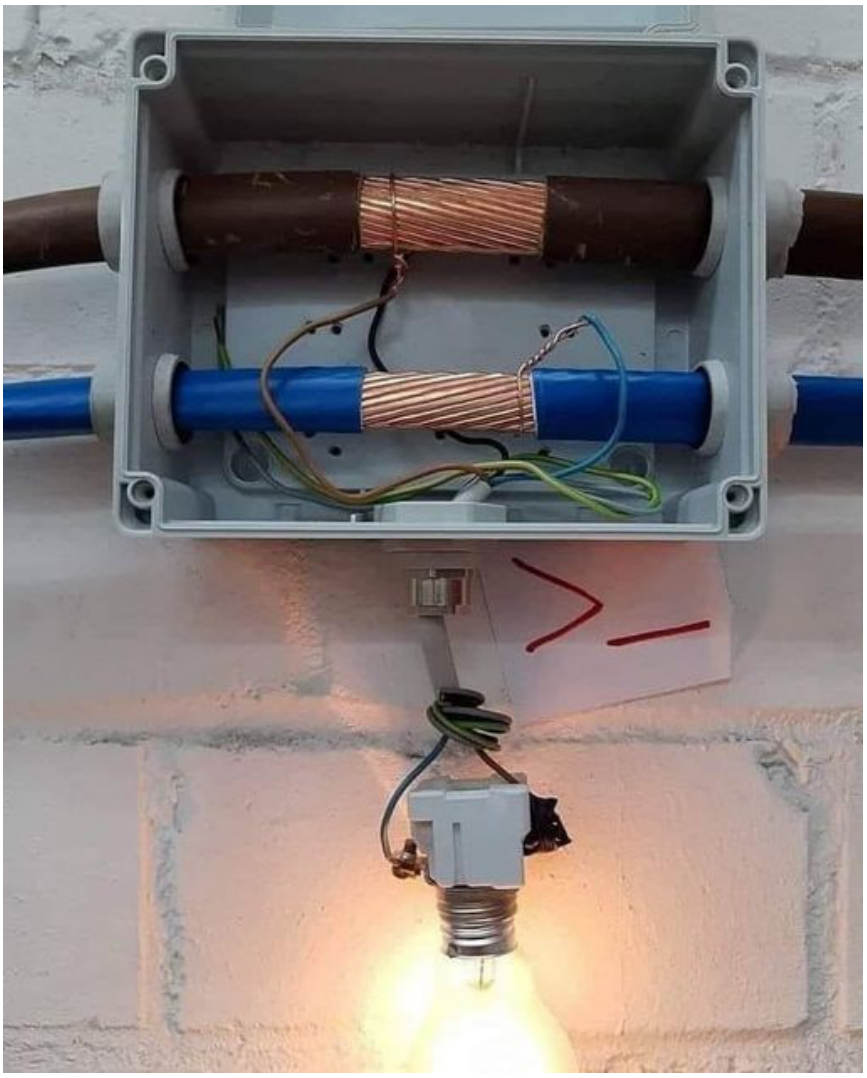
Security Now! #930 - 07-11-23

Rowhammer Indelible Fingerprinting

This week on Security Now!

Could it be that yet another SQL injection flaw was found in the MOVEit Transfer system and what more has been learned about last month's widespread attacks? What's a “Rug Pull”?? What horrible conduct was the popular Avast A/V found to be engaging in? Did China actually create their own OS? Version 1 is out! How many times can we say “TootRoot” while covering one story? What's the controversy surrounding the recent release of Firefox 115? Did Russia just successfully disconnect itself from the Internet? What are modern Internet honeypots discovering? How much of your life's saving should you transfer into online cryptocurrency exchanges? (Okay, that's an easy one.) What did EU agencies just rule against Meta and Google? What happened to Apple's quickly withdrawn Rapid Security Response update? And after a bit of miscellany and listener feedback, we're going to look at the return of Rowhammering for the purpose of creating indelible fingerprints.

A different definition of insanity



Security News

Another Critical Unauthenticated SQLi Flaw Discovered in MOVEit Transfer Software

Anyone who understands the inherent danger of exposing the unnecessarily powerful SQL command stream to web servers will not be surprised to learn that yet another instance of this extremely common and potentially devastating source of vulnerabilities has been uncovered in Progress Software's MOVEit Transfer system. Fortunately, the previous problems drew the attention of security researchers at HackerOne and Trend Micro's Zero Day Initiative who decided to have their own look at MOVEit's code. They responsibly disclosed their discoveries so that patches could be prepared and made available to Progress Software beleaguered users. At the same time, Progress also patched another pair of high severity vulnerabilities which were not SQL injection.

So, it's time to update that system again for the 4th time in as many weeks, and I seriously don't know what I would advise its users to do. I would be feeling extremely uncomfortable if I understood the inherent danger that this system presents. As we covered three weeks ago in episode #928, literally thousands of Progress' customers were penetrated and had their internal private data exfiltrated to Russia, then extorted by Russia's Clop cybercriminal organization.

The problem going forward is that due to the extremely poor security design of this MOVEit system, there's no reason to believe that the last serious data exfiltration problem has now finally been found. Sure, the system is doubtless far more secure than it was, but this class of bug is slippery.

And as for MOVEit...

While we're once again on the topic of MOVEit hacks: The most current forensics from HuntressLabs suggests that the attacks on the thousands of MOVEit using companies appear to have been limited to data theft and data extortion. The Clop cybercrime group has not yet been observed deploying ransomware in any of the incidents linked to its exploitation of MOVEit. Huntress also said that it has not observed the Clop gang expand its access to full network compromises. The gang has apparently deliberately limited itself to infecting and infiltrating only the hacked MOVEit appliance itself. So things could have easily been far worse if a far more aggressive attacker had been behind this.

What's a "Rug Pull" ??

So I was scanning a news blurb about another crypto exchange mess. And understand that the bar has been raised fairly high for me to even bring something like this up. There's just so much of it going on. The entire crypto world really is pandemonium. But then I ran across a term that's taken hold in the industry that left me shaking my head. And, really, there is value to having everyone in this community really appreciate what a mess things are. So here's the news that I encountered, which is interesting in itself, which concludes with this new bit of terminology:

So last Friday, June 7th, approximately $126 million worth of crypto-assets were mysteriously transferred from the accounts of cryptocurrency platform Multichain in an apparent hack, according to blockchain security firms PeckShield, SlowMist, Lookonchain, and CertiK. "Multichain" is sort of a meta platform which interconnects different blockchain platforms

allowing users to exchange tokens. As a result of this incident it was shut down to investigate the incident. At the moment, neither Multichain nor any blockchain experts know exactly what happened. CertiK believes some of the platform's private keys were compromised—although this theory has not been confirmed by anyone else.

Fortunately, due to quick intervention by platforms like Circle and Tether, $65 million of those $126 million in funds which were moved from Multichain wallets were frozen pending the determination of what happened.

This is the second time this year that Multichain has suspended operations. It also halted trading at the end of May. At the time, the company said it couldn't perform server maintenance because its CEO had gone missing, and they didn't have access to the entire platform. The rumor was that its CEO was detained by Chinese authorities. Multichain never indicated whether its CEO has rejoined the company.

If the recent incident is confirmed, this would be the company's third hack in the past two years. It lost $3 million in January 2022 and another $8 million in July 2021.

And then here's the way the piece concluded which made me just shake my head:

“Blockchain experts aren't ruling out a rug-pull either.” And I thought, a what? A rug-pull? Yep. A rug-pull is when a crypto-platform's developers run away with the money themselves. They "pull the rug out from under their own operation.

So we have an industry that is so unstable that the term “rug-pull” has been coined as a shorthand to mean *“the bank's vault got all filled up with value... and then the bankers just decided to take it”* — because, you know, why not? Wow.

“Avast, ye Matey”

You know, Leo, how we were just observing that neither one of us is generally a big fan of attorney-enriching class-action lawsuits. But I stumbled over some news that really makes you shake your head.

The Netherlands has a foundation known as the CUIC for “Consumers United in Court.” This CUIC Foundation has filed a class-action lawsuit against the quite well known security firm Avast. CUIC alleges that from its special position in its users PC's, Avast collected data on its users which it sold to online advertisers without its users' knowledge and consent.

Everyone who used Avast's A/V products or browser extensions between May 2015 and January 2020 had their data collected and sold. Wow.

The list of products includes Avast Online Security, AVG Online Security, the Avast Secure Browser, the AVG Secure Browser, and the AVG Online Security extension. The collected user data was sold through a US-based subsidiary named Jumpshot. Avast shut down Jumpshot in January 2020 after its data trading practices were exposed.

Much as I generally despise class action lawsuits, in this case I hope that the proverbial book gets thrown at them. This is a deep betrayal of trust. And it's sad! Avast has been around for 28 years, since 1995, near the beginning of the PC era ... and it's currently in use by 435 million users. It must have been that the proceeds from collecting and selling data on the habits of that many individuals was just too much for them to resist. I wouldn't mind seeing them put out of business over this. Essentially, this was commercial spyware running in the PCs of those 435 million users.

China's OpenKylin v1

We've been observing that the era where it was feasible to actually create a new operating system from scratch has long since passed. The BeOS occurs to me as a late entry that reached some level of maturity, adoption and critical mass. But that may have been the last one to try. There are also various small hobby efforts that never get very far. And with Linux being open source, and with an incredible amount of effort already having gone into it, that's where any sane project would start today And, sure enough, that's exactly the conclusion that China came to.

Recall that China announced that they were going to be replacing the West's Windows with "their own" operating system. And "their own" is in quotes since begin sane, it's actually Linux under the very quite attractive covers: https://news.itsfoss.com/openkylin-linux-os/

So it's called OpenKylin and after many years it's finally at v1. It's based on the Linux 6.1 kernel and is available for x86, ARM, and RISC-V ("risk five") architectures. More than 200 companies, 74 special interest groups and 3,000 developers contributed to this new operating system, and it does have an English language option.

TootRoot!

I might not normally mention an extremely severe CVSS 9.9 vulnerability in Mastodon, except that when you learn that the vulnerability has been named "TootRoot" you really have no choice but to talk about it.

So, the Mastodon project has fixed the critical rootin tootin "TootRoot" vulnerability which — and get this, it was bad — allowed bad guys to commandeer any Mastodon server simply by posting a Mastodon **toot** containing a malicious multimedia file attachment. As we know, that's not an easy thing to defend against. The good news is that once the trouble had been discovered, everything proceeded properly. Patches were released days before the details were published. TootRoot was discovered by security researchers at Cure53 because they were conducting a security audit at the behest of, get this… Mozilla. In total, Cure53's audit discovered four security flaws. Oh... and did I mention that the now extinguished vulnerability was called TootRoot?

Firefox 115

Those of us who are using Firefox will find that we're now using 115. Since one of my two primary workstation machines is still running Windows 7 and running perfectly, I also received the notice that this would be the final release of Firefox for that machine, other than security

updates. That's fine since security updates is all I really need anyway. Edge and Chrome both gave up on me months ago.

The somewhat controversial new feature in Firefox 115 is Mozilla's declared ability to remotely prevent arbitrary extensions to run on arbitrary websites. For some reason this has upset people, which makes no sense to me. I cannot imagine that Mozilla's motives would ever be anything less than pure. So if they know something I don't, about some extension I'm running, and how some evil website I might mistakenly venture into is able to abuse an extension I'm running... then by all means shut it down and thank you very much.

In their explanation of this, Mozilla wrote:

> *Mozilla maintains an open ecosystem for add-ons, which gives developers many choices in how they create, use, and deploy their work. This same openness also provides malicious actors more opportunities as well. While Mozilla can identify and block malicious add-ons discovered through tooling, reviews, or user reports, this is not always enough. Firefox version 115 introduced Quarantined Domains to protect user privacy and security when we discover significant security issues presented by malicious actors. This feature allows us to prevent attacks by malicious actors targeting specific domains when we have reason to believe there may be malicious add-ons we have not yet discovered. Users can also control this behavior for each add-on in the Add-on Manager (about:addons) starting with Firefox version 116. We will be further improving the UI for users in future releases.*

Okay. So 115 lacks some UI for this, but that's coming in 116 (which I won't get on Windows 7, but I don't care because I'm happy to leave this in Mozilla's far more focused hands.

An example of an extension developer who drew some attention over this is a guy named Jeff Johnson whose posting carried the headline *"Firefox 115 can silently remotely disable my extension on any site."* Yes, by design. And why **would** they if it wasn't for the protection of the users of your extensions? Anyway, Jeff begins his grumbling by writing:

> *Firefox version 115.0 was released on July 4, but I'm not celebrating. I'm concerned about a new "feature" in the release notes.*
>
> > Certain Firefox users may come across a message in the extensions panel indicating that their add-ons are not allowed on the site currently open. We have introduced a new back-end feature to only allow some extensions monitored by Mozilla to run on specific websites for various reasons, including security concerns.
>
> *For various reasons. That's quite uninformative and mysterious.*
>
> *I'm all in favor of giving users control over which extensions are allowed to load on which sites. Safari already has this feature on both macOS and iOS. My concern is not about user control—little of which even exists in Firefox 115, as I'll show later—but rather about the remote control that Mozilla has now given itself.*

As we noted, more control is coming in 116 but, again, who cares? Jeff's posting goes on at

some length and I've left a link to it in the show notes for anyone who wants to see more about what's annoyed him: https://lapcatsoftware.com/articles/2023/7/1.html

Remember that Mozilla is in the driver's seat here. They must review and examine and digitally sign any extension before Firefox will even consider running it. And they also maintain and Firefox enforces a formal black list to disallow any previously signed extensions that are later found to be hazardous. So all they're doing is adding an additional level of granularity by supporting the intersection of extensions and web domains. They must have seen a need and, again, this seems all for the best.

Did Russia Disconnect?

As we've previously observed, Russia has been making noise about their own RuNet — which is what they call the portion of the Internet located within their territory. It appears that Russia now has a law requiring Roskomnadzor to perform an annual test disconnection of the Internet for the purpose of verifying its stability as a freestanding network. And they are claiming that happened in the very early morning last Wednesday, July 5th. But people in Russia are doubtful since.

Someone named Natalia Krapiva Tweeted:

> *Last night, Russia tested disconnecting itself from the global internet. On June 5, around 2-4am Moscow time, authorities tested the Sovereign Internet system which led to disruptions of various websites & government infrastructure services . Russian railroad services and food safety systems were reportedly disrupted after the Sovereign Internet testing on the morning of July 5.*

She then links to a Moscow Times article about this. But someone named Oleg Shakirov quoted her Tweet, replying:

> *Please don't take Russian (or anyone's) claims about testing 'disconnection from the global Internet' at face value. Always remember about incentives within the system to exaggerate one's work. There are plenty of reports that there were no universal break in connectivity. There is no real evidence that the drills caused disruption of Russian Railroads and the agriculture regulator. I mean this in a literal sense: there is no evidence cited in this Moscow Times piece. As I tweeted earlier, RZD problems started before the exercise and continue today; most likely caused by pro-Ukrainian DDoS-attacks.*

Earlier Oleg had Tweeted:

> *The website & app of Russian Railrods have been disrupted for 4 days. Customers are told to purchase tickets offline. According to RZD, this is due to hacker atracks. The problem persists today despite yesterday's statement. IT Army of UA implied its role.*

And finally, someone else (Sig. Ug. / @ug_sig) added:

Yes, a quick look at the sites that monitor Internet plumbing did not reveal any great disruptions during the time period identified in the reporting. One would have expected to see massive routing drops and other disconnections, but I didn't see any of that.

Disconnecting from the Internet in any meaningful way — for the purpose of isolating all of Russia from the West (have you seen how large Russia is?) — would indeed be visible to anyone monitoring the Internet's operation, it seems likely that if anything was done at all, it was some token gesture that, in fact, fell far short of anything that anyone would call a full disconnection. And I had, elsewhere, seen reports that pro-Ukrainian forces were taking credit for shutting down Russia's rail system with DDoS attacks.

Use some honey if you want to catch some flies.
Get a load of this one! Trustwave's SpiderLabs group carried out a 6-month experiment creating a globally diverse honeypot network. They wrote:

To obtain a better perspective of attacks worldwide, Trustwave has implemented a network of honeypots located in multiple countries across the globe. By distributing honeypots in such a manner, we can gather a reliable set of information on the methods and techniques used by attackers and their botnets.In our pursuit to explore the current threat landscape, we established a honeypot sensors network across six countries: Russia, Ukraine, Poland, UK, China, and the United States.

So, basically, they established a widely geographically distributed set of listening posts across the Internet, collecting all of the arriving packet traffic. And what did their 6-months of listening reveal? They found that fully 19% of traffic that probed their test honeypot network was malicious. And of that malicious traffic, 95% came from IoT botnets trying to locate and exploit new devices into their botnet.

Cryptocurrency losses
I know that I've already talked about the crazy losses being visited upon various cryptocurrency exchanges and services. But in preparing today's news I ran across a more comprehensive summary that I also wanted to share. A week ago Monday July 3rd, three days after the first half of 2023 ended, the blockchain monitoring group *"SlowMist"* published their *"2023 Mid-Year Blockchain Security and Anti-Money Laundering Report."* They start off explaining:

This report delves into blockchain ecosystem security, summarizing key security incidents and funds recovery status in the first half of 2023. It aims to help readers identify suspicious transaction patterns and behaviors by analyzing typical cases, and explore the anti-money laundering landscape within the blockchain ecosystem.

Their report is lengthy and detailed, but here are some high points:
More than $922 million worth of cryptocurrency assets were stolen in the first half of 2023

across a total of 185 security incidents. That dollar figure is less than half of what was lost during the first half of 2022, when hackers stole **$2 billion** worth of crypto across 187 incidents. So, the same number of incidents in both first halves of the year.

In total during 2022, "Chainalysis" reported the loss of more than **$3.8 billion** worth of assets. It's unclear why the first half of this year saw fewer funds stolen compared to last year. But no one believes that it's because crypto-platforms have become more secure. As we saw, the total incident count was 185 vs 187.

Nearly half of the funds stolen **this** year were taken from NFT, DeFi, and cross-chain bridge platforms, which lost a total of **$487 million** in 131 incidents. And the year's largest hack so far was the Euler Finance incident, where the platform lost **$197 million**. The hacker eventually returned most of the stolen funds in one of the ten incidents where attackers returned stolen crypto at all — usually in return for amnesty and a "bounty" award.

So it should be abundantly clear that the world is still a long way away from figuring out how to do any of this securely. It's difficult to see how participating in any of this is worth the risk.

International Consumer Data Transit

The world continues to struggle over issues of cross-border Internet consumer privacy and the monetization of consumer data.

The European Union has just ruled that what Meta is doing is illegal under the EU's GDPR. The European Court of Justice has ruled in a case between Meta and Germany's Federal Cartel Office, concluding that Meta's interpretation of the EU's GDPR regulation is illegal. The court sided with the German agency, which ruled in 2019 that Meta was bypassing GDPR privacy protections by taking data collected by various of its services without proper user consent and merging it behind the scenes. The aggregated data allowed Meta to continue tracking German and EU users to feed its advertising business. The court's ruling bars Meta from engaging in such behavior in the future.

Meanwhile, Sweden has just joined Austria, Denmark, France, and Italy in their growing crackdown on Google Analytics use. The Swedish data protection agency has fined two local companies for their use of the Google Analytics service and has recommended against future use of the tool. A €1 million fine was handed out to Swedish telco Tele2 and a €25,000 fine to local online retailer CDON. The fines were handed out because companies allowed Google Analytics to collect data on Swedish citizens which was then transferred to the US. And as I noted, Sweden thus becomes the fifth EU member state to fine or recommend against the use of Google Analytics.

Assuming that the EU's legislation holds up, and they have every right to create and enforce whatever legislation they choose, it looks like it's going to become necessary for our large multinational Internet service providers to establish and run fully independent facilities which are able to demonstrate complete autonomy for each national region. That won't be cheap or easy, and I imagine they're going to go kicking and screaming... but it looks like the future.

Apple's emergency update retraction

Yesterday, Apple issued and then quickly retracted an emergency update. Just a few hours after Apple posted and began pushing one of its new emergency RSR (Rapid Security Response) updates for iOS, iPadOS, macOS Ventura and Safari, the update was canceled and withdrawn.

What happened? According to reports posted to MacRumors forums, Facebook, Instagram, WhatsApp, Zoom, and other websites started producing warnings to their visitors about their updated Apple Safari browsers not being supported.

The updates repaired a WebKit 0-day vulnerability that was discovered while being under active exploitation — bringing Apple's patched 0-days count to 10 for the year so far. But get a load of what happened: the previous versions of the various iOSs were 16.5.1 (13.4.1 for Ventura). What Apple did, since this was meant to just be a quickie RSR WebKit patch, was to leave the primary version number unchanged at 16.5.1 and to simply append a lowercase 'a' in parenthesis (a). So, as a consequence of this change, Safari dutifully appended that same (a) to the end of its User-Agent Version string... and THAT – as was quickly revealed – for some reason deeply upset and confused the User-Agent string parsers being used by a number of quite popular web servers.

User-Agent: Mozilla/5.0 (iPhone;
CPU iPhone OS 16_5_1 like Mac OS X)
AppleWebKit/605.1.15 (KHTML, like Gecko)
Version/16.5.2 (a)
Mobile/15E148 Safari/604.1

So Apple quickly retracted this update to minimize its damage and has posted that they will shortly be releasing a 16.5.1 (b) update which we can assume will likely not be reflected in Safari's User-Agent string, or at least not in any way that will be upsetting any major websites.

Miscellany

Syncthing Revisited

A couple of weeks ago we talked about synchronized Synology NAS boxes, Syncthing, Sync.com and the free Windows app, @MaxSyncUp. Since I have changed what I'm doing, I wanted to quickly correct the record.

I soon recognized that the problem with using the @MaxSyncUp Windows solution was that, while it did provide bidirectional sync, it did not have an active agent running in the local Synology NAS. So it couldn't be proactively notified of any changes made there due to the inter-Synology synchronization. That turned out to be a problem.

Your mention, Leo, that there was a native means for running Syncthing without some messy Docker or other container encapsulation sent me looking and, sure enough, I found it. As you said, by trusting an additional (and very clearly trustworthy) source of Synology Add-ons, I found a native Synology build of Syncthing. So Syncthing is now running natively in each of my locations' Synology NAS's and on the various Windows machines in each LAN. So Syncthing is managing all local LAN file synchronization and performing local-only synchronization to the NAS's. And the NAS's are then performing their own inter-Synology synchronization using Synology's integrated synchronizing. So it's a very cool solution.

I thought I'd share how I'm handling that Internet exposure. Synology offers lots of "logon to your Synology NAS remotely" functions — to which I say "thanks, but no thanks." That presents too great a vulnerability that can go far too wrong if the bits hit the fan. As we know, "featuritis is what bites us." So I have ALL of that "join your NAS to the Synologyland community" stuff turned off. What I want for my NAS's is ZERO 3rd-party and ZERO public Internet exposure... and it's possible to have that:

At each location, a pfSense firewall is facing the Internet. As I've noted before, it's incredibly handy for performing static port mapping to bypass COX's consumer port filtering. But in this case I'm using another nifty feature of pfSense, which is DNS-driven firewall rule updating. In other words, pfSense's firewall rules can track IP changes through DNS.

Each of the pfSense firewalls opens a single port which can only be connected to by the other endpoint — explicit IP to explicit IP. But that means that each endpoint needs to know the other endpoint's IP. pfSense has a very capable and mature DynDNS client module. So I'm using a free DynDNS service to track the IPs of each endpoint for the other endpoint. As I noted, pfSense's firewall filters can be slaved to DNS lookups. So if either endpoint's IP should change, though that doesn't happen often with a cable modem, that local pfSense instance will note the change and its DynDNS client will update the DynDNS service with its new IP. The other end's firewall will then automatically update to allow a connection only from that new IP.

This solution is slightly less secure than the absolutely secure solution of manually configuring the IPs of each endpoint's firewalls. I could do that if it were absolutely necessary. But this protects me from the primary threat of having my use of Synology known in advance from an Internet scan and then being vulnerable to a targeted attack due to any suddenly discovered Synology vulnerability.

Closing the Loop

George Balogi / @GTBalogi

Hi Steve long time listener of the show and I owe a lot to you for all the IT knowledge I've gained. Thank you!! Had a question and was wondering if you might know. Is it possible to read odd file system disks like those found in enterprise level xerox machines. Trying to prove or disprove a theory of what is left on these drives.

It's definitely possible to look at and examine raw disk contents independent of any file system. Under Windows my favorite piece of Freeware is the "HxD - Freeware Hex Editor and Disk Editor." I have a link to it here in the show notes. If you can arrange to attach that drive to any Windows machine, HxD will allow you to "open" the drive without mounting any file system. And you then have all of the familiar Windows browsing tools with scroll bar, search and so on.

https://mh-nexus.de/en/hxd/

SKYNET / @fairlane32

I'm late watching last week's podcast but thanks for sharing about the chirping of the mystery "smoke detectors". Hilarious but not uncommon that you can't find it. ?? I've had that happen at home. You said you didn't have any smoke detectors. What about carbon monoxide detectors? We have both and when I can't find the one chirping I will take all the batteries out of all of them and start putting them back one by one until I find the one that starts beeping. Maybe that'll help

Stephen J. McCalley @Halukos

Long time SN listener. You probably already got the answer since I'm usually listening a day late but the military requires wearing the "Uniform of the day" while on duty. For most branches that is the working Battle Dress Uniform (BDU)/which is camo

.

Simon Zerafa / (@simonzerafa@infosec.exchange) @SimonZerafa

Simon Zerafa brings us our "Cringe of the Week":

@SGgrc / Attended a security conference that offers attendees the option to line-up for a help yourself lunch! However if that queue gets above a certain length, remaining delegates are provided a la carte dining instead. This is to prevent Buffet Overflows

tws / @twshaka

Hi Steve,
During a podcast from couple of months ago, you mentioned a favorite Authenticator app that you use, could you remind me what it was?? Had been using google Authenticator for many years but I do not like these new changes. Thanks so much for making a great podcast and of course spin right (looking forward to the new version)

And speaking of SpinRite...

SpinRite

SpinRite's first RTM release

Sunday evening I posted the 33rd Alpha release and noted that it was the first release candidate for the DOS component of SpinRite v6.1. SpinRite appears to be finished. That release contained SpinRite's new embedded FAQ, and the people who read through it found three hyphen characters that were not displaying correctly. But so far, everything appears to be holding.

This evening, while the newsgroup gang pounds on what we have, I will being the final work of updating SpinRite's Windows app with the new USB formatting InitDisk technology that I started out creating three years ago. I'll add that and bind the SpinRite DOS program into it, and once that's been thoroughly tested it will replace SpinRite v6.0 on GRC's website and v6.1 will have been released.

RTOS-32

Leo: Last Wednesday during Windows Weekly, something, I don't recall what now, caused you to note that for SpinRite's future I had chosen to finally leave DOS in favor of another operating system, for which I had purchased the source code. Because this is where I'm heading for the foreseeable future and because it's going to be a big topic for me, I wanted to explain a bit more about this.

The first exposure many of us had to Intel processors was IBM's PC. As we know, it contained Intel's 8088 then the 8086 and so on. But as we also know, Motorola was bidding to have their 68000 processor chosen and, in fact, that's what Apple used in their first Macintosh PC. To this day I still wish that's what IBM had chosen since it was a truly lovely device. But that's not what we got. My point is that Intel didn't create their processors for the PC -- the IBM PC project chose the already existing Intel processor family as the basis for its PC product line.

So that begs the question, what operating system did other early Intel x86 customers use who weren't building PCs? They were building elevators, trains and planes and automobiles, launching telecommunications satellites and installing commercial HVAC systems. These things had no screens or keyboard. They were known as embedded processors and they did their work mostly without any recognition of any kind.

It turns out that in many cases those companies chose to use a real time operating which had been created by a German named Peter Petersen. To give everyone an idea of who has always been using Peter's RTOS-32, the beginning of his alphabetical customer list read:

3M Company, Adaptec, Agilent Technologies, Airbus, Air Force Research Laboratory, Alcatel, AT&T Bell Laboratories, Audi, Bayer, Blaupunkt, BMW, Boeing, Bosch Telecom, Carl Zeiss, Carnegie Mellon, CERN, Daimler, Deutsche Telekom, Digital Research, Dow Chemicals, DuPont, Ericsson Mobile, ETH Zürich, Ford, Fuji Photo, Goodyear, Hewlett Packard, Honeywell Aerospace, IBM's Research Division, JPL, Lawrence Livermore National Laboratory, Leica.

Anyway, you get the idea. And to that list we can now add Gibson Research Corporation. I suppose it could be thought of as an industrial operating system as opposed to a consumer operating system.

So this OS predates everything. He's been licensing it to the who's who of industrial Intel processor users around the world since the early 1990's and as Peter explained in his "going out of business letter" at the end of last year, the trouble was that it was a finished product. It perfectly interfaced his customer's provided code to their Intel-based hardware. And what's more, over the years he ran out of bugs because it was done. It was complete. It was perfect. There was nothing left to fix. So his faithful customers of decades had stopped paying for annual maintenance because what they never had any problems that needed fixing.

So, I purchased it, lock stock and barrel. I own the result of those 30 years of embedded operating system software refinement. I have the source code so that I can customize it as needed to address any specific SpinRite need, and also because there will be some mass storage technology that follows NVMe that I'm going to want SpinRite 9 or 10 to be able to handle.

Needless to say, I've been patiently biding my time, working to get SpinRite 6.1 finished so that I can get started on SpinRite 7 and begin a whole new adventure. **I can't wait!**

http://www.on-time.com/customers.htm

Rowhammer Indelible Fingerprinting

At the end of February next year, during NDSS, the Network and Distributed System Security Symposium being held in San Diego, the recently completed work of a team of six UC Davis researchers will be presented. Their paper is titled: *"Centauri: Practical Rowhammer Fingerprinting"* – Another title might have been "The creation of a DRAM SuperCookie."

Here's how they described their findings in the paper's Abstract:

> *Fingerprinters leverage the heterogeneity in hardware and software configurations to extract a device fingerprint. Fingerprinting countermeasures attempt to normalize these attributes such that they present a uniform fingerprint across different devices or present different fingerprints for the same device each time. We present Centauri, a Rowhammer fingerprinting approach that can build unique and stable fingerprints even across devices with homogeneous or normalized/obfuscated hardware and software configurations. To this end, Centauri leverages the process variation in the underlying manufacturing process that gives rise to unique distributions of Rowhammer-induced bit flips across different DRAM modules. Centauri's design and implementation is able to overcome memory allocation constraints without requiring root privileges. Our evaluation on a test bed of about one hundred DRAM modules shows that Centauri achieves 99.91% fingerprinting accuracy. Centauri's fingerprints are also stable with daily experiments over a period of 10 days revealing no loss in fingerprinting accuracy. We show that Centauri is efficient, taking as little as 9.92 seconds to extract a fingerprint. Centauri is the first practical Rowhammer fingerprinting approach that is able to extract unique and stable fingerprints efficiently and at-scale.*

This is some brilliant work. And it's one of those discoveries that's immediately obvious in retrospect. Rowhammering is a subject that has come up over and over through the years. What we know about it is that in order to obtain maximum storage densities from our system's dynamic RAM, the storage cells have been shrunk and the number per unit area has grown. And effectively any margin for error has been deemed too costly and has been eliminated. What we wind up with is main system RAM memory that can be pushed over the edge through deliberate abuse. Where in this case "abuse" amounts to just hammering on one memory address which can, with distressing success, cause adjacent memory bits to spontaneously flip from a 1 to 0 or 0 to 1.

This breaks all of the rules since there's nothing inherently wrong with reading one memory address over and over. So, through the years we've seen this unwelcome consequence of too much data being stored in too little space being very cleverly leveraged in many different ways to breach the protective hardware-enforced barriers isolating virtual machines from each other or imbuing unprivileged processes with fill root kernel privilege.

And now, today, we have another consequence of this aggressive hardware engineering: The observation that tiny variations in the manufacturing of today's DRAM chips allows them to be uniquely identified in the field, thus enabling indelible fingerprinting. Where previous Rowhammer researchers needed to search physical DRAM to find a location whose bits could be flipped, these UC Davis researchers realized that identifying the exact location of such flipping, and of which bits were flipped in which direction, produced that that could be used to uniquely

and indelibly identify **one specific piece of DRAM from the multitude** that would never change throughout the service life of that storage device. Here's how they frame their accomplishment in somewhat greater detail:

> *In this work, we investigate a stronger threat model where a fingerprinter aims to extract unique and stable fingerprints for devices with identical hardware and software configurations over extended periods of time. To this end, we aim to capture fundamental differences in the physical properties of the device's hardware as unique fingerprints. Our key insight is that a fingerprinter may be able to extract fingerprints from inherent differences that arise as a result of process variation in the hardware (CMOS) manufacturing process. As users seldom modify their device hardware, these fingerprints remain stable, as long as they account for differences resulting from process variation in the same hardware. While prior research has explored variations in internal clocks, GPUs and CPUs, we are the first to successfully leverage memory (DRAM) for fingerprinting.*

Now that we have some grasp of the basic idea, as always, the devil is in the details and those details turn out to be quite interesting. Suffice to say, this was not simple to accomplish. Here is some of what went into making this technology work:

> *We leverage Rowhammer to extract fingerprints by capturing the side-effects of process variation in memory modules. At a high level, "hammering" a memory row (in other words, repeated read or write operations in a short time interval) results in bit flips in adjacent memory rows. In this paper, we show that the pattern of bit flips due to Rowhammer can be leveraged to build a fingerprint. We also show that the pattern of Rowhammer bit flips is sufficiently unique and stable to build a reliable fingerprint for the population of computing devices (billions of devices). To build intuition, we visualized the distribution of bit flips produced by executing Rowhammer at the same locations on two identical DRAM modules1 at two different points in time. And the results looked promising — the distribution of bit flips was reasonably similar on the same DRAM modules at different points in time while being noticeably different across the pair of DRAM modules. So that was our starting point.*
>
> *Centauri is a practical Rowhammer-based fingerprinting approach that exploits bit flip distributions to extract highly unique and stable fingerprints even among homogeneous devices with identical software and hardware configurations over an extended period of time. Centauri overcomes three main challenges that make it then practical for fingerprinting:*
>
> *First, the bit flips triggered by Rowhammer are non-deterministic (in other words, hammering the same location does not flip the same set of bits). Thus, a fingerprinter has to account for this non-determinism to extract stable fingerprints. We identify certain practical scenarios that exacerbate this non-determinism where comparing set similarity to match fingerprints falls short. With Centauri, we hammer the same locations multiple times to extract a probability distribution of bit flips as fingerprints. We then compare the divergence of these distributions that leads to better re-identification of devices even where there is a drastic difference in the set of bits that flipped.*
>
> *Second, fingerprinters are constrained by the abstractions provided by the operating system to allocate memory. These abstractions provide limited access to contiguous physical memory and hide information about their allocation on the DRAM. Without root privileges, these constraints prevent fingerprinters from trivially tracking the location of bit flips to fingerprint devices. We use the insight from our measurement study that the distribution of bit flips in*

contiguous 2 MB chunks of memory is unique and persistent to overcome this challenge. Armed with the insight, we sample enough 2 MB chunks to guarantee access to the same chunk for fingerprinting. [That's quite clever and a bit chilling since it means that this could actually be accomplished without privileges.]

Third, memory modules implement mitigations against Rowhammer, such as Target Row Refresh (TRR). While prior research has demonstrated ways to craft hammering patterns to bypass TRR, they provide limited insights towards operationalizing them to trigger bit flips at scale. Centauri systematically identifies effective patterns for at-scale fingerprinting using Rowhammer.

We then evaluate Centauri on a set of 98 DIMMs across 6 sets of identical DRAM modules across 2 major DRAM manufacturers. Centauri produces high entropy (many significant bits of fingerprinting precision) with a highest fingerprint accuracy of 99.91% corresponding to a precision of 100%, and recall of 97.06%. Centauri also demonstrates high stability with daily experiments to extract fingerprints from the same devices over a period of ten days without any degradation in fingerprint accuracy. Our experiments show that Centauri only suffers a minor loss in accuracy of 0.9% in presence of external factors that are not under the control of fingerprinters but affect the distribution of bit flips (such as the CPU frequency). We also investigate the trade-off between the accuracy of Centauri's fingerprints against the efficiency of Centauri's approach in terms of the time taken to extract fingerprints. Centauri is able to extract a fingerprint in as little as 9.92 seconds, reducing the overhead by more than 95.01% while degrading accuracy by just 0.64%.

Therefore, our key contributions include:

1. *Practically extracting highly unique and stable fingerprints using Rowhammer: We practically demonstrate Centauri on the largest scale of DRAM modules in current literature.*

2. *Handling non-deterministic bit flips: We handle non-deterministic bit flips by hammering the same memory chunks multiple times and using the divergence between probability distributions of bit flips to re-identify devices.*

3. *Overcoming memory allocation constraints: We overcome memory allocation constraints by devising a novel sampling strategy that guarantees access to the same chunk of memory for fingerprinting.*

4. *Operationalizing bypass techniques for Rowhammer mitigations: We bypass Rowhammer mitigations by identifying effective hammering patterns that can trigger bit flips at-scale.*

Their paper is 16 pages and I just shared an edited-down summary from the first page and a half. Through the balance of their document they proceed to explain in absolutely complete detail how they pulled this off and how anyone else could too. Anyone who is sufficiently skilled, who wanted to implement an indelible tracker for anything that uses DRAM, now has all of the information required to duplicate their technology. Rowhammer strikes again.