Security Now! #619 - 07-11-17

All the Usual Suspects

This week on Security Now!

This week we have all the usual suspects: Governments regulating their citizenry, evolving Internet standards, some brilliant new attack mitigations and some new side-channel attacks, browsers responding to negligent certificate authorities, specious tracking lawsuits, flying device jailbreaking, more IoT tomfoolery, this week's horrifying Android vulnerability, more Vault7 CIA Wikileaks, a great tip about controlling the Internet through DNS... and even more!

In other words, all of the usual suspects!

(And two weeks until our annual BlackHat exploit extravaganza!)

Our Picture of the Week

Security News

Last Thursday the W3C finally decided to add the EME (Encrypted Media Extensions) to the formal HTML5 specification.
Not everyone was happy. Or, as Techdirt's Mike Masnick wrote the next day: Tim Berners-Lee Sells Out His Creation: Officially Supports DRM In HTML
https://www.techdirt.com/articles/20170707/15544137737/tim-berners-lee-sells-out-his-creation-officially-supports-drm-html.shtml

MASNICK: "For years now, we've discussed the various problems with the push (led by the MPAA, but with some help from Netflix) to officially add DRM to the HTML 5 standard. Now, some will quibble with even that description, as supporters of this proposal insist that it's not actually adding DRM, but rather this "Encrypted Media Extensions" (EME) is merely just a system by which DRM might be implemented, but that's a bunch of semantic hogwash. EME is bringing DRM directly into HTML and killing the dream of a truly open internet. Instead, we get a functionally broken internet. Despite widespread protests and concerns about this, W3C boss (and inventor of the Web), Tim Berners-Lee, has signed off on the proposal. Of course, given the years of criticism over this, that signoff has come with a long and detailed defense of the decision.

There are many issues underlying this decision, but there are two key ones that we want to discuss here: whether EME is necessary at all and whether or not the W3C should have included a special protection for security researchers.

Mike doesn't want this at all. But my feeling is that it was inevitable. The alternatives were (1) no access to protected copyright content, (2) browser add-on plug-ins to implement proprietary protections, or (3) a separate custom application published by each provider. Mike feels that the open purity of the Internet is broken by this. But he must have a different Internet in mind than the one we've all been using... because it's already a huge mess, and it's far from pure.

Where I absolutely DO agree 100% with him, though, is in the utter necessity of allowing for and protecting providing bona fide security researchers to freely study and strengthen implementations without fear of being slapped with bogus lawsuits under the DMCA.

On this, the W3C says:
We recommend organizations involved in DRM and EME implementations ensure proper security and privacy protection of their users. We also recommend that such organizations not use the anti-circumvention provisions of the Digital Millennium Copyright Act (DMCA) and similar laws around the world to prevent security and privacy research on the specification or on implementations. We invite them to adopt the proposed best practices for security guidelines (or some variation), intended to protect security and privacy researchers.

<sigh> How many times has everyone heard me pray that security researcher are allowed to continue their work. Every week this podcast overflows with evidence of the incredible value of unfettered security research.

So, our future web browsers will eventually support uniform Encrypted Media Extensions to which a per-provider CDM -- Content Decryption Module -- will attach to manage the on-the-fly decryption of copy-protected content.

The full disclosure/statement from the W3C:
https://lists.w3.org/Archives/Public/public-html-media/2017Jul/0000.html

Introducing OpenBSD's KARL:
We have ASLR - Address Space Layout Randomization, and
KASLR - Kernel Address Space Layout Randomization
Now, exclusively, OpenBSD introduces KARL: Kernel Address Randomized Link.

Starting with the recent v6.1 OpenBSD snapshot, every freshly booted kernel initiates a background process to "randomly relink and rebuild" a new kernel which, once finished, will be used at the next system boot.

Rather than randomly relocating large chunks of user or kernel space code at load time, which, as we know, suffers from problems of low granularity, OpenBSD's KARL is continually recreating an entirely unique kernel for use at every subsequent boot.

In the past we've discussed how ASLR and KASLR can be bypassed by any function which leaks its location. For example, a function might be asked to obtain some information which is held in an internal buffer, so it returns a pointer to that buffer. Now the caller knows where that buffer is, and where that buffer is relative to every other function within the same large module of code. Thus, even with ASLR, once any function has leaked its location, every other function -- and even snippets of known useful code -- that is co-resident in the same block can also be located. And, in practice the ASLR blocks are large and few.

But KARL changes this so that every function within every block occurs in a random order. So now even if a careless function were to leak its own location, no additional useful information could be inferred. Every kernel image would be unique.

The Linux team is jealous and will be looking at "borrowing" the idea... so in the not-to-distant future, this could become a powerful attack-resisting technology for the most popular family of open source operating systems.

Apple's Bug Bounty program -- one year later:
BlackHat 2017 is coming up at the end of the month.
It was one year ago, during BlackHat 2016 that Apple's head of security, Ivan Krstic, announced with a great deal of fanfare that Apple would finally be joining all other large publishers, including Microsoft, Google and Facebook, as well as countless smaller firms, in offering cash payout rewards for reproducible attacks.

Ivan's BlackHat presentation slide detailed five broad categories of bugs and their payouts:

- $200,000 would be paid in return for the disclosure of any flaws in iOS's "Secure boot firmware components";

- $100,000 for a vulnerability allowing the extraction of confidential material protected by the secure enclave processor;
- $50,000 for the execution of arbitrary code with kernel privileges or for unauthorized access to iCloud account data on Apple servers... and, finally,
- $25,000 for demonstrating access from a sandboxed process to user data outside of that sandbox.

Now here we are, nearly one year later approaching the subsequent BlackHat conference... and by any objective measure Apple's bug bounty program has been an utter failure.

Why?? Motherboard did some research. They promised well known exploiters anonymity, and the same for others who were bound by Apple NDA (non-disclosure agreements). And everyone told the same story: Apple bugs are so rare and so difficult to find... and are thus so valuable that no one who is looking for a payday from their efforts would turn an uncovered high-quality exploit over to Apple.

Apple doesn't pay enough. The going rate on the grey market for a multi-exploit iOS jailbreak is $1.5 Million US dollars. and even second-tier exploit purchasers will pay half a million US dollars for similar exploits.

Apple's gesture was nice, but they're not competitive, even without their own sandbox.

BugCrowd:
- https://www.bugcrowd.com/bug-bounty-list/

Google to Fully Distrust WoSign/StartCom SSL Certs in Chrome 61
The wheels turn slowly, but turn they eventually do...
Two years ago, in July of 2015, in a significant breach of certificate authority policy, a user of WoSign's poorly-designed free certificate service obtained a certificate for the entire Github root domain after proving control over a subdomain. As we know, Github's architecture gives individuals content control over their own Github subdomains.

In another implicit failure to abide by CA rules, WoSign either did not detect or did not report and also did not revoke the mis-issued certificate. So no effective auditing and disclosure was present or performed.

This breach went undisclosed until British Mozilla developer Gervase Markham discovered this 13 months later and 11 months ago. He posted to Mozilla's security policy mailing list, saying:

"In June 2015, an applicant found a problem with WoSign's free certificate service, which allowed them to get a certificate for the base domain if they were able to prove control of a subdomain."

What's more, the original discovery was made using ucf.edu certificates, then was retested under Github. The person who discovered the problem responsibly reported the issue to WoSign who revoked the reported certificates... but a year later the originally mis-issued ucf.edu

certificate had still not been revoked. This further demonstrated WoSign's woeful lack of responsibility.

During subsequent investigations in collaboration with Mozilla, Google conducted a public investigation and which uncovered several other cases of WoSign mis-issuance of certificates.

This podcast covered this worrisome news at the time and noted that Google's Chrome browser team was not taking this technical breach, nor all of the subsequent failures it implied, lightly.

Since last October, starting with Chrome 56, the browser has been gradually reducing its trust of certificates signed by WoSign and it's StartCom subsidiary (based upon the date of certificate issuance). With the upcoming release of Chrome 61, all conditional trust of WoSign and StartCom certs will be terminated. Based on the Chromium Development Calendar, this change should be visible in the Chrome Dev channel in the coming weeks, the Chrome Beta channel around late July 2017, and will be released to Stable around mid September 2017.

Meanwhile... Lets Encrypt: Wildcard Certificates Coming January 2018
https://letsencrypt.org/2017/07/06/wildcard-certificates-coming-jan-2018.html

Beginning next year, in 2018, LetsEncrypt will begin offering wildcard certificates for the first time. From is inception, LetsEncrypt has been carefully rolling forward, trying hard NOT to make any big mistakes with something as critical as fully automated certificate issuance. They've held off on the much-requested feature of issuing a single certificate for all of a domain's subdomains (*.example.com) until they felt sufficiently safe and secure in doing so. So in January, rather than requiring every subdomain to approve and obtain its own certificate, the single parent domain will be empowered to obtain a single certificate for all possible subdomains. Since the only thing LetsEncrypt's automation is verifying is domain control, these certificates will be and can only be the least rigorous form of DV - Domain Validation - certs. But that's all that any site needs for protecting, encrypting and authenticating its traffic.

Headlines: "Researchers Crack 1024-bit RSA Encryption in GnuPG Crypto Library"
Except... no... they didn't.
The term "Cracking Encryption" means something. And that WOULD be huge news if it were true. But it's not.

What "Cracking Encryption" does NOT mean is arranging to obtain the private key by exploiting an algorithmic implementation flaw in a particular library. That's not a good thing... but neither is it a "crack" of the encryption.

What DID happen is that an international group of eight crypto researchers used a flawed implementation of RSA to powerfully demonstrate that the underlying Libgcrypt crypto library needs to be fixed. This implementation flaw was used to leak private key information through a side channel. This L3 cache side-channel attack requires an attacker to run carefully designed forensics software on the same hardware where the private RSA key is being used, while it is being used.

ABSTRACT: It is well known that constant-time implementations of modular exponentiation cannot use sliding windows. However, software libraries such as Libgcrypt, used by GnuPG, continue to use sliding windows. It is widely believed that, even if the complete pattern of squarings and multiplications is observed through a side-channel attack, the number of exponent bits leaked is not sufficient to carry out a full key-recovery attack against RSA.

Specifically, 4-bit sliding windows leak only 40% of the bits, and 5-bit sliding windows leak only 33% of the bits.

In this paper we demonstrate a complete break of RSA-1024 as implemented in Libgcrypt. Our attack makes essential use of the fact that Libgcrypt uses the left-to-right method for computing the sliding-window expansion. We show for the first time that the direction of the encoding matters: the pattern of squarings and multiplications in left-to-right sliding windows leaks significantly more information about the exponent than right-to-left. We show how to extend the Heninger-Shacham (Henn-in-ger Sha-kam) algorithm for partial key reconstruction to make use of this information and obtain a very efficient full key recovery for RSA-1024. For RSA-2048 our attack is efficient for 13% of keys.

Facebook can track your browsing even after you've logged out, judge says

The Guardian reported last Monday that a judge had dismissed a lawsuit accusing Facebook of tracking users' web browsing activity even after they logged out of the social networking site.

The plaintiffs alleged that Facebook used the "like" buttons found on other websites to track which sites they visited, allowing Facebook to assemble detailed records of their browsing history. The plaintiffs argued that this violated federal and state privacy and wiretapping laws.

However, US district judge Edward Davila in San Jose, California, dismissed the case because he said that the plaintiffs failed to show that they had a reasonable expectation of privacy or suffered any realistic economic harm or loss.

In other words, the plaintiffs were annoyed, but that's not sufficient grounds for a such a lawsuit.

Davila said that plaintiffs could have taken steps to keep their browsing histories private, for example by using the Digital Advertising Alliance's opt-out tool or using "incognito mode", and failed to show that Facebook illegally "intercepted" or eavesdropped on their communications.

He said: "Facebook's intrusion could have easily been blocked, but plaintiffs chose not to do so." Davila also dismissed an earlier version of the five-year-old case in October 2015.

Davila wrote: "The fact that a user's web browser automatically sends the same information to both parties does not establish that one party intercepted the user's communication with the other."

The plaintiffs cannot bring privacy and wiretapping claims again, Davila said, but can pursue a breach of contract claim again.

It sounds as though the plaintiffs are unhappy that logging out of Facebook does not stop Facebook's tracking for their activities. I think they need a lesson in how the Internet works:

As we know, websites give browser a unique token (a cookie) which is subsequently returned anytime that browser makes any request for an asset from the, in this case Facebook domain. But that's entirely separate from logging in. Even if you NEVER were to login to Facebook, just by touching the Facebook domain, your browser would pickup a Facebook cookie which would then be used to begin assembling an anonymous profile as your browser.

You don't even need to visit the Facebook website -- ever -- since any contact with any webpage containing a Facebook LIKE button will initiate that browser connection to Facebook and, if by some miracle your web browser doesn't already carry a Facebook cookie... it will from then on... and the data gathering and profiling begins.

"Logging in" merely associates its Facebook cookie, which your browser is almost certainly already carrying, with the logged-on identity you provide. And subsequently logging off simply says "Show me the logon page next time I visit and don't let me do anything else until I log back on."

This week in “When Governments React”:
As the Internet continues to mature, one of our continuing threads in this podcast is keeping an eye on regulation both generally and with regard to encryption. In this regard, it's of interest that the Chinese government has recently instructed its Internet carriers to block access to personal VPNs by next February 2018.

As we know, VPNs can be used to hide and tunnel traffic past the view of a local ISP or, in the case of China, past the so-called Great Firewall which allows the Chinese government to regulate and censor the full content of the Internet so that those within its borders only have access to government-approved content. The use of VPNs has long been seen as a means to bypass those border protections.

From inside China, many foreign social-media sites are affected, including Facebook, Twitter, YouTube, and Instagram. Blocked news sources include the New York Times and the Wall Street Journal, along with sites such as Google Scholar. Seven years ago, Google’s reluctance to have its results censored led it to quit the country with the Chinese government later banning most of Google's services.

Chinese VPN services have been under pressure, and most Chinese VPN service clients have already disappeared from Android app stores.

Although China has previously issued edicts banning the use of VPNs, the latest development suggests that the Chinese government is getting much more serious. China's three major telecommunications providers -- China Mobile, China Unicom and China Telecom -- have been ordered to block all VPN usage by the start of next February. This will impact virtually every mobile customer in the country and it represents a significantly more comprehensive and aggressive approach than we've seen so far. Almost all internet users in China go online using services run by the state-owned carriers.

The question remains: Can this actually be practically accomplished?

Old-school VPNs used specific well-known ports which a border firewall could certainly block. But modern VPNs can use any port and can tunnel their traffic over either UDP or TCP.

For example, an external website could be established to accept a URL and display within its own page, the page of another website. So we can demonstrate that it's technically impossible to simply "block" all VPNs... or at least access to all banned content.

But this action is further raising the bar and making banned content less widely available and more difficult to acquire.

Believe it or not, "Drone Jailbreaking" is now a thing.
DJI, the maker of what is arguably the best line of commercial quadcopters -- especially useful for professional aerial photography -- weddings, bike races, watching Apple assemble their solar donut, etc. -- is locking down their drones against a growing army of DIY hackers who argue for freedom of flight.

The controversial DJI firmware enforces no-fly zones, altitude and airspeed controls which upset drone owners who chaff at such limitations. Moreover, online boards are full of reports where the restrictions appeared to be applied arbitrarily, restricting flight for no apparent valid reason.

So, hacking DJI's drone firmware has been on the rise, and DJI has begun fighting back by pushing firmware updates to Internet connected drones and proactively removing previously available vulnerable legacy firmware from their servers. This has, in turn, spawned an underground of firmware archives and "chop shops" for retrofitting "full flight freedom firmware."

In the United States, people jailbreaking their drones are operating in something of a legal grey area concerning a federal copyright law called the Digital Millennium Copyright Act. The Librarian of Congress, which administers specific exemptions to the law, has given wide latitude to tinkerers seeking to break through software locks for the sake of repair or restoring factory settings—it is currently legal to hack into tractors, cars, and cellphones, but it's not legal to jailbreak video game consoles. There is currently no specific exemption for drones, but DJI would have to bring a suit against its consumers to test this for sure.

"Broadpwn" Bug Affects Millions of Android and iOS Devices
Unpatched Broadcom Wi-Fi chips used in both Android and iOS devices are vulnerable to a bug that allows an attacker to execute code on their devices, without any interaction needed from the user... simply by being within radio range of any malicious WiFi access point.

Security researcher Nitay Artenstein discovered and nicknamed the Broadcom firmware flaw "Broadpwn", and tracked as CVE-2017-9417. He will be delivering a presentation about Broadpwn at this year's Black Hat USA security conference at the end of the month.

Artenstein responsibly disclosed the bug to Google, who included a fix for it in last week's July 5th, Android Security Bulletin. Although little public information is available yet, we know and

Artenstien as said that "Broadpwn affects millions of Android and iOS devices" that use Broadcom Wi-Fi chips to handle network communications.

The flaw is present in the firmware for the Broadcom BCM4300 family of Wi-Fi chips included in "an extraordinarily wide range of mobile devices" from vendors such as Google (Nexus), Samsung, HTC, and LG.

After being made curious, another Android security expert reversed engineered last week's July security patch to dig out more details about Broadpwn. He determined that the bug appears to be a heap overflow in the Broadcom firmware. The researcher said that the exploitation occurs when the user's device receives a WME (Quality-of-Service) information element with a malformed length from any connected network.

As I noted, exploitation does not require any user interaction. A victim needs only to enter into the WiFi range of an attacker's signal. Artenstein has confirmed that even connecting to a malicious network is not necessary.

Not surprisingly, Google's security bulletin last week rated Broadpwn as a "critical" severity vulnerability.

Broadcom's BCM4300 line appears to date back at least to 2005, so 12 years ago. It's unclear whether this flaw has been present since then, but "baseband OS firmware" tends not to change nearly as often as the OSes that run on top of it.

It's very clear that if this flaw were to be weaponized by any global state actor, and subject to any exploitation limitations, virtually any unpatched Android device could be compromised simply by getting a malicious portable WiFi access point within range of any targeted Android user.

We keep covering these very serious Android problems. In older and unpatched devices they are rapidly piling up and they have already far exceeded critical mass. Although targeted exploitation is unlikely to affect many of us, opportunistic exploitation on college campuses and in any public settings such as airports, hotels -- or the upcoming BlackHat conference <gulp!> -- could easily affect huge numbers of users.

The conclusion is simple: Android devices that are not being continually, rapidly and responsibly patched CAN NOT BE USED SECURELY.

At this time there is no information on the status of this bug for iOS devices. We'll see what Artenstein says in two weeks during his BlackHat talk.

A new form of clever selective whitelisting will soon be coming to Windows 10.
Known as "Controlled Folder Access", it will be debuting in September's "Fall Creators Update" and Win10 testers now have access to a preview of the changes which include this new controlled folder access feature.

CFA is designed to only allow specific apps to access and read / write to a folder. If enabled, the default list prevents unknown and unauthorized apps -- such as crytomalware -- from accessing

the desktop, pictures, movies, and documents folders.

In the new version, CFA can be enabled in the "Windows Defender Security Center" dialog. There's a "Protected Folders" option and an "Allow an App through Controlled Folder Access."

I think it's a brilliant tradeoff. What's not clear is how this will work interactively. Whether or not it will be enabled by default, whether it might start off being disabled and enabled in the future once Microsoft gains more experience with it. And what programs will be permitted, etc. But, as we've been saying here, whitelisting -- default block and selectively permit -- i the only way to be truly secure. It is the future... and it's nice to see that Microsoft is recognizing this.

The LEAST this represents is a very useful power-tool for power-users. I say Bravo to Microsoft on this one.

Smart home gadget ends a violent dispute by calling police
https://www.engadget.com/2017/07/09/google-home-calls-police-on-violent-dispute/

Two days ago, Engadget reported a bit of somewhat bizarre IoT news: That a smart home voice-response device was responsible for ending a violent dispute by calling the police. Although the device was first reported to be a Google Home device, that was later corrected.

Police in New Mexico reported that a smart home device intervened in a domestic violence incident by calling 911. When Eduardo Barros asked "did you call the sheriffs?" as he threatened his girlfriend with a gun, during a fight, the device interpreted it as a request to call emergency services. The 911 call responders overheard the altercation and called both negotiators and a SWAT team, who arrested Barros over assault, battery and firearms charges after a stand-off.

Barros' girlfriend was hurt in the altercation, though police contend that the situation would almost certainly been much worse. County Sheriff Manuel Gonzales believes that the command "possibly helped save a life," including that of the girlfriend's daughter (who was thankfully unharmed).

I know from first hand experience that unintended sounds -- especially loud sounds -- can cause the command discriminators in these devices to misfire. For some time, I had an Amazon Echo device in my living room listening, with me, to my whole room theater. And from time to time, when nothing to my ear sounded like its wake up trigger word, the device's blue ring would suddenly illuminate, scan around a bit, then go back to sleep.

So it's entirely believable that this could happen upon hearing a shouted "did you call the sherrifs?"... especially if the girlfriend's name was... you know what. :)

Also at BlackHat in two weeks: Skype&Type
A group of researchers have developed and proven yet another side-channel attack. Forbes reported in advance of BlackHat that after sufficient training, which can happen in the background during and over a Skype call, a randomly-chosen password can be determined within seconds... using only information gleaned over the connection from the unique and distinctive sounds and timing of someone typing on a keyboard.

Another Wikileaks/Vault7 leak alleges a tool dubbed "OutlawCountry"
"OutlawCountry" is allegedly a CIA project that allowed the agency to hack and remotely monitor the network traffic of Linux-based machines. The project reportedly allows CIA hackers to redirect all outbound network traffic on the targeted computer to CIA controlled computer systems for exfiltration and infiltration of data.

OutlawCountry s a Linux kernel module which is loaded via shell access to the targeted system. It creates a hidden Netfilter table with an obscure name on a targeted machine.

However, the tool described is not without limitations, the leaked documents indicate that OutlawCountry v1.0 contains one kernel module for 64-bit CentOS/RHEL 6.x, and states that "this module will only work with default kernels." This suggests that the tool may have been developed for a specific targeted application where the intended target was known beforehand.

Firewalla: Complete Cyber Security Solution For Your Home
https://www.kickstarter.com/projects/firewalla/firewalla-complete-cyber-security-solution-for-you

- Doesn't appear to be over-selling what they can do.
- My rant two weeks ago about Cisco overselling "seeing into encrypted" connections which almost certainly reduced to using DNS metadata.
- DNS Thingy
    - https://www.dnsthingy.com/
    - David Redekop / Nerds On Site
    - ASUS routers, ClearOS or pfSense
    - $8/mo.
    - Time based, per-device restrictions, and so so much more!
    - Scan down their blog.

Next Week: A focus on "The Password Reset MitM Attack"
https://www.ieee-security.org/TC/SP2017/papers/207.pdf
Bruce Schneier: "This is nice work..."
Sharing just the juiciest bits until next week, from their Abstract: "We present the password reset MitM (PRMitM) attack and show how it can be used to take over user accounts. The attacker initiates a password reset process with a website and forwards every challenge to the victim who either wishes to register in the attacking site or to access a particular resource on it.

The attack has several variants, including exploitation of a password reset process that relies on the victim's mobile phone, using either SMS or phone call. We evaluated the PRMitM attacks on Google and Facebook users in several experiments, and found that their password reset process is vulnerable to the PRMitM attack.

Since millions of accounts are currently vulnerable to the PRMitM attack, we also present a list of recommendations for implementing and auditing the password reset process.

Miscellany

How secure is 256 bit security?
https://www.youtube.com/watch?v=S9JGmA5_unY
Only made one mistake early near the front of the video, where he said there was no better way then to guess, which would take, on average, 2^256 guesses. But we know that 2^256 unique guesses would GUARANTEE that one of them was correct. So 2^256 is the 100% chance count. 2^255 is the 50% chance count.
http://bit.ly/sha-256

TeeSpring SN Shirts:
https://teespring.com/tno-trust-no-one
https://teespring.com/the-s-in-iot

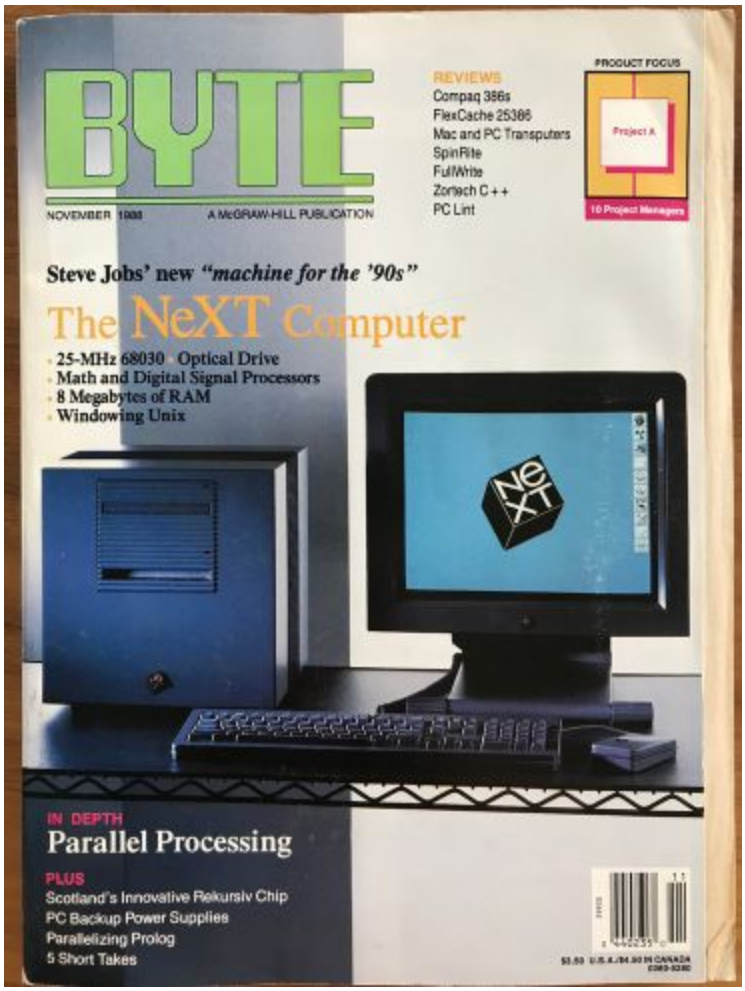
SpinRite

Bill Taroli (@btaroli)

- Sigh. 1 offline uncorrectible read error on my SSD. Time for a level 2 scan with @GibsonResearch @spinrite ! ??#securitynow

- Bill Taroli (@btaroli)
  Replying to @btaroli @GibsonResearch @spinrite
  @spinrite Awesome! SMART says drive is 36% useful life but it feels so much faster now! Full BTRFS defrag under 30% iowait now!

Chris Erickson (@chriserickson)
@SGgrc a fun review of SpinRite from Byte in November 1988.
https://twitter.com/chriserickson/status/881642031235440640



A Must-Use Utility

All I can say about the Gibson Research people is that they did their homework. The user interface is well thought-out and easy to use; all interaction is via an easy-to-navigate window system. The package comes with a hard disk and a 40-page user's manual that is more interesting for its historical content (how the authors of the package made all their discoveries about hard disks) than any other information. The program is so well put together, I found I seldom referred to the manual, anyway.

SpinRite is no 14-disk grand-slam C compiler, but you shouldn't underestimate its usefulness. If you have a PC with a hard disk drive that you spend most of your day relating to, and your heart sinks every time you see the drive's bad sector list, SpinRite is what the word "must" was invented for. ■

Richard Grehan is a BYTE senior technical editor at large. He can be reached on BIX as "rick_g."

Closing The Loop

Ian (@ianc)
@SGgrc Steve - listener for the past year. You often talk of prime #s. Other than basic knowledge of prime, why are they so powerful for crypto?

Acid Trucks (@AcidTrucks)
@SGgrc What are your thoughts on using router features like parental controls to shield the Internet from IOT devices (and vice-versa)?