Security Now! #754 - 02-18-20

The Internet of Troubles

This week on Security Now!

This week...This week we continue following the continuing agony surrounding this month's increasingly troubled Window Update. We example several significant failures which have befallen Windows 10 users after applying the month's "fixes" which have had the tendency of breaking things that weren't broken in the first place. We look at the danger presented by a very popular GDPR-compliance add-in for Wordpress sites, we look at an eye opening report about the stresses that CISOs are being subjected to, and also today's pilot test of Microsoft's new ElectionGuard voting system. We then touch on some SQRL and SpinRite news before taking a close look at two newly revealed IoT -- Internet of Troubles -- security worries.

Security News

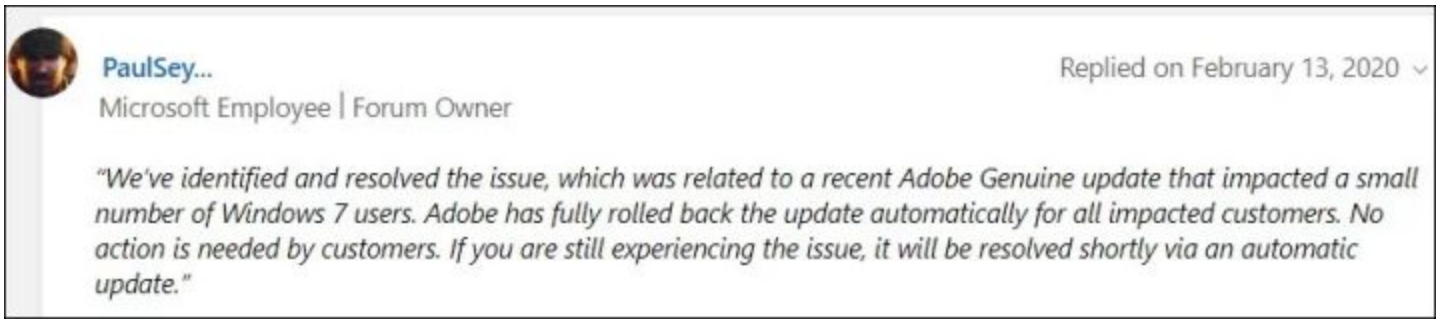
Following up on recent Windows Update Agony...

It turns out that the Windows 7 "You don't have permission to shutdown Windows" is not restricted to Windows 7. Windows 10 users are receiving the same permission denial.

As are many weird things that only affect some subset of Windows users, this turns out to be a subtle interaction with some other 3rd-party software. In this case the culprits are background services installed by Adobe's Creative Cloud. Bleeping Computer provided comprehensive coverage of this situation. They wrote:

> Windows 10 users are reporting being affected by a bug that prevents them from shutting down their devices without logging out first, an issue that we previously thought only Windows 7 customers were experiencing.
>
> On February 6th, Windows 7 users started reporting encountering "You don't have permission to shut down this computer."
>
> Since then, this error has been reported by several Windows 10 users too, one of them saying that he saw the error pop-up on a recently installed device running Adobe Creative Cloud, as initially reported by Günter Born.
>
> Others also confirmed that the issue was impacting their Windows 10 Home edition devices, as well as multiple Windows 10 installations in an environment were Windows 7 devices were also experiencing shut down issues.
>
> There are currently hundreds of user comments in Reddit threads as well as on the Microsoft Answers forums and Twitter. While the shutdown issues aren't as widespread on Windows 10 as they are Windows 7, all reports point at the same error and the same underlying bug being behind the problems.

Last Thursday, February 13th, a Microsoft employee posted: "We've identified and resolved the issue, which was related to a recent Adobe Genuine update that impacted a small number of Windows 7 users. Adobe has fully rolled back the update automatically for all impacted customers. No action is needed by customers. If you are still experiencing the issue it will be resolved shortly via an automatic update.
https://answers.microsoft.com/en-us/windows/forum/windows_7-start/you-do-not-have-permission-to-shut-down-on-windows/c08d3cee-1f87-498b-b8b8-7fb06a20755d?messageId=91921284-e293-4d4e-bcd0-265abb5d8f28



Finally, and helpfully, Bleeping Computer adds:

"While Adobe has already rolled back the update for Windows 7 customers, Windows 10 users are out of luck until the bug is also acknowledged for their platform and a fix is provided by either Adobe or Microsoft.

Until then, you can disable the Adobe services triggering the bug (Adobe Genuine Monitor Service, Adobe Genuine Software Integrity Service, and Adobe Update)..."

So, this was/is not Microsoft's doing and it won't be affecting Microsoft's massive installed base.

And… Win10's "One Button PC Reset" fails after KB4524244.
I'll just mention in case it might affect anyone that in another Patch Tuesday updated problem, one of the updates contained within the monthly rollup which was intended to resolve a security vulnerability that might be introduced by 3rd-party UEFI boot managers, turned out to kill the Windows 10 "One Button PC Reset" feature for both Windows 10 clients and servers. So Microsoft has pulled and has no plans to reissue that update. (This one was Microsoft's fault.)

As Microsoft explained it: "You might restart into recovery with 'Choose an option' at the top of the screen with various options or you might restart to your desktop and receive the error 'There was a problem resetting your PC'"

Any one experiencing this problem is advised to simply remove the update from their affected Win10 machine.

And, also… "The new disappearing User Profile problem" (Desktop and all user data)
Since this month's patch Tuesday, some unlucky Windows 10 users are reporting that another update included in this month's update bundle is causing problems. According to reports, a bug in the KB4532693 update is hiding user profiles and their respective data on some Windows 10 systems. The issue has been reported on Microsoft forums, Twitter, Reddit, and tech sites including AskWoody, Bleeping Computer, and BornCity:

Users are reporting that after installing the standard monthly update rollup, they can no longer view or access their original Windows 10 profile. In other words, according to these reports, after the update, users are left logged into a blank/default Windows 10 profile where =ALL= of their previous data is missing. This includes access to installed apps, desktop wallpapers, desktops files, downloads, and others. As you might imagine, it's quite unsettling.

Last Wednesday, Woody Leonhard tweeted:

> Multiple reports that the Feb Cumulative Update for Win10 (1903? 1909?) resets the desktop -- custom icons missing, background set to Windows logo -- and would not recognize the established logon account. Are you seeing the same? https://t.co/uZTcRqeEMN
> — Woody Leonhard (@AskWoody) February 12, 2020

There is some good news. Nothing was permanently deleted. The data is not gone, it's only been hidden. According to a report on Bleeping Computer, the bug is caused by a faulty KB4532693 installation procedure. The bug occurs when the Windows Update service creates a temporary profile during the installation procedure, then forgets to remove it after installing the update. When the update finishes, this temporary profile remains the one that users are logged into... And all their stuff is gone.

Reports indicate that the original user profile folders are still available on disk, but that they've been renamed with a .000 or .bak extension. While it's technically possible to recover these sequestered profiles by renaming and relinking them, the steps required are error prone and one wrong move might actually cause permanent data loss. So the best solution is to uninstall the faulty KB4532693. Multiple users have reported that removing the faulty update restores their old profiles.

All that said, it's obvious that not all Windows 10 users are impacted by this bug in KB4532693 and most will have no issues installing the update. All of my Win10 machines are current and this didn't happen to me. But it clearly IS happening to many people. So if you have (probably wisely) delayed applying this month's rollup, and IF this should happen to you, simply uninstall it and you should be good to go.

The popular "GDPR Cookie Consent" Wordpress plugin had a critical flaw

So, the plugin is named "GDPR Cookie Consent" and anyone with it installed in their Wordpress site needs to update it to version 1.8.3 or later with high priority due to a serious vulnerability.

As its name implies, it serves as an aid to make websites compliant with the EU's General Data Protection Regulation (GDPR), but until its recent February 10th update it has also been critically flawed in a fashion that, if exploited, could enable attackers to modify the site' content and inject malicious JavaScript into victim websites.

The plugin is popular on Wordpress sites with more than 700,000 active installations which, unfortunately, also makes it a ripe target for attackers. As I mentioned, the vulnerability affects GDPR Cookie Consent versions 1.8.2 and below. Last week, after the developer was notified of the critical flaw, the GDPR Cookie Consent plugin was removed from the WordPress.org plugin directory "pending a full review" according to the plugin's directory page. The new version, 1.8.3, was released by Cookie Law Info, the plugins's developer, last Monday, Feb. 10th.

The vulnerability stems from improper access controls in an endpoint used by the WordPress plugin's AJAX API. AJAX is a technology which allows JavaScript to independently initiate its own outbound HTTP connections for retrieving data for use by the script. For example, I used this in SQRL to create that magic where, without touching the login page, after authenticating with your

smartphone the login page of the site you're logging into spontaneously updates to show that you're logged-in. There's "AJAX" running on that page which periodically pings the site's server for a new URL for it to jump to. AJAX cannot ask for anything from anywhere. It's constrained by same-origin rules. But its use underlies many of the web applications we've grown to take for granted.

But in the case of this plugin, the "_construct" method within the plugin, which is used for initializing code for newly created objects, fails to implement the necessary security checks. Because of this, the AJAX endpoint, which is only intended to be accessible to administrators, also allowed visitors to perform a number of actions that can compromise the site's security.

The "_construct" method accepts three different values from the AJAX API. Two of them, save_contentdata and autosave_contant_data, can be leveraged for exploitation by an attacker.

The mistakenly accessible "save_contentdata" method allows administrators to save the GDPR cookie notices to the database. However, since this method is not checked, an authenticated user or a visitor can modify any existing page or post (or the entire website). And, as is so often the case with HTTP, it's possible to delete or change pages' content. Injected content can include formatted text, local or remote images as well as hyperlinks and shortcodes.

And, as if this weren't bad enough, the second method, "autosave_contant_data", is used to save the GDPR cookie info page in the background while the admin is editing it, by saving the data into a temporary "cli_pg_content_data" database field... but this skips any validation checks, and the lack of checks for this method could allow the injection of JavaScript code into the webpage. This code would then be loaded and executed each time anyone visits the "http://example.com/cli-policy-preview/" page.

So, not surprisingly, the researchers who discovered it urge WordPress plugin users to update immediately. I could not determine whether or not there's any auto-update mechanism, and my own Wordpress blog is not using that plug so I couldn't check myself.

Whoa! The average tenure of a CISO is just 26 months due to high stress and burnout
https://media.nominetcyber.com/wp-content/uploads/2020/02/Nominet_The-CISO-Stress-Report_2020_V10.pdf

Report: The vast majority of interviewed CISO executives (88%) report high levels of stress, one third report stress-caused physical health issues, and half report mental health issues.

We've touched on this in the past. "Information Security" -- cyber-security -- is still a relatively new thing. It's still seen as more of a necessary evil than as an obvious profit center like sales, marketing, or R&D. Not to mention that the High Priests of Information Security appear to speak in a strange language that makes no sense to the other C-Suite executives who establish budgets and schedules. As a consequence, most companies are still not ready to embed CISOs into their company culture and day-to-day operations.

Today, CISO jobs come with low budgets, long working hours, a lack of power on executive

boards, a diminishing pool of trained professionals they can hire, but also a constant stress of not having done enough to secure the company's infrastructure against cyber-attacks, continuous pressure due to newly arising threats, and little thanks for the good work done, but all the blame if everything goes wrong.

Through the years CISOs have often pointed out the problems with their jobs and the stress and damage they inflict. However, there has been no conclusive study to support broad assertions.

So, last November Nominet, an Internet and DNS security firm, independently surveyed 800 CISOs and executives from companies in the US and UK to explore and examine the topic and to determine how much of a role stress plays for CISOs across the industry. The survey's results paint a gloomy picture about one of today's most in-demand jobs. According to the report's data:

- 88% of CISOs reported being "moderately to tremendously stressed"
- 48% of CISOs said work stress has had a detrimental impact on their mental health
- 40% of CISOs said that their stress levels had affected their relationships with their partners or children
- 32% said that their job stress levels had repercussions on their marriage or romantic relationships
- 32% said that their stress levels had affected their personal friendships
- 23% of CISOs said they turned to medication or alcohol

Nominet said that: "Even when they are not at work many CISOs feel unable to switch off." As a result, CISOs reported missing family birthdays, holidays, weddings and even funerals.

"They're also not taking their annual leave, sick days, or time for doctor appointments, which contributes to physical and mental health problems."

Nominet said that while investigating the causes of CISO stress, they found that almost all CISOs were working beyond their contracted hours, by an average of 10 hours of extra-time per week -- for which they are not compensated.

Furthermore, many were under pressure from their boards. Almost a quarter of interviewed CISOs said boards didn't accept or understand that "breaches are inevitable" and said they'd hold them personally accountable for any security incidents.

Nominet said that 29% of CISOs who answered the survey said they'd be fired in the event of a breach, while 20% said they'd be fired anyway, even if they were responsible or not.

The answers explain why most CISOs don't last in their jobs more than 26 months, and why 90% of surveyed CISO were willing to take pay cuts if they could reduce stress levels.

Nominet said CISOs were willing to give up on $9,642 per year, on average, just to reduce stress levels and improve their work-life balance -- which many CISOs said they had problems with.

Nominet's numbers may seem staggering to someone looking in from the outside, but they come

as no surprise to someone working in the field. Although the Nominet study only surveyed high-ranking CISO executive jobs, the problem is widespread across the industry. Infosec -- or cyber-security -- has a habit of grinding through employees due to the rigors of the job.

Low-level infosec positions, like threat analyst or penetration tester, are just as bad in terms of stress level, if not worse, primarily for the same reasons -- constant fear of new incoming attacks, long-working hours, low pay, almost no job satisfaction.

Within the infosec community, signs of the growing problem of stress and burnout leading to mental health issues have been mounting. There are some efforts underway to raise awareness about infosec job stress levels, burnouts, along with the mental health issues arising from ignoring the first two. This has seen the rise of so-called "Mental Health Hackers", an online community that has been attending cybersecurity conferences on a regular basis in order to raise awareness on the topic.

I don't see an obvious solution to the dilemma other than time. The problem, ultimately, is one of respect. It's impossible for other C-level executives to respect what they do not understand. Traditionally, nerds and geeks have enjoyed keeping their dark arts secret. But being understood is vastly more valuable than being mysterious. So part of your job should be to explain and train the other C-level execs so that they can better understand what the job is about. Fortunately, time and additional experience with the realities of cyber-crime are going to slowly bring about a cultural attitude change. It's in process now, but that sort of change takes time.

So, my advice to CISOs would be to try not to carry the entire organization's cyber-security responsibility on one's own shoulders. I'm sure that's easier said than done. But try. And also try to retain a sense of perspective as much as possible. In the end it’s just a job and your life is yours. Don't give it away.

Microsoft’s “ElectionGuard” being used for the first time today!
As the saying goes, if it’s Tuesday there’s an election somewhere. And in this case that somewhere is the small town of Fulton, Wisconsin. But what’s making history there, today, is that the residents of Fulton, Wisconsin will be electing representatives for the Wisconsin Supreme Court using voting machines for the first time powered by Microsoft's ElectionGuard software.

These are the first voting machines deployed in any US election that will be running Microsoft's new voting software, which we've been keeping our eye on in this podcast. Recall that "ElectionGuard" is a fully open SDK that Microsoft has made available at no charge on GitHub: https://github.com/microsoft/electionguard

The project's goal is to create voting software that uses strong encryption, built by some of the world's top cryptographers, and allowing it to be extensively audited for bugs.

The project has moved with startling speed since it is viewed with great hope and optimism by US election officials. Announced in May of 2019, it matured from a simple idea to a US election pilot program in only nine months.
Microsoft first demonstrated their prototype voting machines to the small audience of the Aspen

Security Forum last July, they then released the first ElectionGuard code to GitHub in September, then opened a bug bounty program the following month in October.

Today's pilot test is deliberately small, with only a few hundred votes expected to be cast, but this will provide voting machine vendors, as well as quite anxious US election officials, with a real-world test of the software to see whether it's worth a shot and ready for wider deployment.

Before today's event, Tom Burt, Microsoft's Vice President for Customer Security & Trust said that using ElectionGuard won't be complicated since Microsoft designed the software from the ground up around ease of use, accessibility, and a user-friendly interface. He explained that the voting experience is a three-step process:

1. First, a voter will select candidates on a touchscreen and verify their choices.
2. Second, the voter will print and review for accuracy a paper ballot and simultaneously receive a separate tracking code.
3. Third, the voter will deposit their ballot into a ballot box for counting."

But, as we've described, there's a LOT of wonderful quite advanced crypto technology happening behind the scenes:

- After casting their ballot, each voter receives a tracking code.

- They can later use the tracking code on an election website to verify that their vote has been counted and that the vote has not been altered.

- The tracking code does not reveal the vote, so it won't allow third-parties to see who voted for whom.

ElectionGuard employs a homomorphic encryption scheme developed in-house at Microsoft under Senior Cryptographer Josh Benaloh. Counterintuitive though it is, this form of encryption allows the counting of individual votes while keeping the votes encrypted.

The ElectionGuard SDK also supports third-party "verifier" apps to independently check that encrypted votes have been counted properly and not altered. Verifier apps were created for use by voting officials, the media, or any third party interested in the voting process.

And ElectionGuard machines can also produce paper ballots, as a printed record of their vote, which voters can place inside voting boxes, like old-fashioned ballots. And, finally, ElectionGuard supports voting through open accessibility hardware.

The voting machines being deployed tomorrow in Fulton were built by VotingWorks at "Voting.Works" https://voting.works/ And their homepage is exactly what we want to see. It states: Democracy is a choice. / VotingWorks is a non-partisan non-profit, building a secure, affordable, and delightful voting system. Our voting machine creates paper ballots that voters can directly verify. Our risk-limiting audit software ensures votes cast on any paper-based system are correctly tabulated. Our source code is available on GitHub. You can help by making a tax-deductible donation or joining our team.
And VotingWorks is not alone. Other voting machine vendors including Smartmatic and Clear

Ballot have also announced partnerships with Microsoft to build ElectionGuard-based voting machines and a fourth group, Dominion Voting Systems, is also exploring the use of Microsoft's SDK. This is a perfect storm outcome since once officials see how this works, what it means for the systems to be open and auditable, and what this system features, no one who isn't doing this will continue being viable. This makes the welcome and long-overdue end to proprietary closed voting machine systems. And good riddance!

SQRL

To the growing list of SQRL implementations we now add a general-purpose pure PHP implementation of SQRL for the Laravel (lara-vel) PHP framework.

https://sqrl.grc.com/threads/new-finished-project-sqrl-for-laravel-package.1088/
https://github.com/DestruidorPT/laravel-sqrl-auth

SpinRite

Just built the FreeDOS kernel.

The Internet of Troubles

IoT lightbulb vulnerabilities are not such a joke, after all.
Our listeners know that I often joke about having our internal networks hacked and attacked by something as ridiculous-seeming as an IoT lightbulb. I chose "lightbulbs" to receive that abuse over the general lack of attention to IoT security because they are pretty much the dumbest, lowest rung of the ladder and least fancy IoT device we have.

Well... guess what: It turns out that the extremely popular Philips IoT lightbulbs, or in this case the "bridge" they require, ARE able to expose our internal WiFi networks to bad guys.

The Hacker News begins their coverage of this latest threat by observing that:

> "There are over a hundred potential ways hackers can ruin your life by having access to your WiFi network that's also connected to your computers, smartphones, and other smart devices.
>
> Whether it's about exploiting operating system and software vulnerabilities or manipulating network traffic, every attack relies on the reachability between an attacker and the targeted devices.
> In recent years, we have seen how hundreds of widely used smart-but-insecure devices made it easier for remote attackers to sneak into connected networks without breaking WiFi passwords.
> In the latest research shared with The Hacker News, Check Point experts revealed a new

high-severity vulnerability affecting Philips Hue Smart Light Bulbs that can be exploited over-the-air from over 100 meters away to gain entry into a targeted WiFi network.

The underlying high-severity vulnerability, tracked as CVE-2020-6007, resides in the way Philips implemented the Zigbee communication protocol in its smart light bulb, leading to a heap-based buffer overflow issue. [Whoops!]

As we know, ZigBee is the widely used "mesh" wireless technology that allows each device to communicate with any other device on the network. It's the protocol built into tens of millions of devices worldwide, including Amazon Echo, Samsung SmartThings, Belkin Emo and more.

The Check Point researchers said: "Through this exploitation, a threat actor can infiltrate a home or office's computer network over-the-air, spreading ransomware or spyware, by using nothing but a laptop and an antenna from over 100 meters away."

Check Point also confirmed that the buffer overflow happens on a component called the "bridge" which is the module that receives remote commands sent to the bulb over the Zigbee protocol from other devices like a mobile app or the Amazon Echo home assistant.

Due to its severity, Check Point is, so far, not revealing the full technical details, nor are they providing any proof-of-concept exploit for the flaw in order to give affected users some time to apply patches. But here's what we know from Check Point's posting about the flaw: https://blog.checkpoint.com/2020/02/05/the-dark-side-of-smart-lighting-check-point-research-shows-how-business-and-home-networks-can-be-hacked-from-a-lightbulb/

CheckPoint wrote:

With the help of the CheckPoint Institute for Information Security (CPIIS) at Tel Aviv University, the researchers were able to take control of a Hue lightbulb on a target network and install malicious firmware on it. From that point, they used the lightbulb as a platform to take over the bulbs' control bridge, and attacked the target network as follows:

- The hacker controls the bulb's color or brightness to trick users into thinking the bulb has a glitch. The bulb appears as 'Unreachable' in the user's control app, so they try to 'reset' it.
- The only way to reset the bulb is to delete it from the app, and then instruct the control bridge to re-discover the bulb.
- The bridge discovers the compromised bulb, and the user adds it back onto their network.
- The hacker-controlled bulb with updated firmware then uses the ZigBee protocol vulnerabilities to trigger a heap-based buffer overflow on the control bridge, by sending a large amount of data to it. This data also enables the hacker to install malware on the bridge – which is in turn connected to the target business or home network.

The malware connects back to the hacker and using a known exploit (such as EternalBlue), they can infiltrate the target IP network from the bridge to spread ransomware or spyware.

In other words, the attackers first take control of one of the lightbulbs. They install malign firmware onto it, then makes it go crazy so that its owner thinks "what the hell? It's gone nuts!" So the owner deletes and re-adds the lightbulb to their system… and in the re-pairing process

the bulb is able to download additional malign firmware onto the bridge to more fully and significantly compromise the internal network, thus establishing a beachhead from which further attacks are possible. <sigh> We really are living in a strange new world.

This research was first disclosed to Philips and Signify, who is the owner of the Philips Hue brand, back in November. Signify confirmed the existence of the vulnerability in their product, and after about two months issued a patched firmware version (Firmware 1935144040) which is now available on their site:

https://www2.meethue.com/en-us/support/release-notes/bridge

CheckPoint said: "In a joint decision with Signify, we decided to postpone the release of the full technical details of our research in order to allow Philips Hue clients to have enough time to safely update their products to the latest version. The full technical details of this research will only be published in our research blog (https://www.research.checkpoint.com/) in the upcoming weeks. Stay tuned.

I was curious to know whether Philips Hue lightbulbs and their bridge would probably be automatically updated, so I checked out the "MeetHue/com" site and I found the following:

- Bridge V1 = Round-shape bridge
- Bridge V2 = Square-shape bridge (support Apple HomeKit)

If you don't want to miss any improvements on quality, security or performance and you want your Hue System full compatible with the upcoming new Hue products, please be sure that you enable automatic updates for your Hue System in the Hue app: (Settings -> Software update -> Automatic Update)

January 13, 2020

- Firmware 1935144040 (Bridge V2)
- We regularly update your Hue Bridge to improve the performance and reliability of the system.
- This update includes a patch for a security vulnerability in the Hue Bridge v2.

Okay... that was just the warm-up!

They're known as the "SweynTooth Vulnerabilities"

They are a set of more than 12 newly discovered vulnerabilities across a wide range of Bluetooth

devices, many of which will never be updated, which allow for, among other things, full device compromise. Only 12 have been disclosed so far since some BlueTooth vendors have not yet released updated SDKs... so more will be forthcoming.

But let's back up a bit...

First of all, I know that everyone is thinking "SweynTooth?!?!" The etymology of "SweynTooth" is not as immediately obvious as are many other named vulnerabilities.

In this case, "SweynTooth" was formed from the names of Sweyn Forkbeard and his father, King Harald Bluetooth -- King Harald BlueTooth, of course, being the namesake of our widely used Bluetooth Technology. As it happens, Harald's upstart son Sweyn revolted against his father, forcing King Harald into exile, which shortly thereafter led to the King's death. The discoverers of these vulnerabilities wrote that they "envision that if SweynTooth style vulnerabilities are not appropriately handled by BLE vendors, then the technology can become a breeding ground for attackers… which may, in turn, lead the Bluetooth technology to become obsolete."

https://asset-group.github.io/disclosures/sweyntooth/

"**SweynTooth: Unleashing Mayhem over Bluetooth Low Energy**"
https://asset-group.github.io/disclosures/sweyntooth/sweyntooth.pdf

The authors begin their 11-page partial disclosure paper by writing:

> SWEYNTOOTH captures a family of 12 vulnerabilities (more under non-disclosure) across different BLE software development kits (SDKs) of seven major system-on-a-chip (SoC) vendors. The vulnerabilities expose flaws in specific BLE SoC implementations that allow an attacker in radio range to trigger deadlocks, crashes and buffer overflows or completely bypass security, depending on the circumstances.
>
> SWEYNTOOTH potentially affects IoT products in appliances such as smarthomes, wearables and environmental tracking or sensing. We have also identified several medical and logistics products that could be affected. As of today, SWEYNTOOTH vulnerabilities are found in the BLE SDKs sold by major SoC vendors, such as Texas Instruments, NXP, Cypress, Dialog Semiconductors, Microchip, STMicroelectronics and Telink Semiconductor. By no means, this list of SoC vendors is exhaustive in terms of being affected by SWEYNTOOTH. We have followed responsive disclosure during our discovery, which allowed almost all SoC vendors to publicly release their respective patches already. However, a substantial number of IoT products relying on the affected SoCs for BLE connectivity will still need to independently receive patches from their respective vendors, as long as a firmware update mechanism is supported by the vendor. SWEYNTOOTH highlights concrete flaws in the BLE stack certification process. We envision substantial amendments to the BLE stack certification to avoid SWEYNTOOTH style security flaws. We also urge SoC vendors and IoT product manufacturers to be aware of such security issues and to initiate focused effort in security testing. A proper classification of the vulnerability set is presented in the next section.
> 1.1 Types of vulnerabilities

We have classified the SWEYNTOOTH vulnerabilities according to their types and their behaviours on the affected BLEdevices.

- Crash: Vulnerabilities in this category can remotely crash a device by triggering hard faults. This happens due to some incorrect code behaviour or memory corruption, e.g., when a buffer overflow on BLE reception buffer occurs. When a device crash occurs, they usually restart. However, such a restart capability depends on whether a correct hard fault handling mechanism was implemented in the product that uses the vulnerable BLE SoC.

- Deadlock: Deadlocks are vulnerabilities that affect the availability of the BLE connection without causing a hard fault or memory corruption. Usually they occur due to some improper synchronisation between user code and the SDK firmware distributed by the SoC vendor, leaving the user code being stuck at some point. Crashes originated from hard faults, if not properly handled, can become a deadlock if the device is not automatically restarted. In most cases, when a deadlock occurs, the user is required to manually power off and power on the device to re-establish proper BLE communication.

- Security Bypass: This vulnerability is the most critical one. This is because the vulnerability allows attackers in radio range to bypass the latest secure pairing mode of BLE, i.e., the Secure Connections pairing mode [12]. After the bypass is completed, an attacker in the radio range has arbitrary read or write access to the device's functions. Function which are only intended to be accessed by authorised users.

This research has led to the issuance of a raft of new CVEs:

Type	*Vulnerability Name*	*Affected Vendors*	*CVE*
Crash	Link Layer Length Overflow	Cypress NXP	CVE-2019-16336 (6.1) CVE-2019-17519 (6.1)
	Truncated L2CAP	Dialog Semiconductors	CVE-2019-17517 (6.3)
	Silent Length Overflow	Dialog Semiconductors	CVE-2019-17518 (6.4)
	Public Key Crash	Texas Instruments	CVE-2019-17520 (6.6)
	Invalid L2CAP Fragment	Microchip	CVE-2019-19195 (6.8)
	Key Size Overflow	Telink Semiconductor	CVE-2019-19196 (6.9)
Deadlock	LLID Deadlock	Cypress NXP	CVE-2019-17061 (6.2) CVE-2019-17060 (6.2)
	Sequential ATT Deadlock	STMicroelectronics	CVE-2019-19192 (6.7)
	Invalid Connection Request	Texas Instruments	CVE-2019-19193 (6.5)
Security Bypass	Zero LTK Installation	Telink Semiconductor	CVE-2019-19194 (6.10)

The authors provide a table of Vulnerabilities and SDK versions of the affected SoCs:

Vuln.	SoC Vendor	SoC Model	SDK Ver.	Qualification ID(s)
	BLE Version 5.0/5.1			
6.1,6.2	Cypress (PSoC 6)	CYBLE-416045	2.10	99158
6.5,6.6	Texas Instruments	CC2640R2	3.30.00.20	94079
6.9,6.10	Telink	TLSR8258	3.4.0	92269, 136037
6.7	STMicroelectronics	WB55	1.3.0	111668
6.7	STMicroelectroncis	BlueNRG-2	3.1.0	87428, 106700, 94075
6.4	Dialog	DA1469X*	10.0.6	100899
6.3	Dialog	DA14585/6*	6.0.12.1020	91436
	BLE Version 4.2			
6.1,6.2	Cypress (PSoC 4)	CYBL11573	3.60	62243, 136808, 79697, 82951, 79480
6.1,6.2	NXP	KW41Z	2.2.1	84040
6.4	Dialog	DA14680	1.0.14.X	87407, 84084, 71309, 75255
	BLE Version 4.1			
6.5	Texas Instruments	CC2540	1.5.0	23454, 127418
6.3	Dialog	DA14580	5.0.4	83573
6.8	Microchip	ATSAMB11	6.2	73346

The biggest problem we face is the BlueTooth consumer supply chain.

These guys are way way back up the supply chain, at the SDK -- the Software Development Kit -- level where the engineers customize the SoC supplier's supplied BlueTooth software stack for their application and then burn that as firmware into their product. Today, by far and away, MOST BlueTooth-enabled devices are not our mainstream smartphones and home hubs which have an active and responsible ecosystem backing them up. MOST BlueTooth-enabled devices are what CONNECT to those smartphones or base stations. And MOST of those BlueTooth-enabled devices will NEVER see a firmware update. Even if one was available on some Chinese website somewhere, how would the user, who is unknown to the seller, ever find out, or even care?

Imagine some random BLE-based corporate or residential alarm system that was purchased through Amazon from a Chinese company that used a BLE chip from Cypress, TI, NXP (Philips), Microchip, or apparently any of the other BLE suppliers. We KNOW that original manufacturer doesn't care about after-sales support. It's not offered. So that alarm system is highly unlikely to ever receive a firmware update. It works, yes… but it will also be FOREVER vulnerable to wireless proximity attacks that will eventually be made fully public. And in the specific case of an alarm system, it doesn't need to be vulnerable to the least common of these attacks -- the security bypass. It may be sufficient for an attacking burgler's purpose to create a deadlock so that it's unable to sound the alarm. But if a security bypass can be found, even more damage could be done.

We've all seen science fiction where, as with Neo at the start of The Matrix, some lesser-skilled individuals purchase some advanced hacking technology from a more highly skilled hacker. I've always regarded this sort of world as more fanciful than real: The idea that anything can be hacked for a price.

But it is becoming increasingly clear that the way things are going, the fundamentally insecure way that we cavalierly and casually purchase, deploy, use and rely upon technology that CAN be hacked and will FOREVER be known to be hackable -- really does suggest a brave new world where ANYTHING can be hacked for a price.

[RINGgggggg…] "Hey, what's up?" [...] "Oh, you want to bypass a Chimera 412 home alarm system? Sure. Piece a' cake. Those use the old Cypress 2313 BLE chip that its manufacturer never updated. The hack for that's been around forever, but mine adds a few extra touches. Transfer 1/10th of a bitcoin to my wallet and once it's there I'll shoot you a script that you can run on any rooted Android smartphone that'll completely and silently shutdown any Chimera 412."

As we know, that day hasn't arrived yet... but it is seeming less and less far fetched with each passing podcast.