Security Now! #629 - 09-19-17

Apple Bakes Cookies

This week on Security Now!

This week Padre and I discuss what was up with SN's recent audio troubles, more on the Equifax Fiasco, the EFF & Cory Doctorow weigh in on forthcoming browser encrypted media extensions (EME), an emerging browser-based payment standard, when 2-factor is not 2-factor, the CCleaner breach and what it means, a new Bluetooth-based attack, an incredibly welcome and brilliant cookie privacy feature in iOS 11, and a heads-up caution about the volatility of Google's Android smartphone cloud backups.

Our Picture of the Week

Security News

The Audio Fix.

- Coax fittings that I hadn't examined in a long time (if it's not broke).
- An unterminated bridge to nowhere.
- A newer USB/Audio interface that demanded to much from the PC I'd been using.
- Cox could NOT have been better and more super-helpful.
- And not only the folks who know me.
- Changing cable modems requires some interaction and they've been wonderful too.

The Equifax Fiasco -- Followup with Father Robert

- My own advice was minimal: Just lock credit reports.
- But Father Robert had some additional thoughts and suggestions.

We're beginning to learn a bit more about Equifax

http://www.marketwatch.com/story/equifax-ceo-hired-a-music-major-as-the-companys-chief-security-officer-2017-09-15
Opinion: "Equifax hired a music major as chief security officer and she has just retired"

Susan Mauldin, whose identity is being scrubbed from the internet, studied music composition
Susan Mauldin's LinkedIn page was made private and her last name replaced with "M."

MarketWatch:
When Congress hauls in Equifax CEO Richard Smith to grill him, it can start by asking why he put someone with degrees in music in charge of the company's data security.

And then they might also ask him if anyone at the company has been involved in efforts to cover up Susan Mauldin's lack of educational qualifications since the data breach became public.

It would be fascinating to hear Smith try to explain both of those extraordinary items.

If those events don't put the final nails in his professional coffin, accountability in the U.S. is officially dead. And late Friday Equifax said both Mauldin and the company's chief information officer have retired effective immediately.

Equifax "Chief Security Officer" Susan Mauldin has a bachelor's degree and a master of fine arts degree in music composition from the University of Georgia. Her LinkedIn professional profile lists no education related to technology or security.

Reporting by a few tech-savvy blogs has found that as soon as the Equifax data breach became public, someone began to scrub the internet of information about Mauldin.

Her LinkedIn page was made private and her last name replaced with "M." Two videos of interviews with Mauldin have been removed from YouTube. A podcast of an interview has also been taken down.

The marketWatch piece concludes: "Everything about this fiasco just gets more and more surreal."

"Legalist: Stand Up to Equifax"
https://www.legalist.com/equifax/
Social networking for attorneys and prospective clients.

VERY IMPORTANT: Yesterday, the EFF (Cory Doctorow) weighs in on browser EME
https://www.eff.org/deeplinks/2017/09/open-letter-w3c-director-ceo-team-and-membership

Dear Jeff, Tim, and colleagues,

In 2013, EFF was disappointed to learn that the W3C had taken on the project of standardizing “Encrypted Media Extensions,” an API whose sole function was to provide a first-class role for DRM within the Web browser ecosystem. By doing so, the organization offered the use of its patent pool, its staff support, and its moral authority to the idea that browsers can and should be designed to cede control over key aspects from users to remote parties.

When it became clear, following our formal objection, that the W3C's largest corporate members and leadership were wedded to this project despite strong discontent from within the W3C membership and staff, their most important partners, and other supporters of the open Web, we proposed a compromise. We agreed to stand down regarding the EME standard, provided that the W3C extend its existing IPR policies [IPR = Intellectual Property Rights] to deter members from using DRM laws in connection with the EME (such as Section 1201 of the US Digital Millennium Copyright Act or European national implementations of Article 6 of the EUCD) except in combination with another cause of action.

[Our listeners know quite well how every lesson we're learning shows us how clearly crucial it is that academic security researchers be allowed to have verification oversight over proprietary encryption systems upon which many people depend. This MUST NOT be closed.]

This covenant would allow the W3C's large corporate members to enforce their copyrights. Indeed, it kept intact every legal right to which entertainment companies, DRM vendors, and their business partners can otherwise lay claim. The compromise merely restricted their ability to use the W3C's DRM to shut down legitimate activities, like research and modifications, that required circumvention of DRM. It would signal to the world that the W3C wanted to make a difference in how DRM was enforced: that it would use its authority to draw a line between the acceptability of DRM as an optional technology, as opposed to an excuse to undermine legitimate research and innovation.

More directly, such a covenant would have helped protect the key stakeholders, present and future, who both depend on the openness of the Web, and who actively work to protect its safety and universality. It would offer some legal clarity for those who bypass DRM to engage in security research to find defects that would endanger billions of web users; or who automate the creation of enhanced, accessible video for people with disabilities; or who archive the Web for posterity. It would help protect new market entrants intent on creating competitive, innovative products, unimagined by the vendors locking down web video.

Despite the support of W3C members from many sectors, the leadership of the W3C rejected this compromise.

The W3C leadership countered with proposals — like the chartering of a nonbinding discussion group on the policy questions that was not scheduled to report in until long after the EME ship had sailed — that would have still left researchers, governments, archives, security experts unprotected.

The W3C is a body that ostensibly operates on consensus. Nevertheless, as the coalition in support of a DRM compromise grew and grew — and the large corporate members continued to reject any meaningful compromise — the W3C leadership persisted in treating EME as topic that could be decided by one side of the debate. In essence, a core of EME proponents was able to impose its will on the Consortium, over the wishes of a sizeable group of objectors — and every person who uses the web. The Director decided to personally override every single objection raised by the members, articulating several benefits that EME offered over the DRM that HTML5 had made impossible.

But those very benefits (such as improvements to accessibility and privacy) depend on the public being able to exercise rights they lose under DRM law — which meant that without the compromise the Director was overriding, none of those benefits could be realized, either. That rejection prompted the first appeal against the Director in W3C history.

In our campaigning on this issue, we have spoken to many, many members' representatives who privately confided their belief that the EME was a terrible idea (generally they used stronger language) and their sincere desire that their employer wasn't on the wrong side of this issue. This is unsurprising. You have to search long and hard to find an independent technologist who believes that DRM is possible, let alone a good idea.

[Again... as we've often said here... DRM *is* a fundamental impossibility when content decryption for playback must be performed in front of the content consumer. it just cannot be done.]

Yet, somewhere along the way, the business values of those outside the web got important enough, and the values of technologists who built it got disposable enough, that even the wise elders who make our standards voted for something they know to be a fool's errand.

We believe they will regret that choice. Today, the W3C bequeaths an legally unauditable attack-surface to browsers used by billions of people. They give media companies the power to sue or intimidate away those who might re-purpose video for people with disabilities. They side against the archivists who are scrambling to preserve the public record of our era. The W3C process has been abused by companies that made their fortunes by upsetting the established order, and now, thanks to EME, they’ll be able to ensure no one ever subjects them to the same innovative pressures.

So we'll keep fighting to fight to keep the web free and open. We'll keep suing the US government to overturn the laws that make DRM so toxic, and we'll keep bringing that fight to the world's legislatures that are being misled by the US Trade Representative to instigate local equivalents to America's legal mistakes.

We will renew our work to battle the media companies that fail to adapt videos for accessibility purposes, even though the W3C squandered the perfect moment to exact a promise to protect those who are doing that work for them.

We will defend those who are put in harm's way for blowing the whistle on defects in EME implementations.

It is a tragedy that we will be doing that without our friends at the W3C, and with the world believing that the pioneers and creators of the web no longer care about these matters.

Effective today, EFF is resigning from the W3C.

Thank you,

Cory Doctorow
Advisory Committee Representative to the W3C for the Electronic Frontier Foundation

Browsers on board for W3C Payment Request API
https://www.finextra.com/pressarticle/70744/browsers-on-board-for-w3c-payment-request-api

Yesterday: Browsers on board for W3C Payment Request API / 18 September 2017

The World Wide Web Consortium (W3C) today called for broad implementation and testing of Web technologies to make online checkout easier for users and improve conversions and security for merchants.
All major browser makers are now implementing Payment Request API. The Web Payments Working Group encourages merchants, Web developers, and users to experiment with these early implementations and provide feedback to the group. In parallel, the Working Group will be expanding its test suite for the API to help ensure browser interoperability.

Improved User Experience

Making purchases on the web, particularly on mobile, can be a frustrating experience. Every web site has its own flow, and most require users to manually type in the same addresses, contact information, and payment credentials again and again. This can lead to shopping cart abandonment and lost customer loyalty. Likewise, users may abandon checkout if their preferred payment methods are unavailable, but it can be difficult and time-consuming for developers to create and maintain checkout pages that support multiple payment methods.

The Payment Request API (and supporting specifications) enable merchants to create streamlined checkout pages where people reuse previously stored information, saving time and effort and reducing error.

With these technologies, users no longer complete Web forms to provide payment credentials, shipping information, and contact information. Instead, the user registers support for different payment methods —such as card payments, proprietary native mobile payments, bitcoin or other distributed ledgers, or credit transfers— with the browser or other user agent. During checkout, the browser determines which of the user's payment methods match those accepted by the merchant. The browser displays just the matches, which simplifies selection of the user's preferred payment application and makes the experience consistent across the Web. The user then chooses a payment method, after which the merchant receives relevant information through the standard API in order to complete the transaction.
Increased Conversions and Security for Merchants

Payment Request API is expected to lower the cost of creating and maintaining a checkout page and increase payment security. The standard will make it easier to bring more secure payment methods (e.g., tokenized card payments) to the Web. The standard also means that merchants or their service providers can achieve a streamlined user experience without having to store customer payment credentials, potentially reducing their liability.

For more information about using the APIs, security, and the relation to various rules and regulations, please see the Payment Request FAQ.

More Choice Via Third Party Payment Apps

The Web Payments Working Group envisions that a diverse ecosystem of third party payment apps will give merchants and users more payment choices.

Browsers and other current implementations of Payment Request API allow users to store credit and debit card information for convenient reuse. Some also already support user registration of native mobile payment apps.

In addition, to enable users to make payments from Web sites, the Web Payments Working Group is also working on the Payment Handler API.

What is it?

- Its a means for a remote site to send to the browser a request for the payment-related information it wants.

- The browser displays a standardized payment information request to the user who either approves or disapproves.

- If approved, the browser returns the requested information to the remote server to process the payment.

- The server-side payment flow itself is unchanged.

- The Payment Request API is simply a well-defined, managed and overseen database query from a remote webserver to the user's browser... which has previously been loaded with all required purchase information.

More demonstrations of the insecurity of text messages as a 2nd factor.
https://www.theverge.com/2017/9/18/16328172/sms-two-factor-authentication-hack-password-bitcoin

For a long time, security experts have warned that text messages are vulnerable to hijacking — and this morning, they showed what it looks like in practice. A demonstration video posted by Positive Technologies (and first reported by Forbes) shows how easy it is to hack into a bitcoin wallet by intercepting text messages in transit.

The group targeted a Coinbase account protected by two-factor authentication, which was registered to a Gmail account also protected by two-factor. By exploiting known flaws in the cell network, the group was able to intercept all text messages sent to the number for a set period of time. That was enough to reset the password to the Gmail account and then take control of the Coinbase wallet. All the group needed was the name, surname and phone number of the targeted Bitcoin user. These were security researchers rather than criminals, so they didn't actually steal anyone's bitcoin, although that would have been an easy step to take.

[[snip]]

There are a few concrete steps you can take to protect yourself from this kind of attack. On some services, you can revoke the option for SMS two-factor and account recovery entirely, which you should do as soon as you've got a more secure app-based method established. Google, for instance, will let you manage two-factor and account recovery here and here; just set up Authenticator or a recovery code, then go to the SMS option for each and click "Remove Phone."

Still, the industry as a whole has been very slow in moving away from SMS as a second factor, which has severely weakened the overall security of the system. As long as SMS is included as an option for two-factor, we'll continue to see attacks like this.

The crux of the problem is that another factor has been added and is now being depended upon, but it is WEAKER than the existing factors... so that forms a readily exploitable loophole. In other words, this form of 2-factor authentication makes it WORSE!!

It's not "2-Factor" when the second factor can be used to recover from the loss of the first!... it's "alternative factor" where the alternate is actually WEAKER than the primary factor.

Yesterday, CCleaner disclosed a breach in their own security.
For a period of four weeks -- between August 15th, 2017 and September 12th, 2017 -- The downloadable file delivering the 32-bit version of CCleaner v5.33 was compromised with the "Floxif" malware which infects Windows executable and DLLs, backdooring the machine to install additional malware.

2.27 million CCleaner users inadvertently downloaded the malware during this time.

To check for infection, open the system's Registry Editor and navigate to: HKEY_LOCAL_MACHINE\SOFTWARE\Piriform\Agomo. Under this key will be two data values

named MUID and TCID, which are used by the installed Floxif infection.
The malware executed only if the user was using an admin account. Users of low-privileged accounts who installed CCleaner 5.33 would not have been affected.

http://www.piriform.com/news/release-announcements/2017/9/18/security-notification-for-ccleaner-v5336162-and-ccleaner-cloud-v1073191-for-32-bit-windows-users

http://blog.talosintelligence.com/2017/09/avast-distributes-malware.html?m=1

On September 13, 2017 while conducting customer beta testing of our new exploit detection technology, Cisco Talos identified a specific executable which was triggering our advanced malware protection systems. Upon closer inspection, the executable in question was the installer for CCleaner v5.33, which was being delivered to endpoints by the legitimate CCleaner download servers. Talos began initial analysis to determine what was causing this technology to flag CCleaner.

We identified that even though the downloaded installation executable was signed using a valid digital signature issued to Piriform, CCleaner was not the only application that came with the download. During the installation of CCleaner 5.33, the 32-bit CCleaner binary that was included also contained a malicious payload that featured a Domain Generation Algorithm (DGA) as well as hardcoded Command and Control (C2) functionality. We confirmed that this malicious version of CCleaner was being hosted directly on CCleaner's download server as recently as September 11, 2017.

In reviewing the Version History page on the CCleaner download site, it appears that the affected version (5.33) was released on August 15, 2017. On September 12, 2017 version 5.34 was released. The version containing the malicious payload (5.33) was being distributed between these dates. This version was signed using a valid certificate that was issued to Piriform Ltd by Symantec and is valid through 10/10/2018.

Piriform was the company that Avast recently acquired and was the original company who developed the CCleaner software application.

Piriform responds:

Security Notification for CCleaner v5.33.6162 and CCleaner Cloud v1.07.3191 for 32-bit Windows users

We recently determined that older versions of our Piriform CCleaner v5.33.6162 and CCleaner Cloud v1.07.3191 had been compromised. We estimate that 2.27 million people used the affected software. We resolved this quickly and believe no harm was done to any of our users. This compromise only affected customers with the 32-bit version of the v5.33.6162 of CCleaner and the v1.07.3191 of CCleaner Cloud. No other Piriform or CCleaner products were affected. We encourage all users of the 32-bit version of CCleaner v5.33.6162 to download v5.34 here: download. We apologize and are taking extra measures to ensure this does not happen again.

Issue Summary: Our new parent company, the security company Avast, determined on the 12th of September that the 32-bit version of our CCleaner v5.33.6162 and CCleaner Cloud v1.07.3191 products, which may have been used by up to 3% of our users, had been compromised in a sophisticated manner. Piriform CCleaner v5.33.6162 was released on the 15th of August, and a regularly scheduled update to CCleaner, without compromised code, was released on the 12th of September. CCleaner Cloud v1.07.3191 was released on the 24th of August, and updated with a version without compromised code on September 15. The compromise could cause the transmission of non-sensitive data (computer name, IP address, list of installed software, list of active software, list of network adapters) to a 3rd party computer server in the USA. We have no indications that any other data has been sent to the server. Working with US law enforcement, we caused this server to be shut down on the 15th of September before any known harm was done. It would have been an impediment to the law enforcement agency's investigation to have gone public with this before the server was disabled and we completed our initial assessment. Between the 12th and the 15th, we took immediate action to make sure that our Piriform CCleaner v5.33.6162 and CCleaner Cloud v1.07.3191 users were safe - we worked with download sites to remove CCleaner v5.33.6162, we pushed out a notification to update CCleaner users from v5.33.6162 to v5.34, we automatically updated those where it was possible to do so, and we automatically updated CCleaner Cloud users from v1.07.3191 to 1.07.3214.

We are continuing to investigate how this compromise happened, who did it, and why. We are working with US law enforcement in their investigation. A more technical description of the issue is on our Piriform blog at: www.piriform.com/news/blog. Again, we sincerely apologize for this and are committed to making sure nothing similar happens again. We encourage any user of the 32-bit version of CCleaner v5.33.6162 to download the latest version of Piriform CCleaner found here: www.piriform.com/ccleaner/download/standard.

The "BlueBorne" attack on BlueTooth

https://lifehacker.com/stop-leaving-your-smartphones-bluetooth-on-1817176967

A security research company, Armis Incorporated in Palo Alto, describes "BlueBorne" (as in "Airbourne" buqt over bluetooth) as a new attack vector which exposes almost every connected device.

They found and responsibly disclosed a large number of 0-day vulnerabilities -- and developed working proof-of-exploits.

The Bluetooth stacks within our devices all run with high privilege and are thus present a highly target rich environment. And the various BT protocols which have been developed and enhanced over time are powerful and ambitious. So getting them PERFECT, as security requires, is not easy.

Coordinated Disclosure:

- Google – Contacted on April 19, 2017, after which details were shared. Released public security update and security bulletin on September 4th, 2017. Coordinated disclosure on September 12th, 2017.

- Microsoft – Contacted on April 19, 2017 after which details were shared. Updates were made on July 11. Public disclosure on September 12, 2017 as part of coordinated disclosure.

- Apple – Contacted on August 9, 2017. Apple had no vulnerability in its current versions.

- Samsung – Contact on three separate occasions in April, May, and June. No response was received back from any outreach.

- Linux – Contacted August 15 and 17, 2017. On September 5, 2017, we connected and provided the necessary information to the the Linux kernel security team and to the Linux distributions security contact list and conversations followed from there. Targeting updates for on or about September 12, 2017 for coordinated disclosure.

This podcast has talked a LOT about always turning off unneeded radios. And iOS has long had the practice of turning Bluetooth back on after a OS update... much to the annoyance of our listeners who kept nothing, as I did, that unneeded and unwanted BT kept popping back up.

"Inspect" - Extension to inspect HTTPS certificate
https://itunes.apple.com/us/app/inspect-extension-to-inspect-https-certificate/id1074957486?mt=8

Google Will Delete Your Backup If You Haven't Used Your Android Phone in 60 Days
https://lifehacker.com/google-will-delete-your-backup-if-you-havent-used-your-1816363955
Just a heads-up to our listeners: A spare and idle Android device will have its backup expunged after 60 days.

SpinRite

Col P @FoodLovingProg
@SGgrc Your 'stache is back :), Do you reckon i'd be able recover a broken Nexus 5X phone with SpinRite? Could i get non-encrypted data off it?

There's a good chance... and I recall that Father Robert has had some success doing this in the past? IF the device can look like a USB storage device, some people HAVE succeeded with the recovery!

Apple's BRILLIANT tracking cookie solution

iOS 11 to introduce improved tracking cookie privacy
https://webkit.org/blog/7675/intelligent-tracking-prevention/

<APPLE> How Does Intelligent Tracking Prevention Work?

Intelligent Tracking Prevention collects statistics on resource loads as well as user interactions such as taps, clicks, and text entries. The statistics are put into buckets per top privately-controlled domain or TLD+1.
Machine Learning Classifier

A machine learning model is used to classify which top privately-controlled domains have the ability to track the user cross-site, based on the collected statistics. Out of the various statistics collected, three vectors turned out to have strong signal for classification based on current tracking practices: subresource under number of unique domains, sub frame under number of unique domains, and number of unique domains redirected to. All data collection and classification happens on-device.
Actions Taken After Classification

Let's say Intelligent Tracking Prevention classifies example.com as having the ability to track the user cross-site. What happens from that point?

If the user has not interacted with example.com in the last 30 days, example.com website data and cookies are immediately purged and continue to be purged if new data is added.

However, if the user interacts with example.com as the top domain, often referred to as a first-party domain, Intelligent Tracking Prevention considers it a signal that the user is interested in the website and temporarily adjusts its behavior as depicted in this timeline:
Intelligent Tracking Prevention Timeline

If the user interacted with example.com the last 24 hours, its cookies will be available when example.com is a third-party. This allows for "Sign in with my X account on Y" login scenarios.

This means users only have long-term persistent cookies and website data from the sites they actually interact with and tracking data is removed proactively as they browse the web.
Partitioned Cookies

If the user interacted with example.com the last 30 days but not the last 24 hours, example.com gets to keep its cookies but they will be partitioned. Partitioned means third-parties get unique, isolated storage per top privately-controlled domain or TLD+1, e.g. account.example.com and www.example.com share the partition example.com.

This makes sure users stay logged in even if they only visit a site occasionally while restricting the use of cookies for cross-site tracking. Note that WebKit already partitions caches and HTML5 storage for all third-party domains.

Meanwhile, advertisers are freaking out! (good!)

https://arstechnica.com/tech-policy/2017/09/ad-industry-deeply-concerned-about-safaris-new-ad-tracking-restrictions/

Six major advertising groups have just published an open letter blasting the new tracking restrictions Apple unveiled in June. They say they are "deeply concerned" about them:

<QUOTE> The infrastructure of the modern Internet depends on consistent and generally applicable standards for cookies, so digital companies can innovate to build content, services, and advertising that are personalized for users and remember their visits. Apple's Safari move breaks those standards and replaces them with an amorphous set of shifting rules that will hurt the user experience and sabotage the economic model for the Internet.

Apple's unilateral and heavy-handed approach is bad for consumer choice and bad for the ad-supported online content and services consumers love. Blocking cookies in this manner will drive a wedge between brands and their customers, and it will make advertising more generic and less timely and useful.