, were also popular during this time." And of course we were talking about them on this podcast all the time.

They wrote: "It's a bit odd to look back at the progression from buying bugs in what was simply known as 'Java,' to buying bugs in 'Sun Microsystems Java,' to buying bugs in 'Oracle's Java.' This time period also saw the first Pwn2Own contest, which was in 2007. The contest launched at a time when 'I'm a Mac; I'm a PC' commercials dominated the airwaves..."

Leo: That puts it into context.

Steve: Remember that? Yeah.

Leo: Yeah, long time ago, yeah.

Steve: Yeah, "...and Apple devices had an aura of invincibility about them. Astute security researchers knew better, and Dino Dai Zovi proved it, winning himself a MacBook and $10,000. The contest has grown exponentially since then. There are now three different competitions: Pwn2Own Vancouver" - the main one that we often, well, we always cover - "which focuses on enterprise software; Pwn2Own Tokyo, which focuses on consumer devices; and Pwn2Own Miami, introduced this year with a focus on SCADA products. Pwn2Own also served as a 'coming out' for many high-profile researchers who, after winning the contest, went on to work on various prestigious teams and projects."

So from 2010 to 2015, their second five-year block, they said: "This was a transitional period for the program as 3Com, together with ZDI, was purchased by Hewlett-Packard, then later split off as part of HP Enterprise. However, the core principles upon which the program was founded remain the core principles we operate by today," four of them: "Encourage the responsible disclosure of zero-day vulnerabilities to the affected vendors. Fairly credit and compensate the participating researchers, including yearly bonuses for researchers who are especially productive within the program. Hold product vendors accountable by setting a reasonable deadline for remediating reported vulnerabilities."

And remember we talked about a six-month ZDI, the patience that ZDI had for six months and then finally disclosed it publicly. That was one of Microsoft's, one of those two zero-days that Microsoft didn't fix until ZDI disclosed it, and then they thought, uh-oh, because they just dragged their heels. And, finally: "Protect our customers and the larger ecosystem."

So they said: "By this time the ZDI was large enough to have an impact on the overall ecosystem. It was during this period that we grew to become the world's largest vendor-agnostic bug bounty program, a title we still hold. In 2011 we had our first public zero-day disclosure when a vendor failed to meet the patch deadline. Over the years, holding vendors accountable has helped lower their response time from more than 180 days to less than 120. Even though we reduced our disclosure window, the rate of zero-day disclosure stayed relatively consistent.

"Another big change during this period was the increase in research work done by the vulnerability researchers employed by the ZDI program. There have always been great people working on the program doing root cause analysis on submissions, but an increase in the size of the team allowed for members of ZDI to begin reporting their own bugs, as well. ZDI researchers increasingly published their findings and expanded their speaking at high-profile conferences including Black Hat and Defcon.

"The increased size also helped spot some trends in exploitation. It was during this time that we saw a surge in submissions of Java bugs. However, once browsers implemented click-to-play, practical exploitation became more difficult. Bugs exploiting use-after-free conditions in IE were also quite common until the Isolated Heap and MemGC mitigation were silently introduced by Microsoft. ZDI researchers found a way to exploit the mitigations and were awarded $125,000 from Microsoft for their submission. Interestingly, Microsoft chose not to fix all the submitted bugs, so a portion of the report ended up as a public-release zero-day." And they said: "In case you're wondering, all of the money was donated to various STEM charities.

"During this timeframe, the bug bounty landscape became normalized and broadened. Vendors such as Microsoft and Google started their own bounty programs. And bug bounty programs were created that allowed companies like Starbucks and Uber to offer bounties." And as we know, by "bug bounty programs were created," what they mean, without naming them, of course, is HackerOne, which we have spoken of often and just recently.

They wrote: "The idea of crowdsourcing research entered the mainstream. Not every program was successful, as some vendors suddenly realized that, if you offer money for bug reports, you get bug reports. This left some companies scrambling to react after starting their program with mixed results. It was definitely a time of growth and learning throughout the industry.

"Pwn2Own continued to grow, as well. 2010 saw Pwn2Own's first successful mobile device exploit, demonstrated by Ralf-Philipp Weinmann and Vincenzo Iozzo against the Apple iPhone 3GS. We also started seeing vendors release large patches just before the contest. Since the rules require the 'latest version' for all exploits, contestants" - Pwn2Own contestants - "often found themselves 'patched out' just before the contest. It also meant the ZDI had to scramble to get the targets up to date with all of the latest patches, often staying up all night installing updates. In 2012, a second contest, Mobile Pwn2Own, was added to focus on phones and tablets."

And finally, the final five years, 2015 to present: "In 2015, Trend Micro acquired the HP TippingPoint IPS and the ZDI program along with it. This opened a new world of opportunity for ZDI, as the vulnerability intelligence produced by the ZDI program could now be used to improve not only the TippingPoint IPS, but other products within Trend Micro's line of security solutions, as well. ZDI's association with Trend Micro also resulted in a massive increase in interest in vulnerabilities in Trend Micro products themselves. To their credit, Trend Micro product teams have not shied away from the work of fixing the bugs submitted by independent ZDI researchers, and we have established a Targeted Initiative Program just for select Trend products.

"The threat landscape shifted, as well. Before 2015, we rarely saw an Adobe Reader submission outside of Pwn2Own. Once we reached 2015, there were more than 100 submissions. Many of those reports were submitted by ZDI researchers. Overall, internal finds represent about 20% of all the cases we process every year. Bugs affecting Acrobat, Foxit, and other PDF readers continue to be prevalent. But we've also seen the rise of deserialization bugs and a sharp increase in SCADA vulnerabilities. Home routers have also become a popular target since they can be compromised en masse to be used in botnets and DDoS attacks. As a result, the ZDI adapted and began accepting hardware-related submissions, especially those related to IoT devices.

"The introduction of the Wassenaar Arrangement posed some challenges, especially when purchasing bug reports from member countries. However, we were able to navigate the paperwork needed to transfer 'cyber arms' and stay on the right side of the law. The virtualization category was introduced to Pwn2Own in 2016, and since that time we've had several guest-to-host escapes demonstrated." Of course we've talked about those on

the podcast. "The contest celebrated its 10th anniversary in 2017 by acquiring 51 zero-day vulnerabilities over the three-day contest. In 2019 we partnered with Tesla to award a Model 3 to a pair of researchers who exploited the car's infotainment system. ZDI researchers also demonstrated their own exploit of the infotainment system.

"The contestants have changed over the years, as well. In the beginning, individual researchers made up the majority of entries with only a few teams participating. At one point, this shifted to most participants being teams sponsored by their employers. There have been instances of teams filing bug reports with vendors before the contest in the hopes of killing their competitors' exploits. In the past couple of years, that has shifted back towards individuals and small independent teams.

"And we've never stopped growing. We hit our peak of 1,450 published advisories in 2018, and we're set to eclipse that this year. In fact, we've been recognized as the world's leading vulnerability research organization for the past 13 years. According to Omdia, the ZDI was responsible for over half of all measured vulnerability disclosures in 2019, more than any other vendor."

And finally, moving forward, they said: "Over the past 15 years we've seen trends in the exploit economy and vulnerability marketplace come and go. But through it all, we've been laser-focused on one thing: making the digital world more secure, one CVE at a time. Through the tireless work of ZDI researchers and the wider community, we're determined to continue disrupting the vast cybercrime economy and raising the bar for enterprise software security for the next 15 years and beyond."

So anyway, interesting walk through the past 15 years, which pretty much corresponds with the podcast. And we've covered all this stuff along the way.

Leo: Completely parallel, yeah.

Steve: Very cool. So a couple of bits of miscellany. I mentioned that I had finally found what I consider to be a useful bloatware remover for Windows 10. I actually knew about it before, and I had forgotten about it. And I was reminded of it by a tweet for something else this company does. And I thought, oh, yeah, I remember O&O. Anyway, they're O ampersand O, O&O. And the one I like is O&O AppBuster. If you just google "O&O AppBuster," you'll find it, from O&O Software. It's free. They have various commercial offerings. So I think this is sort of a bit of a loss leader for them. But of all the things I've tried, I like this one the best. It is comprehensive. If you enable the display of hidden things, which you probably should not, then you are able to really get yourself in trouble.

But anyway, I just wanted to point our listeners at O&O AppBuster. It's a nice utility. You don't need to install it. It just runs. It's standalone. So you can just drop it on your desktop. If you make changes, it will create a little companion file outside of itself where it stores those changes. But that also allows you to move them around. Anyway, I like it a lot. And it's what I've been using, and I will continue to use when I want to decrapify a new installation of Windows 10 with all this ridiculous animated tiles and candy cane crap. I just, again, I can't believe what's been done to Windows.

On a serious note, I've been using something now for about a year that I can honestly say I have fallen in love with. It is a modest program called Remote Utilities for Windows. It is paid. It's a commercial app. It's a remote control app. I've looked around at them all and went through a period about a year ago of trying them all. In my case, the need was that Lorrie, my significant other, whom you've all heard me mention from time to time, we wanted to set her up with the ability to do remote neurofeedback. What that meant was that she would send out a laptop, an EEG amplifier, and the required EEG electrodes.

In that laptop would be a bunch of software which would provide real-time feedback about some aspects of her client's brain functioning.

And it works like regular, you know, any sort of feedback where it exposes something that your brain is doing, and you learn to push it in the direction you're supposed to, and you're able to thus modify the function of your brain. It works. She's had a whole bunch of interesting and really heartwarming successes, especially with kids. But the point is she needed remote access to these laptops. She needed to be able to work with the person remotely, change settings, basically remote control. I looked at everything. And this is what we've been using for the last year, and it is such a win, I just wanted to put it on our listeners' radar.

As I said, it's commercial. It is not a subscription. Being an old fart myself, I would not consider it if you had to pay by the month. And everything is turning into a subscription model, which just irks me. So they have a number of different licenses to suit enterprise needs. If you have modest needs, they have a free license that will allow you to put the connection settings for up to 10 remote machines in your viewer's address book. I purchased a license because - and in fact Greg, my own tech support guy, has a little business on the side helping his clients. He completely fell in love with it and has switched to using it. Lorrie has, as I said, been using it for a year.

Leo: How much?

Steve: It's not expensive. You can find - they have, like, various pricing plans. Again, you can use, for free, you can put 10 remote computers under control. The host app is what goes on the remote machine. What they call the "viewer" is used by you, the tech, in order to get access to the remote machines. You can use their infrastructure in order to knit the machines together. But if that makes you feel uncomfortable, you can also use a self-hosted server which you put somewhere, which allows these things to connect to each other through NAT.

So this all does NAT routing and rendezvous services. There's also an agent which can be used for spontaneous access to a remote system without installation. You would just, if you immediately needed to get access to a remote system, you would just send this agent to that person, who would run it on their machine, and then you would have access to it with the proper security.

PCWorld in their review of Remote Utilities wrote: "For power users, there's plenty to like about Remote Utilities. Several connection modes are offered beyond the full remote desktop experience. There's also file transfer mode, remote device manager, a registry viewer, remote webcam access, and a terminal mode which is an excellent way to perform simple command line tasks remotely."

There's an MSI Configurator to create custom Host installers for unattended access or to customize the remote Agent module where you could put your own company logo and welcome text for attended support. It supports Power Control Mode, allowing you to remotely restart a PC, either in normal or safe mode, shut it down, lock it, put it to sleep. Active Directory support, you can fetch an Active Directory tree, add new domain controllers, and access Active Directory workstations and servers with one click. It's got two-factor authentication, time-based token, for access to specific or every remote host. I mean, it just goes on and on.

All the connections are TLS 1.2 that cannot be turned off. Encryption is always on. You can encrypt your address book in case the viewer's workstation was ever compromised to keep a bad guy from using that to get into the remote systems that you have access to.

Host identity is certificate-based to ensure that you are connecting to the same host that you intend to and not some sort of spoof. You can deploy it in a totally isolated environment over a LAN where you have direct connection, or over the public Internet, which is how, for example, Lorrie and I use it. Blank passwords are not allowed. There are no default passwords, either.

I mean, these guys clearly understand security. They did everything right. It's got built-in protection against brute force cracking. When an excessive number of incorrect password attempts is seen, the system automatically begins increasing the amount of time required and will lock out an IP that is failing at multiple requests. And of course there is no ability to brute force because most systems are behind NAT so there's no open ports, either. Anyway, it goes on and on. It's just called Remote Utilities. I am in love with it, and I wanted to make sure our listeners knew about it.

The other thing I wanted to make sure that I sort of reminded people of. I'm still in love with Sync.com. But it's limited to synchronizing a single folder tree among two or more Windows devices. It's perfect for what it does. I'm using it. I'm loving it. For that it's great. But in another aspect of what we needed for Lorrie's deployment, we wanted a little directory tree that's underneath each of these neurofeedback apps that are out in each of these deployed laptops, and there's 20 of them out at the moment. We wanted a little snippet of a directory tree to get synchronized back to the Drobo that we have in our location.

And Syncthing, again, I've mentioned it before. I wanted just to remind people of it. I was asked by somebody in Twitter who had a couple of QNAP servers how he and his buddy could synchronize them to back each other's stuff up. And I was reminded of Syncthing, which just does that perfectly, that you can run Syncthing on QNAP. I'm running it on my Drobos. It runs on Mac and Windows and Linux, FreeBSD, Solaris, and OpenBSD. It's open everything. Protocol, open source, open committee, I mean, it's a great tool. And it supports a far more flexible, like you can get yourself lost, you can create something so complex.

So I have all of the machines that are out there synchronizing a snippet of their directory structure back to a compound directory structure on the Drobo which Lorrie is able to see. Some of the folders are one-way synchronizing so that, for example, logs that are external synchronize back to us. Some of them are one way in the other direction so that, if Lorrie drops a media file that she wants to go out to everybody, she just drops it in the media folder on the Drobo, and the next time everybody connects, they get updated automatically. And then some things are bidirectional synchronized so that the most current copy is synchronized.

Anyway, Syncthing.net is that. And there's an investment you need because it's kind of funky the way it works. It took a while for me to - I was going "Huh?" for a while. But once you understand the way it works, there's just nothing it can't do in terms of options and features and the complexity that you're able to maintain. And again, it's all NAT-penetrating. You can open a port if you want. You can allow it to use UPnP if you want. Or you can allow it to use external relay servers, and it'll do that, as well.

Leo: As you can see, I've been a very happy user for the last five or six months.

Steve: I didn't know that. Very, very cool.

Leo: Oh, yeah. I love Syncthing because you don't need a third-party cloud storage. If you have enough devices...

Steve: Exactly.

Leo: ...just sync them all. In fact, after you mentioned Sync.com, I kind of was looking for other things. I tried other third-party stuff. Then I found Syncthing. I said, oh, that's exactly what Steve needs. I've been meaning to mention this to you for some time now. But it's really great. All my systems are on it, which is nice if you have multiple systems because then it keeps stuff in sync. I just did a hands-on Macintosh piece on it this past Saturday, ironically.

Steve: Ah, very cool.

Leo: So, you know, I've been meaning to ask you because I really like this idea that each device has its unique identifier, plus each folder has its unique identifier. And even though I just showed both of those, it's secure because if you entered into your Syncthing, oh, well, I want to join Leo's W6MUCOA9UF folder, I'd still have to give you permission to do it. So it's secure in that regard.

Steve: Correct. What happens is, if somebody were to grab that and drop it into their Syncthing and try to create a connection, a request on your end would pop up saying, hey, somebody got your ID. Do you want to allow it? But that's also the cool thing is that...

Leo: You can share with anybody, yes, yeah.

Steve: ...you could easily create a directory and share it with a friend. They use that, and then you link those two things together. It's just, I mean, they just nailed it.

Leo: It reminds me of our old friend BitTorrent Sync except it's done right. It's really - I think it's just exactly. And it's open source, so we don't have to guess what the protocols are.

Steve: It's open source. Everything is certificate-based. That's basically a large fingerprint of the certificate that uniquely identifies that machine. And you're able to do things like say, if a new subfolder appears, or if a machine I'm connected to creates a new folder, I want to automatically grab it and start syncing it, or I don't. And, I mean, it just...

Leo: Oh, on and on.

Steve: Again, there are so many features that you can get yourself a little tangled up. But for a power user, they nailed it. And massively cross-platform. It runs on everything.

Leo: That's why I use it. It's on my Linux machines. It's a daemon that runs in the background on all of my machines - Linux, Mac, and Windows. I love the file versioning. You have lots of choices of file versioning. And I often will do it "send

only" so I don't have to worry about syncing deletions. If you make it "send only," then that's basically a backup. Everything you change here will be synchronized, but nobody else's changes will be synchronized. So, you know, it does take a little time to kind of figure it all out. But I'm supremely impressed with it. I'm glad you agree. I've been meaning to ask you about it. Good, good, good, good. And it's free.

Steve: Yes, and free, exactly.

Leo: Free.

Steve: Thank you very much, world.

Leo: World.

Steve: So over the weekend, using DigiCert's entirely automated self-service system, I rekeyed all of GRC's certificates ahead of the next Tuesday, September 1st deadline, after which certs can only have a 397-day life, rather than twice that. And among those that I rekeyed was GRC's revoked.grc.com cert, which I mentioned last week had expired so that it was being dishonored due to expiration rather than revocation. And for those listeners who had been using the service, and I learned that there were, what is it, 580-some a day are going to the revoked.grc.com page, that system is up and running again. So I bought myself an extra year before I need to do all of that again.

But I also realized a benefit of having the expiration date of all of my certs now synchronized. If we're going to be needing to do this annually, as I will starting two years from now because from now on certificates are going to be expiring annually, doing that certificate renewal work in a single batch will at least be much more convenient than the interrupted multiple times per year.

Leo: You have a new holiday on your calendar, Cert Renewal Day, and you just do it every year, yeah.

Steve: Now, of course, it is the case that it's only necessary because I'm using OV certs, you know, Organization Validation, as a class above the DV certs, the Domain Validation, which can be, and I recognize this, fully automated. And a lot of people are just going to say, okay, Gibson, I don't care, I'm just going to use the ACME protocol with Let's Encrypt and let my server keep itself updated. And I get it. Maybe someday that'll happen. But for now, DigiCert has made this so simple for me that it's something I enjoy, and all my certs are now synchronized. And I just wanted to mention that revoked.grc.com is back up and running again.

Oh, and one last bit. I meant to mention last week, Leo, and I know you'll appreciate this, that as a result of all the benchmarking R&D we've done, we have learned a great deal about SSDs operating in the real world - Samsung, Kingston, OCZ-Vertex, Crucial, and others. And what has surprised us all is the non-uniformity that many of them show in their operation. It's not what anyone would expect from something with the seeming purity of solid-state memory. Their operation was kind of all over the map and varied widely at the five different points where we are benchmarking them.

The thing I meant to note was that one single brand stood out from all others. Samsung was by far the most rock solid. And every one of those that we saw, and there were many of them represented in the population that have been looked at so far, they all followed the governing specifications to the letter, which many did not. But also the performance was just solid.

Leo: Yeah. There's my favorite, the EVOs, yeah. Really love those, yeah.

Steve: I just wanted to say that I'm sure that the gang who are working with me in the spinrite.dev group all will see their future purchases biased, all other things being equal, toward Samsung because we all were going, wow. Because, I mean, we're all sharing all of our results. And it's like, whoa, okay. So we already know that smartphone cameras now have sufficient resolution, and our software's become sufficiently clever, that a photo of a traditional house key at a distance can be used to reconstruct a working physical key. And we also know that the vibrations of objects in a distant room - we've talked about balloons, a bag of potato chips, a light bulb, or even the leaves of a plant - can be observed optically by laser or similar technology at a distance to reconstruct the acoustic waves those objects are being subjected to, to eavesdrop on conversations occurring in that room.

And now, with the publication of some intriguing new research, another piece of our traditional perception and assumption of security has just fallen to the wayside. The research paper, which documents the detailed and painstaking work by three quite enterprising students in the Department of Computer Science at the National University of Singapore, bears the title "Listen to Your Key: Towards Acoustics-based Physical Key Inference."

Leo: Oh, oh, oh, no, no, no. No. Oh, my goodness.

Steve: Yes.

Leo: Okay.

Steve: The abstract of the paper reads: "Physical locks are one of the most prevalent mechanisms for securing objects such as doors. While many of these locks are vulnerable to lock picking, they are still widely used as lock picking requires specific training with tailored instruments, and easily raises suspicion. In this paper we propose SpiKey, a novel attack that significantly lowers the bar for an attacker, as opposed to the lock-picking attack, by requiring only the use of a smartphone microphone to infer the shape of the victim's key, namely bittings, or cut depths, which form the secret of a key.

When a victim inserts his or her key into the lock, the emitted sound is captured by the attacker's microphone. SpiKey leverages the time difference between audible clicks to ultimately infer the bitting information, i.e., the shape of the physical key. As a proof of concept we provide a simulation, based on real-world recordings, and demonstrate a significant reduction in search space from a pool of more than 330,000 keys to three candidate keys for the most frequent case."

So in other words, yes, Leo, these researchers have shown that just capturing the sound of a traditional physical key being slid into its lock is all that's needed to recreate that key with a high level of confidence. A nearby smartphone or even the house's nearby smart

doorbell microphone provides audio which is sufficiently accurate to provide the clues. We all know how a traditional physical lock and key work; right? Inside the lock are a series of six spring-loaded pins which are each split at a different location along their length. When the proper key is inserted into the lock, the ridges on the key pushing against those internal springs positions each of the pins such that the splits in the pins line up with the edge of the lock's cylinder. Thus no pin prevents the cylinder from then freely rotating in the lock.

And I suppose because I'm a bit odd, throughout my lifetime I've often stopped to appreciate the sheer beauty of that simple invention. It requires no power. It is durable and largely weatherproof except in the face of extreme freezing. And it's extremely reliable, so much so that its failure is vanishingly infrequent. And when it does eventually fail, typically after decades of reliable use and wear, it does so in a fail-soft fashion only after providing ample clues that its need for servicing is becoming acute, such that "jiggling the key in the lock" is a longstanding meme. But mostly it achieves all this in an example of a brilliant tradeoff. We get all of that in return for accepting that it's not perfect protection.

Is it cryptographically secure? Of course not. Can it be picked and defeated by anyone skilled in the art with a few simple lock-picking tools? Yup. Are there sufficient combinations that no one else's key will open it? No. A famous hack is just to try locks with keys they don't belong to. Sometimes you just get lucky, specifically because the universe of all possible combinations is comparatively small. But the likelihood of any random key working in any random lock is low enough that no one bothers to try.

But it's exactly that comparatively small universe of possibilities that allows this research to succeed. Once the audio of a key insertion has been obtained, SpiKey's inference software gets to work filtering the signal to extract the comparatively strong metallic clicks as the key's ridges hit the lock's pins. The click occurs when one of the spring-loaded pins crosses over the top of any of the key's ridges. I have a picture, a photo, a diagram from their PDF, which shows the instance of the click occurring on the six pins. They explain, as I just did, how the lock works mechanically, and then the event of the click.

And I actually made - they have a photogram of the audio, which plays from Google, which you can hear. And Leo, you should probably put this into the podcast. It's GRC's Shortcut of the Week, so it's grc.sc/781. And there it is.

Leo: And of course we've all heard this. But you don't pay any attention to it; right?

Steve: Exactly.

Leo: And it doesn't sound this clear. But I guess modern phone microphones are good enough, they can pick it up.

Steve: Yes. So basically they say, well, so these clicks...

Leo: So the grooves on the side don't do diddly.

Steve: No, correct.

Leo: It's just the teeth.

Steve: Right.

Leo: The bittings.

Steve: Well, now, okay. So the grooves in the side do create classes of keys which will work in the lock.

Leo: Ah, right.

Steve: And those do differ among brands and within brands. So that does create subsets.

Leo: Right. But you can easily - there probably aren't that many sets, I would guess.

Steve: Exactly. There are not. And so for example many times your key won't even go in. And then, if it does, then it will go in, but it won't turn.

Leo: Right.

Steve: So anyway, the clicks drive the inference analysis. It's the time between the clicks which allows the SpiKey software, which they developed, to compute the key's inter-ridge distances. And what locksmiths refer to as the bitting depth of those ridges, which is how deeply they cut down into the key shaft and where they plateau out. If a key were to be inserted at a non-constant speed, the analysis would be defeated, though the software can compensate for small insertion speed variations. So, but if you were, like, if this freaked you out, and you were at high risk, you felt, then you could simply start inserting your key at a non-constant pace, and you would defeat this.

But given all the available acoustic information, complete disambiguity cannot be obtained. So they end up with multiple possible keyings in the best case. And this is why the paper's abstract noted that the SpiKey software will output the three most likely key designs to fit the lock that was used in the audio provided by that file, which does reduce the potential search space, as they said, from 330,000, which is the universe of possible combinations, down to just three.

They said: "When a victim inserts a key into the door lock, an attacker walking by records the sound with a smartphone microphone. SpiKey detects the timing of these clicks from the sound. We then utilize the click timestamps to compute the adjacent inter-ridge distances given a constant insertion speed. We use the computed distances to infer the relative differences of adjacent bitting depths, which SpiKey exploits to ultimately obtain a small subset of candidate keys that includes the victim's key code."

They said: "We detect all click events from the audio recording." They do subject it to a high-pass filter to reduce the impact of low-frequency ambient noise, retaining only frequencies above 15kHz that contains the information, the acoustic information about the clicks. And they said: "Subsequently, we identify the starting point of each click, or

its onset, in the pre-processed signal by applying change-point detection algorithm on short time windows around the computed peaks to account for their millisecond granularity."

They said: "It finds the least sum of standard deviations across two regions that transition from low to high amplitude." That is, in terms of the amplitude of the click sound. So they did some serious acoustic processing to just absolutely nail down the time event of the click. For anyone who's interested, I've got the PDF link to their research in the paper. It goes on to explain exactly how they convert the click onset timings into a few possible candidate keyings. So anyway, I just thought, you know, one more longstanding time-honored piece of real-world technology has just fallen.

Leo: It's terrible.

Steve: We can no longer insert our key into a lock without the possibility of somebody simply eavesdropping. And you can imagine, Leo, if you had a telephoto microphone at a distance, aimed at that lock, and somebody were to insert the key, it would be able to pick it up at a distance with a big parabolic mic and capture the sound. And that would be enough.

Leo: Isn't that amazing.

Steve: Isn't that cool.

Leo: Just amazing. How doable do you think that is? I mean, I know it's theoretically, of course, but...

Steve: I mean, they did it. They recorded it.

Leo: And they were able to make the key.

Steve: They wrote the software, and it designed three keys, and one of those three opened the lock.

Leo: Wow. Isn't that great. I think I just - you've got to admire the ingenuity and the cleverness involved, whether this is a - I can see a big three-letter agency using this. They should write this into the next Jason Bourne script or something. I think that'd be...

Steve: So it would be useful in a situation where you would be observed picking a lock. Certainly a three-letter agency would have people who can pick a lock.

Leo: Get right in.

Steve: I can pick a lock. You can pick a lock. You know, techies know how to do that. It's just, you know, it's not that difficult. But during the process you're observable. So if you had a scenario where someone posing as a cable TV serviceman or maybe the house cleaner needed to just be able to walk up, quickly insert the key, and enter, you'd want to be prepped ahead of time. And so this would allow you to produce one of three keys where they could look like they were fumbling for the right key among their key ring, but in fact they were trying the subset of possibilities and then say, oh, yeah, there it is, and then they waltz right in with everybody else standing around watching them.

Leo: Wow, that's wild. Steve, you've done it again, always do. It's fascinating stuff, and that's why we listen each and every Tuesday to Security Now!. You can get Steve's famous SpinRite, the world's best hard drive maintenance and recovery utility at his website, GRC.com. Version 6 is out; 6.1 is on its way. Buy 6 today, you'll get 6.1 for free, and you could participate in the building of 6.1, which is moving apace. That's all at GRC.com.

You'll also find the show there. He's got 16Kb and 64Kb versions of the show, the audio. And he's got transcriptions, which are very handy if you like to read along while you listen. He also has lots of other free stuff there. So check it out, GRC.com. You can leave feedback there, GRC.com/feedback. Or leave it for him on Twitter. Steve is, yes, a Twitter user, and his Twitter handle is @SGgrc. Do you respond to people though, as much as just, if people leave you a message, you get it?

Steve: I do when I can. But frankly, there's, I mean, it's just...

Leo: That would be overwhelming, yeah.

Steve: It is, yeah. I figure people would rather I got SpinRite 6.1 done than spend a day responding to private tweets. So I try to read them, yeah.

Leo: You don't have to explain to me. I get a lot of email every day. And it breaks my heart because it's always, for me, it's people who listen to The Tech Guy and who have fairly basic questions, and they're suffering, and they can't find help. But if I answered that email I wouldn't be able to do anything else. So it's all we can do. We do what we do as best we can. Of course what you should do is just come back here every Tuesday, round about 1:30 Pacific, that's 4:30 Eastern, that's 20:30 UTC. That's when we record the show.

You can watch us do it live at TWiT.tv/live. There's audio and video there. If you're doing that, chat with us. The chatroom is irc.twit.tv. You can also get on-demand versions of the show, not just at Steve's site, but at our site, in this case TWiT.tv/sn. Of course it's on YouTube, and you can always subscribe. Get your favorite podcast app and subscribe to Security Now! because you don't want to miss an episode. You want the complete set. Collect all 999. Eventually that's the number. This is Episode 781. And I thank you, Steve. Have a great week, and we'll see you next week.

Steve: Thank you, my friend. Right-o, bye.