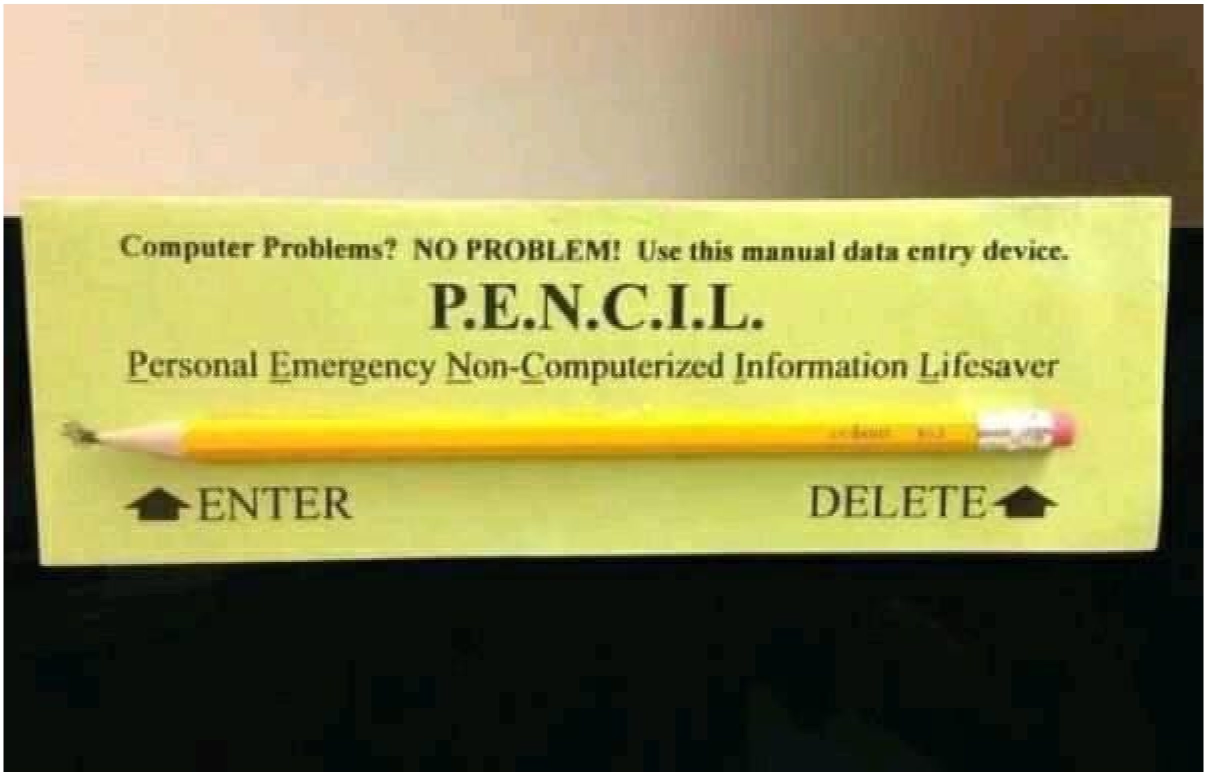
Security Now! #727 - 08-13-19

BlackHat & DefCon

This week on Security Now!

This week, as expected, we look at some of the events and announcements from last week's Black Hat and Def Con conference events. Microsoft and Apple have upped the ante for bug hunters, the Chaos Computer Club shreds a hotel's door lock security, a serious philosophical design flaw is revealed to be present in 40 signed device drivers, and Google vows to continue its Incognito-mode battle. We also have some SQRL news, some fun miscellany and some interesting closing the loop feedback from our terrific listeners.


Computer Problems? NO PROBLEM! Use this manual data entry device.
P.E.N.C.I.L.
Personal Emergency Non-Computerized Information Lifesaver
ENTER
DELETE

Security News

BlackHat and DefCon LAS Vegas 2019
This year's 23rd annual Black Hat Las Vegas keynote speech was delivered by Dino Dai Zovi, the mobile security lead at Square. Dino titled his keynote: “Start with Yes” and in it he discussed the ongoing transformation in security’s role in the workplace. Threatpost wrote up a very nice and succinct summary of Dino's commentary, which I want to share with our listeners since it's a perfect introduction to this year's Black Hat:

Taking as a first principle the idea that security teams now have the ear of company boards and the C-suite, the challenge becomes figuring out how to communicate most effectively within this newly collaborative environment, and how to have the most impact organizationally.

One of the first things to do is to realize that in today’s software-centric world, where internal teams rely on software-as-a-service and the cloud for core missions, and where DevOps is becoming the norm, security must become a shared responsibility and resource. Thus, listening to the “asks” from different division leaders in terms of building security processes that don’t cause friction is in many ways Job 1 for dedicated security personnel.

Dino said: “Saying ‘yes’ keeps the conversation going, keeps it collaborative and constructive, and opens the door for real change and real impact.”

Digging beyond this umbrella idea, Dino highlighted three transformational principles for boosting the impact of security within organizations. First, Work backward from the job to be done. Second, Seek and apply leverage, develop feedback loops and scale with software automation. And Third, Understand that culture trumps strategy and tactics, every time.

Regarding working backwards from the job… Job theory says that a job is both a function but also represents emotional context. He used the example of milkshake research at McDonalds. Market researchers found that most milkshakes were sold before 8:30 a.m., often as the only item, and via the drive through. In asking buyers what their motivation was, it turns out that they wanted something that would be easy to eat and would occupy a long commute.

Dino said: “So the job the milkshake was doing wasn’t solving hunger, it’s alleviating boredom on a long commute. Similarly, security teams need to determine what the job is to be done. Talk to internal teams, try to understand their struggles, what are they setting out to do, what adds friction and what makes things easier. When and why do they interact with security?”

In understanding what their “hiring” criteria is for a security solution (as well as the “firing criteria’) – it becomes possible to build agile way for the need at hand rather than spending time overlaying security principles that may or may not be useful, adopted or practical.

To seek and apply leverage… Dino touched on the idea of how to have the most impact with limited resources. Taking Archimedes’ classic idea of using a lever as a force multiplier to lift an object much larger than oneself, it’s possible to see this play out in security, with automation as the lever.

For example, “Security is still a small community, and the problems that we tackle can be huge, using fuzzing for finding vulnerabilities as an example of scaling security’s effectiveness via automation. We must work smarter, not just harder, through better software and better automation.”

Stressing the importance of having automated feedback loops, he said: “We have to build them explicitly. And, the tighter feedback loop wins. We have to build security services for observability, so you can understand if the protections are working and also perform anomaly detection. We have to be able to identify attackers when they’re probing, learning, attacking and succeeding.”

And as for Culture, Strategy & Tactics...

Culture is of course the term for what companies value and promote, and how its employees interact and communicate. Without a cultural shift towards embracing security, the technical aspects will fail despite best-laid plans.

Dino explained: “We in security are not outsiders anymore; we’re inside communities and companies. Now we need to use that access to improve things.”

Dino noted that making security a shared issue can go a long way to creating a safer organization. For instance, at Square, security engineers have to write code like everyone else.

“This created a cultural change – there’s a lot more collaboration and empathy for how people are operating,” Dai Zovi said. “A software engineering team would write security features, then actively go to the security team to talk about it and for advice. We want to develop generative cultures, where risk is shared. It’s everyone’s concern. If you build security responsibility into every team, you can scale much more powerfully than if security is only the security staff’s responsibility.”

This involves implementing a blame-free post-mortem process when it comes to responding to an anomaly or a vulnerability report.

Dino said: “Turn these events into inquiries where you focus on getting at the root causes of the problems.”

In the end, security teams should see themselves as an extension of internal teams, not a division apart.

He said: “Instead of saying no, start with yes and here’s how we can help. It’s all about cultivating empathy. It’s something you practice and grow. This is the way we meet the challenge of leveling up on security.”

Microsoft dangles $300,000 for Azure hacks at BlackHat...
Hoping to benignly discover unknown bugs lurking in its Azure cloud platform, at Black Hat Microsoft announced that it will be offering bounties of up to $300,000 for researchers who launch successful test exploits against the platform.

In order to support this effort in a customer-safe fashion, Microsoft has launched a dedicated Azure cloud host testing environment, called the Azure Security Lab (ASL). The ASL program allows security researchers to test attacks on infrastructure-as-a-service (IaaS) scenarios without impacting customers. These hosts are isolated from the Azure production environments that customers use, meaning that researchers will have more flexibility to research and test live exploits.

Microsoft's Kymberlee Price, the principal security program manager for the Microsoft Community and Partner Engagement Programs wrote in a blog post last week: "The isolation of the Azure Security Lab allows us offer something new: Researchers can not only research vulnerabilities in Azure, they can attempt to exploit them. To make it easier for security researchers to confidently and aggressively test Azure, we are inviting a select group of talented individuals to come and do their worst to emulate criminal hackers in a customer-safe cloud environment called the Azure Security Lab.”

Researchers with access to the Azure Security Lab may also attempt scenario-based challenges with top awards of $300,000. Starting yesterday, researchers can apply for this access at Microsoft’s website.

And, on top of the ASL announcement, Microsoft announced that it is doubling its bug-bounty rewards for researchers who discover Azure vulnerabilities. At the start of this year, Microsoft launched a bug-bounty program designed to find flaws in Azure DevOps with top rewards of up to $20,000. Now that $20,000 is being doubled to $40,000. For those who are not aware, "Azure DevOps" is a cloud service launched in 2018 that enables collaboration on code development across the breadth of a development lifecycle. The two in-scope services for the bounty program include Azure DevOps Services (formerly Visual Studio Team Services) and the latest publicly available versions of Azure DevOps Server and Team Foundation Server.

And Microsoft has quietly been paying out some nice hefty rewards. Microsoft said that it has paid out $4.4 million dollars in bounty rewards over the past 12 months across various programs.

Last month Microsoft initiated a bug-bounty program, offering payouts as high as $100,000 for the discovery of holes in identity services and implementations of the OpenID standard. These include Microsoft Account and Azure Active Directory, which offer identity and access capabilities for both consumer and enterprise applications – as well as its OpenID authentication protocol. For a company with the cash and resources of Microsoft, spending their money like this really does make sense.

And back last March, inspired by the Meltdown and Spectre flaws, Microsoft started another new bug bounty program targeting speculative execution side-channel vulnerabilities. We talked about this at the time, noting that it's a limited time program operating only through the end of the year and offering up to $250,000 for identifying new categories of speculative execution attacks that Microsoft and other industry partners are not yet aware of.

And, in keeping with doing this the right way, on Monday Microsoft also implemented explicit safe harbor terms and conditions which clearly outline how researchers, acting in good faith, can safely report bugs without facing legal repercussions. This is something that ought to be codified into law for the benefit of our cyber future.

Kymberlee Price wrote: “Microsoft is committed to ensuring our cloud is secure from modern threats. We built Azure with security in mind from the beginning, and work to help customers secure their Azure cloud environment with products such as Azure Sentinel and Azure Security Center. And if a situation arises, our Cloud Defense Operation Center (CDOC) and security teams work around the clock to identify, analyze and respond to threats in real time.”

Hotel Chaos from Germany's Chaos Computer Club

The latest trend in high-end hotel door keying is known as "Mobile Keys" -- "Mobile" as in "Mobile Phone."

Check out this website: https://www.openkey.co/

“Universal Mobile Keyless Entry For Hotels Worldwide”
Improve guest reviews and gain competitive advantage with the latest hotel technology.

The Industry Standard For Hospitality Technology

The mobile revolution is here. Guests want to use their smartphones to control every aspect of their stay – and the major hotel chains around the world are responding. Keyless entry, mobile check-in and check-out equal a quantum leap for the guest experience. OpenKey makes delivering on guest expectations simple, fast and affordable.

- Step 1: Reservation confirmation email sent 24 hours prior to arrival contains mobile key info and download link
- Step 2: Guest downloads app and registers with name and mobile number
- Step 3: Guest can check-in via mobile, at kiosk or at front desk. Mobile key is sent upon Check-In
- Step 4: Guest uses smartphone to access their room with the tap of a key button

Unfortunately, the guys at BlackHat demonstrated Step 5: “Bad guys can waltz into any locked hotel room they choose.”

So, yeah... It sounds really great. It reduces check-in friction. What with such a slick website you just KNOW it must be super-secure! But probably not too surprising by now, it's not.

The reporting reads: Researchers developed an exploit that allowed them to perform an array of malicious functions against these so called “mobile keys”. A vulnerability in a popular IoT lock key – used chiefly by a high-end hotel in Europe – allowed researchers to break into hotel rooms. The locks in question are dubbed “mobile keys” because of their reliance on mobile phones as opposed to card-based access such as those based on mag-strips and RFID.

Researchers at Black Hat USA 2019 showcased how they were able to circumvent an Internet of Things connected key system utilized by an unnamed European hotel. The name of the hotel and specific IoT lock system was not identified for safety reasons, as the locks are still deployed in the hotel.

The researchers explained: “We went to do the one thing a mobile hotel key is supposed to prevent: wirelessly sniff someone entering his room – or just unlocking the elevator – and then reconstruct the needed data to open the door with any BTLE enabled PC or even a Raspberry Pi.”

Okay, so… without knowing ANY of the details, this is ridiculously horrific because it is such a trivial problem to solve with a proper design. At its root, from what we know, we're talking about a classic replay attack. All that's needed to prevent this is for the door, when challenged to unlock, to provide a nonce for the phone to sign and return. The door contains a software ratchet. This is a counter which feeds a secretly-keyed AES symmetric cipher. Each door lock is configured with its own secret key which is never exposed. The AES cipher which encrypts a counter, produces a public elliptic key which is used to verify signatures. So the door lock first checks the key that it is currently valid for and has been using. If that fails, it checks ahead to the next public key to see whether that one can verify the returned signature. If not, it ignores the request. But if the next key does successfully verify the request's signature it makes the next key permanent, ratcheting forward and forgetting the previous no-longer-valid key. This means that the door locks do not need to communicate with the hotel. Each door lock is able to operate autonomously with its own secret key which determines the sequence of its public keys. The hotel system knows each room’s secret key so it's able to issue the proper private signing key to each guest for the proper room. If that system is designed correctly, no one with a copy of the Mobile Key software, and the ability to eavesdrop on the conversation, is able to gain any advantage from doing so. This is Crypto 101.

And this is what I mean when I say that our modern cryptographic tools are so cool and can provide so much nifty functionality. And it's all available for free in the well-tested and audited LibSodium library on Github.

Meanwhile, a presentation during Def Con last Saturday generated tech press headlines including:
“Driver Disaster: Over 40 Signed Drivers Can’t Pass Security Muster”
“Researchers find security flaws in 40 kernel drivers from 20 vendors” and
“Over 40 Drivers Could Let Hackers Install Persistent Backdoor On Windows PCs”

And when you stop to think about it, this just makes sense. Device drivers occupy a sort of security loophole in our sytstems today. They are not the primary OS, so they don't get the scrutiny that the OS gets. They are provided by random 3rd parties who probably have the best intentions, but they are likely operating under pressure to ship and their focus is on the driver working and being stable, not on it being bulletproof against direct attack. And, finally, they often run down in the kernel alongside the OS with direct access to the system's hardware and the high privileges that such access requires. So... Yeah... This was sort of inevitable.

In ZDNet's words...

At the DEF CON 27 security conference today in Las Vegas, security researchers from Eclypsium gave a talk about common design flaws they found in more than 40 kernel drivers from 20 different hardware vendors.

The common design flaws is that low-privileged applications can use legitimate driver functions to execute malicious actions in the most sensitive areas of the Windows operating system, such as the Windows kernel.

Mickey Shkatov, Principal Researcher at Eclypsium told ZDNet in an email earlier this week: "There are a number of hardware resources that are normally only accessible by privileged software such as the Windows kernel and need to be protected from malicious read/write from userspace applications. The design flaw surfaces when signed drivers provide functionality which can be misused by userspace applications to perform arbitrary read/write of these sensitive resources without any restriction or checks from Microsoft."

Shkatov blames the issues he discovered on bad coding practices, which don't take security into account: "This is a common software design anti-pattern where, rather than making the driver only perform specific tasks, it's written in a flexible way to just perform arbitrary actions on behalf of userspace."

[[Okay... That's even WAY worse than I was expecting. This means that the driver is an interpreter that takes commands from userland and follows them. Unbelievable.]]

Mickey explains: "It's easier to develop software by structuring drivers and applications this way, but it opens the system up for exploitation."

[[Amen to that!]]

Shkatov said his company has notified each of the hardware vendors that were shipping drivers that allow userspace apps to run kernel code. Vendors who issued updates are listed below.

- American Megatrends International (AMI)
- ASRock
- ASUSTeK Computer
- ATI Technologies (AMD)
- Biostar
- EVGA
- Getac
- GIGABYTE
- Huawei
- Insyde
- Intel
- Micro-Star International (MSI)
- NVIDIA
- Phoenix Technologies
- Realtek Semiconductor
- SuperMicro
- Toshiba

“Some vendors, like Intel and Huawei, have already issued updates. Some which are IBVs [independent BIOS vendors] like Phoenix and Insyde are releasing their updates to their customer OEMs,” Shkatov told ZDNet.

Mickey said he did not name all the impacted vendors, though, as some "needed extra time due to special circumstances" and future fixes and advisories will be released in the future.

He said he plans to publish the list of affected drivers and their hashes on GitHub, after the talk so users and administrators can block the affected drivers.

In addition, Shaktov said Microsoft will be using its HVCI (Hypervisor-enforced Code Integrity) capability to blacklist drivers that are reported to them.

However, Shaktov said that the HVCI feature is only supported on 7th gen Intel CPUs and onwards. Manual intervention will be needed on older systems, and even on newer Intel CPUs where HVCI can't be enabled.

"In order to exploit vulnerable drivers, an attacker would need to have already compromised the computer," Microsoft said in a statement. "To help mitigate this class of issues, Microsoft recommends that customers use Windows Defender Application Control to block known vulnerable software and drivers. Customers can further protect themselves by turning on memory integrity for capable devices in Windows Security. Microsoft works diligently with industry partners to address to privately disclose vulnerabilities and work together to help protect customers."

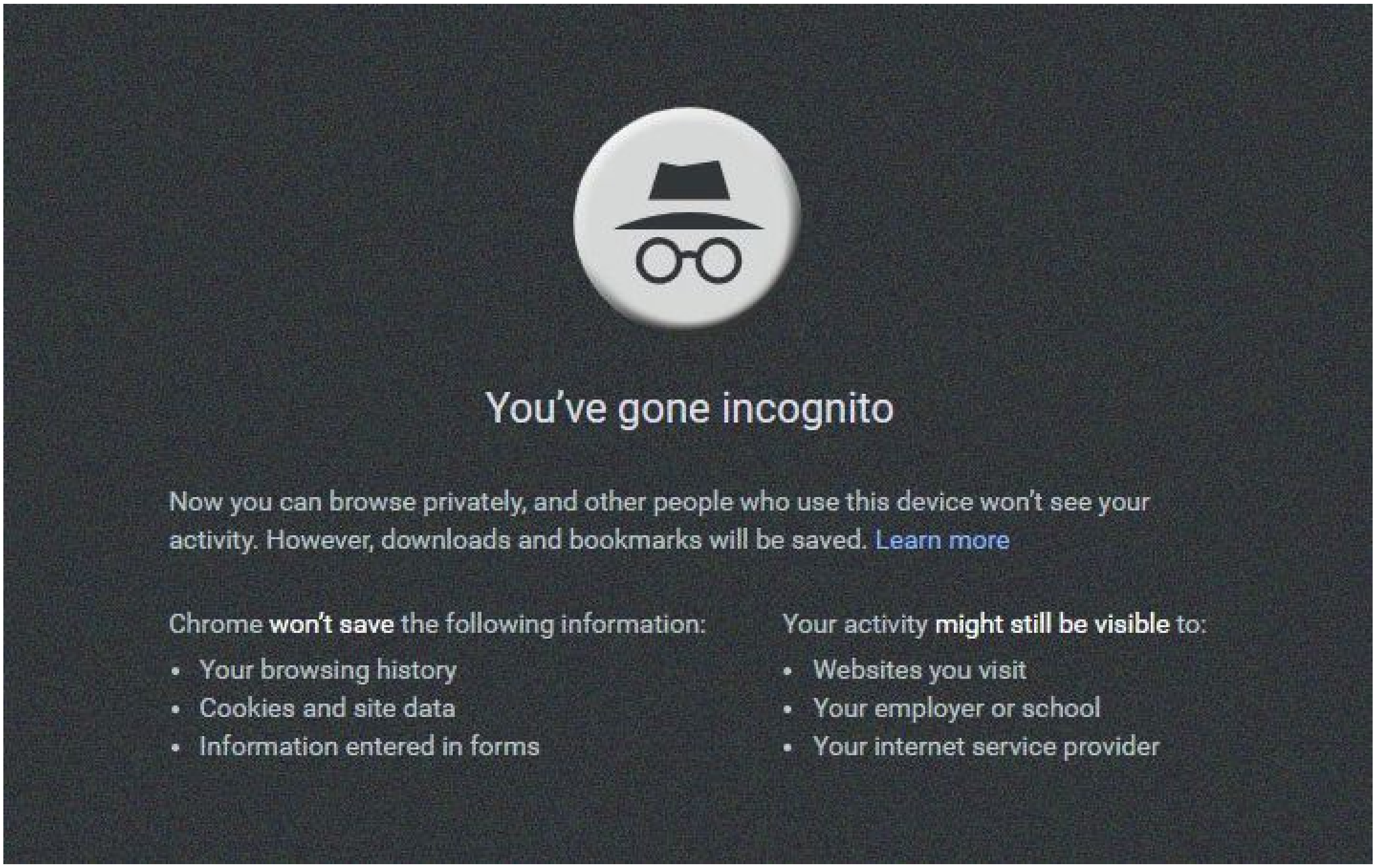
[[Nice try, Microsoft... but that's just a bunch of CYA nonsense.]]

The day after their talk, so Sunday, the Eclypsium guys published additional details in what has got to be one of the best titled blog postings in recent times:

https://eclypsium.com/2019/08/10/screwed-drivers-signed-sealed-delivered/
https://eclypsium.com/wp-content/uploads/2019/08/Screwed-Drivers.pdf

"Screwed Drivers – Signed, Sealed, Delivered”

Google’s battle to allow its Incognito users Incognito’ness to be Incognito…



You’ve gone incognito

Now you can browse privately, and other people who use this device won’t see your activity. However, downloads and bookmarks will be saved. Learn more

Chrome **won’t save** the following information:

- Your browsing history
- Cookies and site data
- Information entered in forms

Your activity **might still be visible** to:

- Websites you visit
- Your employer or school
- Your internet service provider

As we know, Chrome's release 76 was intended to close a loophole that various commercial paywalled websites were using to detect when their visitors were viewing the site through their web browser's incognito mode.

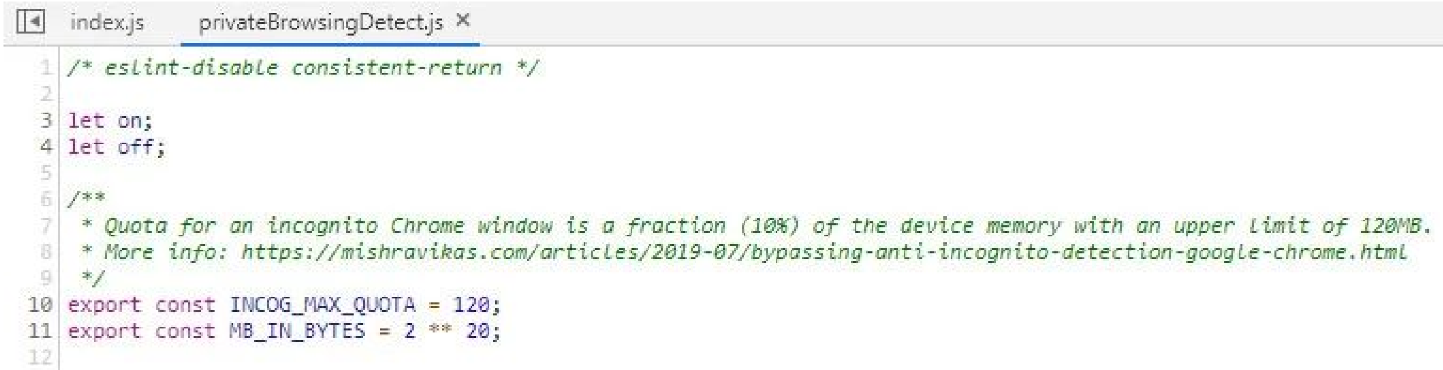
Incognito detection had been implemented because Incognito mode inherently flushed the simple-minded cookie histories that were being used to permit a paywall access compromise and tease where a limited number of pages **could** be viewed before the site's paywall would slam shut to block further access.

The understandable feeling was that if a visitor wished to be Incognito when surfing the web they should have that right, and that should also extend to include the fact that they were choosing to be Incognito in the first place. Prior to Chrome's release 76, Incognito mode simply disabled the JavaScript FileSystem API. Since this was trivial for a site's scripting to detect, Incognito-mode visitors were blocked from having **any** access. So it was, essentially: “Let us store our crap on your computer or you don't get to see any of our site.”

What Chrome 76 **now** does is implement a RAM-based FileSystem API so that the file system appears to work, but it's volatile and won't store anything permanently.

So we know what happened next, right? A RAM-based file system won’t behave exactly like a non-volatile file system, which makes it detectable. And, yes, some sites such as The New York Times immediately adapted their code to do just that.

Chrome's filesystem API presents a smaller maximum file system size of just 120MB. This is not seen when the browser is NOT in Incognito mode. And writes to RAM present a consistent timing compared to writes to physical media where timing will vary. Both of these Incognito detection bypass bypasses have been demonstrated and, as I mentioned, they have been found to be used in the wild.



```
/* eslint-disable consistent-return */

let on;
let off;

/**
 * Quota for an incognito Chrome window is a fraction (10%) of the device memory with an upper limit of 120MB.
 * More info: https://mishravikas.com/articles/2019-07/bypassing-anti-incognito-detection-google-chrome.html
 */
export const INCOG_MAX_QUOTA = 120;
export const MB_IN_BYTES = 2 ** 20;

```

A snippet of updated Incognito-detection code from The New York Times.

Bleeping Computer reached out to Google to inquire about these two new Incognito detection methods. They were told that Google stands by their previous statement and position that they will <quote> “work to remedy any other current or future means of Incognito Mode detection.”

<sigh> So we have another new cat & mouse game. And this is dumb, since this is a game that Chrome can and will ultimately win. Chrome can trivially simulate the varying timing of reads from and writes to volatile media, and they can remove the declared RAM-based file system size limitation.

As Google said, if sites wish to be paywalled they should require all visitors to create an account. Once that account is created, then some free access could be metered out under the site’s control. But since visitors could still create throwaway accounts, a credit card or other payment means would also be needed to provide an anchor for the user’s identity. And all of this would clearly reduce the site’s traffic, since many users would choose to go elsewhere rather than to create yet one more account somewhere. And it would also mean that no one could use the site without at least creating a temporary account. This is all clearly a mess which arises from the fundamentally irresolvable conflict inherent in the goal of wanting to provide **some** access to visitors who wish to have complete anonymity.

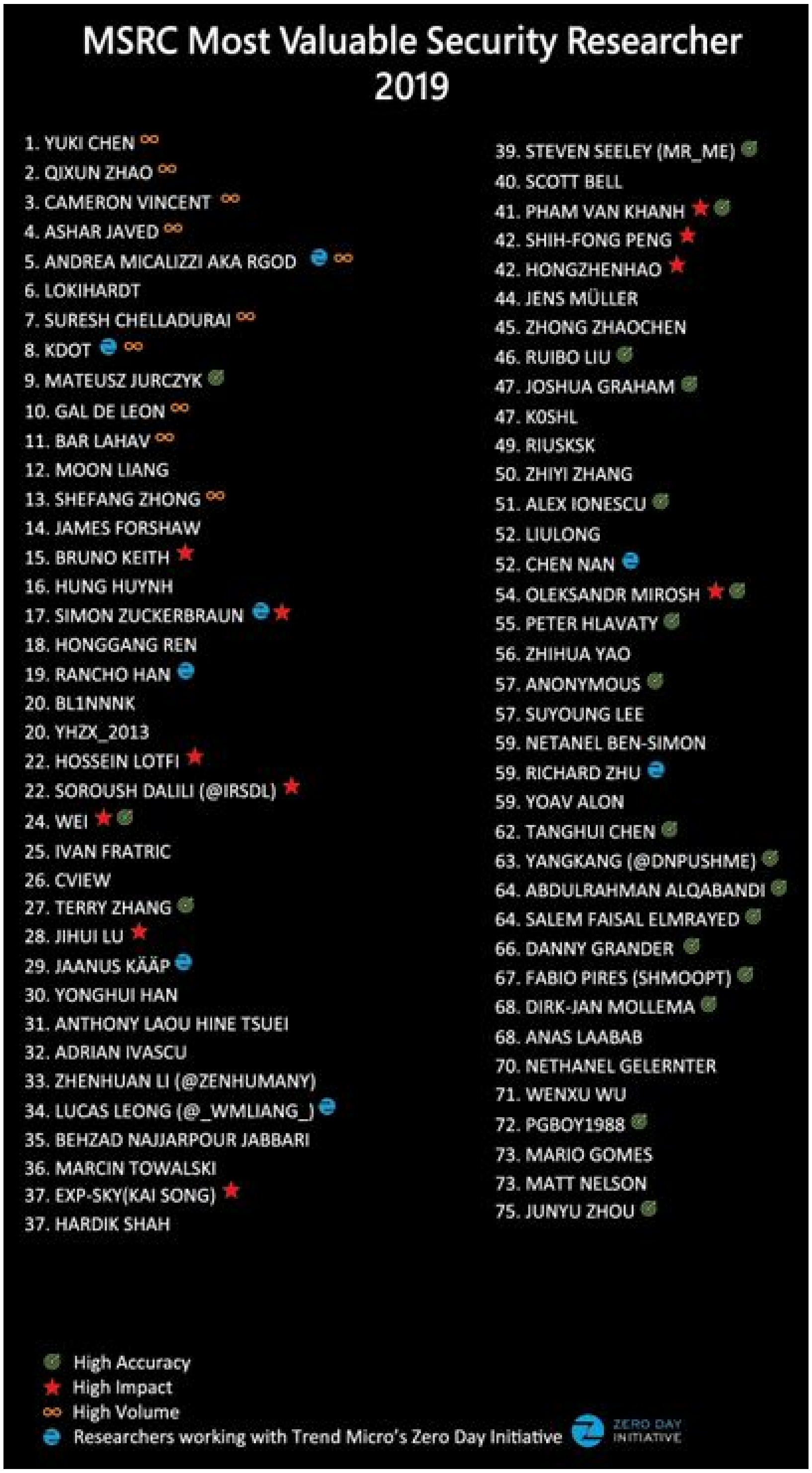
So to Google I say “Bravo!” and to these websites I say “good luck with that.”

Once again, Microsoft ranks the industry's top bug hunters

As has become something of a Black Hat tradition, Microsoft this year again announced the top security researchers and enterprise partners who responsibly discovered and disclosed the greatest number of vulnerability and zero-day reports affecting its software products. Many in the security industry now use this list as a guide to today's top bug hunters.

And, with good cause, security researchers who rank on the list often tout it as one of their highest career achievements... because there's a LOT of competition there.

According to Microsoft, this year's top security researcher is Yuki Chen of Qihoo 360's Vulcan team. And taking second place is Yuki's colleague, Qixun Zhao, who also won a Pwnie Award for Best Privilege Escalation Bug. All in all, Qihoo 360's Vulcan team managed to place eight researchers in this year's ranking of the top 75…



MSRC Most Valuable Security Researcher
2019

1. YUKI CHEN ∞
2. QIXUN ZHAO ∞
3. CAMERON VINCENT ∞
4. ASHAR JAVED ∞
5. ANDREA MICALIZZI AKA RGOD ⓔ ∞
6. LOKIHARDT
7. SURESH CHELLADURAI ∞
8. KDOT ⓔ ∞
9. MATEUSZ JURCZYK ✓
10. GAL DE LEON ∞
11. BAR LAHAV ∞
12. MOON LIANG
13. SHEFANG ZHONG ∞
14. JAMES FORSHAW
15. BRUNO KEITH ★
16. HUNG HUYNH
17. SIMON ZUCKERBRAUN ⓔ ★
18. HONGGANG REN
19. RANCHO HAN ⓔ
20. BL1NNNK
20. YHZX_2013
22. HOSSEIN LOTFI ★
22. SOROUSH DALILI (@IRSDL) ★
24. WEI ★ ✓
25. IVAN FRATRIC
26. CVIEW
27. TERRY ZHANG ✓
28. JIHUI LU ★
29. JAANUS KÄÄP ⓔ
30. YONGHUI HAN
31. ANTHONY LAOU HINE TSUEI
32. ADRIAN IVASCU
33. ZHENHUAN LI (@ZENHUMANY)
34. LUCAS LEONG (@_WMLIANG_) ⓔ
35. BEHZAD NAJJARPOUR JABBARI
36. MARCIN TOWALSKI
37. EXP-SKY(KAI SONG) ★
37. HARDIK SHAH
39. STEVEN SEELEY (MR_ME) ✓
40. SCOTT BELL
41. PHAM VAN KHANH ★ ✓
42. SHIH-FONG PENG ★
42. HONGZHENHAO ★
44. JENS MÜLLER
45. ZHONG ZHAOCHEN
46. RUIBO LIU ✓
47. JOSHUA GRAHAM ✓
47. K0SHL
49. RIUSKSK
50. ZHIYI ZHANG
51. ALEX IONESCU ✓
52. LIULONG
52. CHEN NAN ⓔ
54. OLEKSANDR MIROSH ★ ✓
55. PETER HLAVATY ✓
56. ZHIHUA YAO
57. ANONYMOUS ✓
57. SUYOUNG LEE
59. NETANEL BEN-SIMON
59. RICHARD ZHU ⓔ
59. YOAV ALON
62. TANGHUI CHEN ✓
63. YANGKANG (@DNPUSHME) ✓
64. ABDULRAHMAN ALQABANDI ✓
64. SALEM FAISAL ELMRAYED ✓
66. DANNY GRANDER ✓
67. FABIO PIRES (SHMOOPT) ✓
68. DIRK-JAN MOLLEMA ✓
68. ANAS LAABAB
70. NETHANEL GELERNTER
71. WENXU WU
72. PGBOY1988 ✓
73. MARIO GOMES
73. MATT NELSON
75. JUNYU ZHOU ✓

✓ High Accuracy
★ High Impact
∞ High Volume
ⓔ Researchers working with Trend Micro's Zero Day Initiative

And not to be left behind, Apple has also bumped its bounties!
Apple has just updated the rules of its bug bounty program, announcing some major changes which will come into effect this fall.

Apple has enormously increased the maximum reward for its bug bounty program from $200,000 to $1 million making it, by far, the biggest bug bounty offered by any major tech company for reporting vulnerabilities in its products. The $1 million payouts will be rewarded for a severe deadly exploit—a zero-click kernel code execution vulnerability that enables complete, persistent control of a device's kernel. Less severe exploits will qualify for smaller payouts. So it's now possible to become a hacker millionaire overnight by finding just one serious bug in an Apple product.

And, moreover, Apple's bug bounty program does not only apply to security vulnerabilities in the iOS mobile operating system, but also covers all of its operating systems, including macOS, watchOS, tvOS, iPadOS, and iCloud. Until now, for the past three years, Apple's bug bounty program has only rewarded security researchers and bug bounty hunters for discovering vulnerabilities in iOS. (Note that limitation remains in effect until the newly expanded program comes online this fall.)

But wait... there's more! (Or as Steve Jobs would have said: "One More Thing"...)

Starting next year, Apple will also provide pre-jailbroken iPhones to select trusted security researchers as part of the iOS Security Research Device Program. These devices will have far deeper access than iPhones available to everyday users, including access to ssh, root shell, and advanced debug capabilities, allowing researchers to hunt for vulnerabilities at the secure shell level. Although anyone can apply to receive one of these special iPhones from Apple, the company will hand out only a limited number of these devices and only to qualified researchers.

And there's still more...

In addition to its maximum reward of $1 million, Apple is also offering a 50% bonus to researchers who find and report security vulnerabilities in its pre-release software ahead of its public release. So, potentially, a maximum reward of up to $1.5 million. This is smart, since finding pre-release bugs is good for everyone.

Applications for Apple's revised bug bounty program will be open later this year, and this will be open to all researchers, rather than some limited number of security experts approved by Apple.

This expansion and massive boost in the payout of Apple's bug bounty program are likely to be welcomed by security researchers and bug bounty hunters who either publicly disclose vulnerabilities they discovered in Apple products or sell it to private vendors like Zerodium, Cellebrite, and Grayshift who deal in zero-day exploits, for profit. This seems like a clear and smart move on Apple's part. A researcher who does find a show-stopping exploit in an Apple device would be hard-pressed to offer it to Apple for a pittance when the likes of Zerodium would pay much more. Now the developer can be a good guy, do the right thing... and receive a top-of-the-industry monetary reward.

SQRL

Ben Fletcher in Dublin, Ireland
Subject: Dublin Visit
Date: 09 Aug 2019 16:24:36
:

Dear Steve, I was delighted to hear on Security NowN that you are coming to Dublin and I will be at your OWASP talk. I moved to Dublin recently having served in the Royal Air Force in the UK for the last 16 years. I was originally a mechanical engineer by degree but quickly realised that IT was the future. I started volunteering for IT engineering jobs and haven't looked back. However, this transition left me with a pretty steep learning curve. You might remember the days when Conficker hit a number of networks, we were not immune. As I frantically searched the web for information to be able to understand the issue and brief my superior officers I came across Security Now. Having listened ever since and all previous. Suddenly I was the subject matter expert, deploying and commanding tiger teams to remediate the problem across the UK and the world. I honestly can't thank you enough for the invaluable service you have provided me and the knowledge I've gained over the years. I am now working in Grant Thornton as a Cybersecurity consultant doing all sorts of fun! I'm sure your timetable will be tight while in Ireland but I would be delighted to take you out for dinner with our team or anything you fancy whether it be something cultural or touristy. It is the least I can offer for you for helping me develop as an IT professional. If there is anything we can do for you while you are in Ireland please do not hesitate to ask. I look forward to meeting you in person. Kind regards, Ben

First off, I very much appreciate the offer. But Lorrie and I are already committed to sharing a meal with the Dubin OWASP gang, and when we're not doing that I strongly suspect that we're going to want to just wander around aimlessly soaking up the local environment and culture.

I did want to note that I have put up a short calendar listing the three forthcoming planning SQRL presentations, here in Orange County, California next Thursday, and in both Dublin, Ireland and Gothenburg, Sweden. The Calendar entries contain links to the presentation announcements as well as links to register to attend the meetings if you are interested and able to:

https://www.grc.com/calendar.htm
https://grc.sc/cal

Miscellany

I expected a robust response from our listeners to last week's "Steve's File Sync Journey" podcast, and our listeners did not disappoint. It's very clear that this topic is of great interest to our audience. So there will be a follow-up extensive results podcast.

The way the timing of the SQRL/OWASP presentation trip is working out, with my trip abroad sending in Boston to meet Leo for the LastPass event the following Thursday, I'm going to be away for two Security Now! Podcasts. So Leo and I will pre-record the Security Now! Podcast for the first Tuesday I'll be away on the Saturday afternoon before I depart. But for the following Tuesday, since we were unable to find a time soon after we return to record for that one late, we're instead going to pre-record a podcast to fill-in that week... on the results of my several months of exploring multi-system file synchronization solutions.

Along those lines, a note from listener Cliff Spooner in Utah struck me as intriguing and, I thought, very clever: **Subject: SN: 726 - File encryption option - Veracrypt w/ Dropbox:**

Steve,

After listening to your podcast #726 I thought I would share what I do to secure sensitive information in Dropbox. I use Veracrypt, which sounds like a terrible idea because of the large file size of one encrypted container, but Dropbox is magic and it is the only sync solution that is able to transfer only the changes to the container when they are made. Other sync solutions need to re-upload the entire container. You will upload a large container once which may take some time, but after that it will only be changes. Maybe this will help solve some of your issues.

Thanks,
Cliff

<<< Explain why this is kind of genius >>>

Although I still need more time to reach conclusions about robust, flexible and secure multi-machine file synching, I need to acknowledge the massive response I received about one solution in particular: **SyncThing**

I now have it running on my Drobo and on both location's workstations. Both locations are behind NAT routers, and my secondary site is actually behind chained double NAT, since I placed a NetGate SG-1100 pfSense firewall/router in front of the ASUS router. And although I do have static mappings and filters in place at each end-point with pfSense -- and I could manually punch some holes through again if necessary -- I first wanted to try firing these babies up with everything behind those NAT routers. SyncThing is UPnP capable, so =if= I had UPnP enabled anywhere (I don't) we know that it could have setup port mappings itself. But as we talked about many moons ago on this podcast, it's possible to punch-through NAT routers with a properly set up Rendezvous server. So it can work when it's done right... and SyncThing does it right. **All** of the SyncThing instances are directly connected without any external relaying required despite the fact that everything is safely behind NAT. And if anyone feels skittish about having a 3rd-party performing NAT punch-through, SyncThing allows you to operate your own NAT Rendezvous server.

I have an idea for an interesting hack that might work to bridge SyncThing to hosted cloud providers for some of the benefits that provides such as automatic versioning and public links.

Regarding last week's Picture of the Week: The "2 & 4 then 3" door lock:
A good friend was familiar with those combination door locks. He noted that "Press 2 & 4 together, then 3" is, in addition to being printed on the door's paper... also the default factory-set combination for those locks! Unbelievable.

SpinRite

"Toe Stubbed" (from the SQRL newsgroup...)
Sue's trusty and crusty old Win XP machine finally died. I had a pair of drives in a mirrored RAID, but one died silently and then the second one bit theS dust. Two weeks ago Sue phoned to give me the bad news. So after that week's podcast I grabbed her machine. SpinRite was able to bring the drive back to life and I made an image of it so that I could capture the important volatile files.

Then I set up a Win7 machine for her. I chose Win7 since it more easily runs "XP mode" in case some 16-bit code is mission critical. As it happens, we had still been keeping an instance of FoxPro v2.6 alive all these years so that Sue could look up early purchasers of SpinRite in case someone with a v1.0, v2.0, v3.1, 4.0 or 5.0 might wish to upgrade to v6.0. I had explained all of this in a note to the SQRL newsgroup since this temporarily side-tracking my focus on the SQRL cryptography documentation, which is what I'm working on now. A member of that group named "Jeff Root" responded to that with the following note:

Jeff Root:
I was a v.5 owner, and when I wanted to get the latest, I just bought the v.6 release. It honestly never occurred to me that GRC would be able to find my previous purchase in their records. Especially since years had passed.

Now you're running FoxPro in an emulator, just in case someone wants to upgrade? I wish Microsoft has as much respect and sense of responsibility for their customers. Perhaps if GRC had shareholders, the answer would be different. ;)

Closing The Loop

Spawnandjesus @spawnandjesus
Hey Steve. I haven't jumped into this week's podcast yet, but I thought I'd let you know your latest Ransomware creators are obviously into anime. The word Ryuk, should be pronounced REE-OOK, as he's a character from a series called Death Note.

Also, thanks for the best tech podcast on the wire. I've been listening for over a year, originally discovered your website as a kid in the 1990s. I love SQRL, I love SpinRite. I'm patiently waiting for the next version to purchase another copy!

Graham Booker @gbooker
Random SN question: I've noticed in many of your Show Notes for SN, you have "~30~" at the end. What's the history/meaning of this? The SQRL logo in its place in 725 reminded me of this.

Wikipedia: -30- has been traditionally used by journalists in North America to indicate the end of a story. It is commonly found at the end of a press release. There are many theories about how the usage came into being, e.g. from that number's use in the 92 Code of telegraphic shorthand to signify the end of a transmission in the American Civil War era. It was included in the Associated Press Phillips Code of abbreviations and short markings for common use. It was commonly used when writing on deadline and sending bits at a time to be typeset, as a necessary way to indicate the end of the article.

Loren Burlingame @TheIxian
Steve, I just listened to the latest SN where you were talking about synchronization. I am not sure if you have run across my favorite synchronization utility, SyncBack (2brightsparks.com/syncback/sbpro…), but figured it is worth mentioning in case you haven't. It is Windows software and can run from scheduled tasks. But it supports everything you could ever want in a sync utility including synchronization to S3 and other cloud providers (OneDrive, GDrive, etc), ransomware protection, versioning, encryption, etc. You have to pay for the Pro version to get all of those features, but it is worth the money, imo. Thank you so much for Security Now, I never miss an episode and every Wednesday morning is like a mini Christmas when I see the show in my feed.
I should almost mention that I have also run into all of those issues that you described with the OneDrive and GDrive clients and I stopped using them in favor of SyncBack. The beauty is that, since I am using OneDrive's (and GDrive's) back-end, I can access the files on mobile platforms as well.

[[I liked that Loren noted that he or she had also run into all of the issues that I described with Google Drive and OneDrive. But I'll also note that none of those problems manifested immediately. So it'll likely take a bit of time to be sure about the best solution.]]