Security Now! #705 - 03-12-19

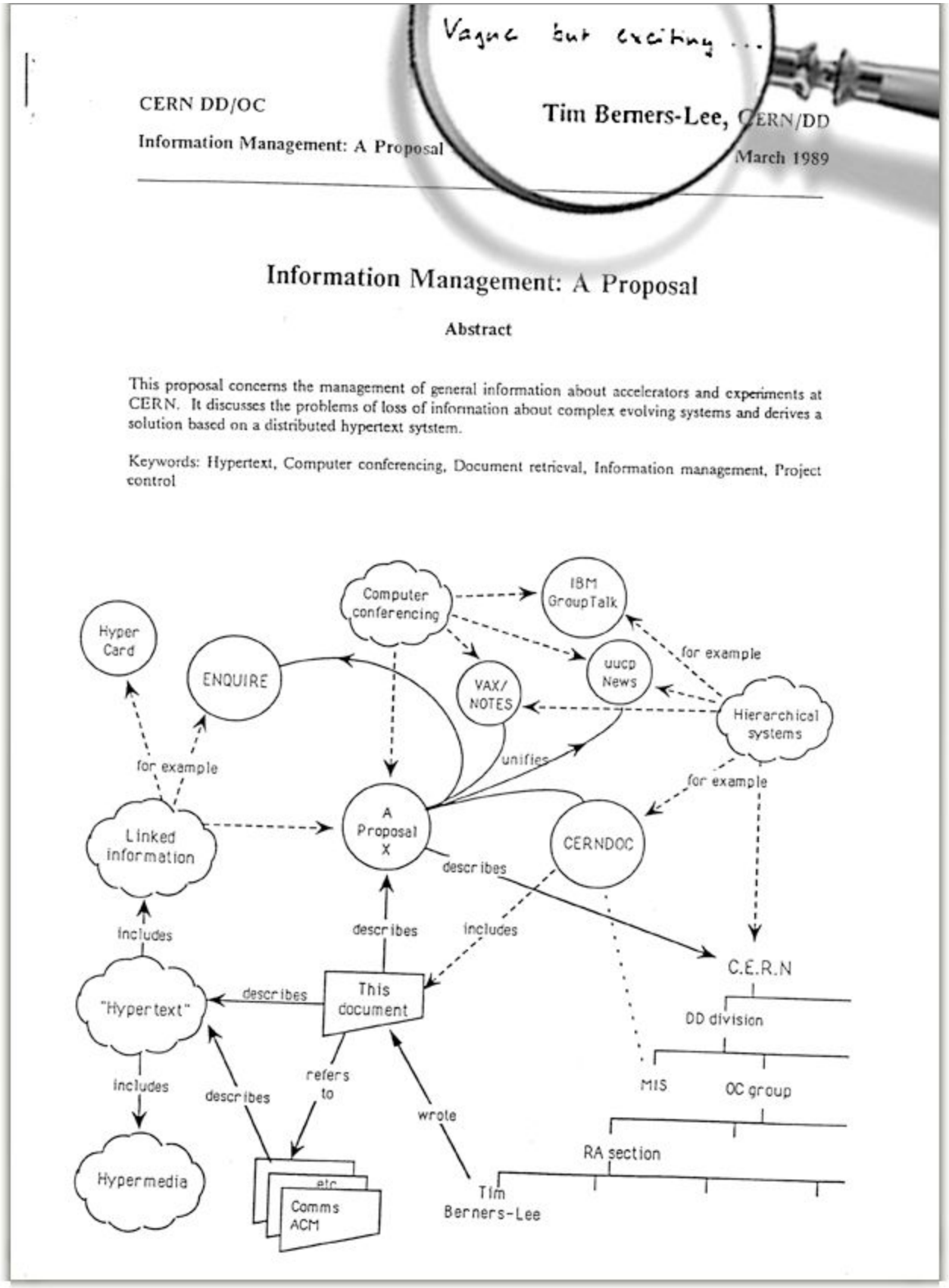
Spoiler

This week on Security Now!

This week we look at the 0-day exploit bidding war that's underway, the NSA's release of Ghidra, Firefox's addition of privacy enhancements which were first developed for the Tor version of Firefox, a pair of 0-days that were biting people in the wild, news of a worrisome breach at Citrix, the risk of claiming to be an unhackable aftermarket car alarm, a new and interesting "windows developers chatting with users" idea at Microsoft, a semi-solution to Windows updates crashing systems, detailed news of the Marriott/Starwood breach, a bit of miscellany from Elaine, a SpinRite question answered, and then we finish with SPOILER, the latest research exploiting yet another new and different consequence of speculation on Intel machines.

See next page for the picture of the week...

Happy 30th Birthday to the World Wide Web!



Vague but exciting ...

CERN DD/OC

Information Management: A Proposal

Tim Berners-Lee, CERN/DD

March 1989

Information Management: A Proposal

Abstract

This proposal concerns the management of general information about accelerators and experiments at CERN. It discusses the problems of loss of information about complex evolving systems and derives a solution based on a distributed hypertext sytstem.

Keywords: Hypertext, Computer conferencing, Document retrieval, Information management, Project control

http://info.cern.ch/Proposal.html

Security News

Zerodium: $500,000 for a Hypervisor 0-Day
Last week's topic was "Careers in Bug Hunting." While we were delivering that podcast, and although we were primarily talking about HackerOne, I mentioned Zerodium as a admitted alternative cash-out source if someone found a particularly tasty and important 0-day flaw. As it happens, while we were delivering that podcast Zerodium was upping the ante! They tweeted:

> We're paying up to $500,000 for #0day exploits targeting VMware ESXi (vSphere) or Microsoft Hyper-V, and allowing Guest-to-Host escapes. The exploits must work with default configs, be reliable, and lead to full access to the host. Contact us: https://t.co/8NeubPvSdj — Zerodium (@Zerodium) March 5, 2019

Zerodium does these [what they call] "acquisition raids" offering limited time increased "incentives" for classes of flaws they have a particular interest in obtaining. Either, I presume, to balance out their exploit portfolio, or perhaps if they have a "customer" who is in particular need.

In this "Acquisition Raid" they bumped the payout for a high-quality, solid and stable hypervisor exploit from $200,000 to $500,000. ZDNet asked Zerodium's CEO Chaouki Bekrar what the deal was. He replied by eMail: "Our new payout for hypervisors will last for a couple of months, and we'll then decide if we reduce it or keep it high, depending on the number of acquisitions we will make."

Wow... "The number of acquisitions." <<shudder>> Zerodium has previously held such "acquisition raids" for 0-days in iOS, instant messaging apps, the Tor Browser, Linux, Adobe Flash Player, routers, and USB thumb drives.

You know what's really going on here? Believe it or not, it's become a bidding war and Zerodium was being outbid! Last summer we talked about Microsoft's decision to buy its own flaws when they upped the ante for Hyper-V bugs to $250,000, thus outbidding Zerodium and other exploit buyers. And, frankly, whereas selling a powerful 0-day to Zerodium would leave me feeling a bit compromised, I would feel 100% great selling Microsoft one of their own bugs! THAT would be wonderful!!

But, Zerodium's Bekrar told ZDNet: "Microsoft's bounty for Hyper-V exploits is very attractive for researchers, however, VMWare is not paying anything to zero-day hunters. We have decided to fill this gap, and we've been paying $200,000 for such exploits, and we've acquired many of them so far. However, we've recently observed an increase in demand from customers, [and] we have decided to increase the bounty to $500,000 to outbid vendors and all existing buyers."

Sheesh. Zerodium claims that the company's hungry customers are government and law enforcement agencies... and these do seem like "government level" purchase prices.

Last week the NSA Released GHIDRA v9
Last Wednesday, during the RSA security conference, as planned and expected, the NSA release GHIDRA (gee-dra) v9. This is their free, powerful and mature interactive multi-platform reverse engineering tool. And it's now on Github in all its (soon to be) open source glory. It's not quite yet open.

https://github.com/NationalSecurityAgency/ghidra

https://ghidra-sre.org/

After our initial mention of it several months ago on the podcast I received all sorts of terrific feedback about GHIDRA's backstory and pronunciation. So I hope I've got it right this time.

GHIDRA has been used internally at the NSA and other similar closely-aligned government agencies for more than a decade, during which time it has been evolving and developing. It will doubtless prove to be extremely useful for anyone researching the operation and security of closed-source software including for reverse engineering malware.

The NSA explained that their general plan was to release Ghidra to enable security researchers to get up to speed and used to working with it before applying for positions at the NSA or other government intelligence agencies with which the NSA has previously shared Ghidra in private.

As we explained when we first noted this coming release, Ghidra is a free alternative to IDA Pro (Interactive Dis-Assembler), which is a similar reverse engineering tool that's only available under a very expensive commercial license, priced at thousands of US dollars per year. By being offered for free, most experts expect Ghidra to snap up a big portion of the reverse engineering tools market share within weeks, especially since early user reviews have been almost all entirely positive.

On Github, the NSA has this to say:

Ghidra Software Reverse Engineering Framework

Ghidra is a software reverse engineering (SRE) framework created and maintained by the National Security Agency Research Directorate. This framework includes a suite of full-featured, high-end software analysis tools that enable users to analyze compiled code on a variety of platforms including Windows, macOS, and Linux. Capabilities include disassembly, assembly, decompilation, graphing, and scripting, along with hundreds of other features. Ghidra supports a wide variety of process instruction sets and executable formats and can be run in both user-interactive and automated modes. Users may also develop their own Ghidra plug-in components and/or scripts using Java or Python.

In support of NSA's Cybersecurity mission, Ghidra was built to solve scaling and teaming problems on complex SRE efforts, and to provide a customizable and extensible SRE research platform. NSA has applied Ghidra SRE capabilities to a variety of problems that involve analyzing malicious code and generating deep insights for SRE analysts who seek a better understanding of potential vulnerabilities in networks and systems.

This repository is a placeholder for the full open source release. Be assured efforts are under way to make the software available here. In the meantime, enjoy using Ghidra on your SRE efforts, developing your own scripts and plugins, and perusing the over-one-million-lines of Java and Sleigh code released within the initial public release. The release can be downloaded from our project homepage. Please consider taking a look at our contributor guide to see how you can participate in this open source project when it becomes available.

If you are interested in projects like this and would like to develop this, and other cybersecurity tools, for NSA to help protect our nation and its allies, consider applying for a career with us.

Interesting recruiting angle.

Installing Ghidra is as simple as unpacking a ZIP archive. The only requirement is a version of the Java Development Kit 11 or later that's needed to run the app's GUI. The tool's official docs have the following to say about installation (or not):

Ghidra does not use a traditional installer program. Instead, the Ghidra distribution file is simply extracted in-place on the filesystem. This approach has advantages and disadvantages. On the up side, administrative privilege is not required to install Ghidra for personal use. Also, because installing Ghidra does not update any OS configurations such as the registry on Windows, removing Ghidra is as simple as deleting the Ghidra installation directory.

Besides an installation guide, Ghidra's docs also come with classes and exercises for beginners, intermediates, and advanced levels that will help users get used to the tool's GUI, which is very different from any similar tools.

So... Not good news for the Hex-Rays folks who have for many years been gouging their customers with extremely high-priced reverse engineering tools. The entire market for such tools has just collapsed.

Firefox to add Tor Browser anti-fingerprinting technique called "letterboxing"

I suppose that I underestimate how determined online advertisers and profilers are to track us. But I certainly get it that knowing all the places that a user goes can reveal a great deal of information about the user. So I suppose I'm glad that my browsing habits are likely to put anyone tracking me to sleep.

But I also get it that the idea of unseen tracking and profiling deeply offends the sensibilities of many users. They are, after all, not being asked and not being remunerated for providing the history of their use of the Internet to unknown and unseen 3rd-parties who are apparently making money from that data collection.

My underestimation of the determination of the trackers is revealed by this next story to which I just shake my head:

The Tor project uses a version of Firefox as its browser. And the Tor folks have gone to, and continue going to, extreme lengths to protect the privacy which their users VERY clearly want.

We know that JavaScript running on a webpage -- and specifically in an advertisement -- is able to query for the size of the browser's display window. This is super-useful for making "Responsive" websites which dynamically alter their appearance to deliver a good experience regardless of viewport size.

But, the instantaneous size of the browser's window can also a "signal" that leaks identifying and deanonymizing information to webpage scripts. This is especially true when a user has their browser's window at some intermediate height and width rather than running edge to edge. In that case the JavaScript will see a much more unique and arbitrary height and width that could provide substantial disambiguating information to anyone with tracking intent.

As we've discussed in the past, from time to time the mainstream Firefox browser has been back-porting various privacy enhancements developed and extensively tested for the Tor version of Firefox. This will happen again with release #67 of Firefox, slated for mid-May, two months from now, when Firefox will inherit optional "letterboxing". Letterboxing masks the browser window's true dimensions by rounding the window width and height down to multiples of 200px and 100px, respectively, during window resizing to generate much less unique window dimensions for its users. And during dragging, "gray space" will be added at the top, bottom, left, or right of the current page as needed to fill in the gaps.

If that sounds like something you don't want (and I'm certainly not as concerned as perhaps I should be) the good news is that this new behavior will be a capability only and will be disabled by default. But once we arrive at release 67, setting "privacy.resistFingerprinting" to True in Firefox's "about:config" page will enable this letterboxing to give tracking scripts one less exact measure to track.

https://bugzilla.mozilla.org/show_bug.cgi?id=1407366

<quote>
Window dimensions are a big source of fingerprintable entropy on the web. Maximized windows reveal available screen width and height (excluding toolbars) and fullscreen windows reveal screen width and height. Non-maxmimized windows can allow strong correlation between two tabs in the same window. While bug 1330882 takes care of new window [creation], it would be ideal to protect windows continuously even if users resize or maximize their window or enter fullscreen.

For Tor Browser, we experimented with an approach to dynamically round the viewport (content rectangle) dimensions to multiples of 200 x 100. In our prototype, when a user drags the corner of a window, the viewport "snaps" to the largest contained multiple of 200 x 100, leaving a temporary margin of empty gray space in the window chrome. Then, when the user stops resizing (the "resize end" event), the margin of the window "shrinks" to nothing so that the outer chrome tightly encloses the viewport again.

When the user maximizes the window, the largest possible viewport is used, again a multiple of 200 x 100. Empty gray margins in the chrome part of the window cover the rest of the screen. Similarly, in fullscreen, the viewport is again given dimensions a multiple of 200 x 100, and the chrome areas around it are set to black.

Finally, an extra zoom was applied to the viewport in fullscreen and maximized modes to use as much of the screen as possible and minimize the size of the empty margins. In that case, the window had a "letterbox" (margins at top and bottom only) or "pillbox" (margins at left and right only) appearance. window.devicePixelRatio was always spoofed to 1.0 even when device pixels != CSS pixels.
</quote>

So this makes it clear that while dragging the browser windows the content size will "jump" in multiples of 200 for width and 100 for height. And that when released the browser's exterior window size will snap down to fit snugly around the "rounded down" interior window size.

Chrome on Windows 7 = a 0-day
Week before last on Wednesday February 27th, and coming to light a week later, last Wednesday, security researcher Clement Lecigne, who is with Google's Threat Analysis Group, discovered and reported a high severity vulnerability in Chrome that could allow remote attackers to execute arbitrary code and take full control of the computers.

... and Google has warned that this zero-day remote code execution vulnerability is being actively exploited in the wild to attack Chrome users.

The vulnerability was assigned as CVE-2019-5786, and affects the web browsing software for all major operating systems including Windows, macOS, and Linux.

Until the majority us Chrome's users have been auto-updated the the Chrome security team is deliberately keeping all technical details to themselves. They have only stated that the trouble is a use-after-free vulnerability in the FileReader component of the Chrome browser, which leads to remote code execution attacks.

("Use-after-free" refers to memory that was temporarily allocated by the operating system for some purpose. After it is no longer needed it is released back to the operating system... but due to a coding error in the application, an accessible pointer to that memory persisted and could be used by a sufficiently clever attacker to execute code of the attacker's choosing.)

Google security said: "Access to bug details and links may be kept restricted until a majority of users are updated with a fix," the Chrome security team notes. "We will also retain restrictions if the bug exists in a third party library that other projects similarly depend on, but haven’t yet fixed."

The glitch is in "FileReader", a standard API that has been designed to allow web applications to asynchronously read the contents of files (or raw data buffers) stored on a user's computer, using 'File' or 'Blob' objects to specify the file or data to read.

And then, the next day, last Thursday, we further learned that the active in-the-wild attack was actually leveraging a pair of 0-days, the one in Chrome and also a previously unknown 0-day existing in Windows 7 but not affecting Windows 10.

https://security.googleblog.com/2019/03/disclosing-vulnerabilities-to-protect.html

On Wednesday, February 27th, we reported two 0-day vulnerabilities — previously publicly-unknown vulnerabilities — one affecting Google Chrome and another in Microsoft Windows that were being exploited together.

To remediate the Chrome vulnerability (CVE-2019-5786), Google released an update for all Chrome platforms on March 1; this update was pushed through Chrome auto-update. We encourage users to verify that Chrome auto-update has already updated Chrome to 72.0.3626.121 or later.

The second vulnerability was in Microsoft Windows. It is a local privilege escalation in the Windows win32k.sys kernel driver that can be used as a security sandbox escape. The vulnerability is a NULL pointer dereference in win32k!MNGetpItemFromIndex when NtUserMNDragOver() system call is called under specific circumstances.

We strongly believe this vulnerability may only be exploitable on Windows 7 due to recent exploit mitigations added in newer versions of Windows. To date, we have only observed active exploitation against Windows 7 32-bit systems.

Pursuant to Google's vulnerability disclosure policy, when we discovered the vulnerability we reported it to Microsoft. Today, also in compliance with our policy, we are publicly disclosing its existence, because it is a serious vulnerability in Windows that we know was being actively exploited in targeted attacks. The unpatched Windows vulnerability can still be used to elevate privileges or combined with another browser vulnerability to evade security sandboxes. Microsoft have told us they are working on a fix.

So... today is patch Tuesday and I haven't yet had the chance to see whether a fix for this 32-bit Win7 flaw might have made it into today's patches. Certainly, given the fact that it's known and being exploited in the wild, it would have been given some priority by Microsoft.

Citrix is not having a good year

Last Wednesday Citrix learned from the FBI that their network had been infiltrated. When I read the blog posting by Stan Black, their chief security and information officer (CSIO) Stan Black, it didn't seem like such a big deal. Stan wrote:

https://www.citrix.com/blogs/2019/03/08/citrix-investigating-unauthorized-access-to-internal-network/

On March 6, 2019, the FBI contacted Citrix to advise they had reason to believe that international cyber criminals gained access to the internal Citrix network.

Citrix has taken action to contain this incident. We commenced a forensic investigation; engaged a leading cyber security firm to assist; took actions to secure our internal network; and continue to cooperate with the FBI.

Citrix is moving as quickly as possible, with the understanding that these investigations are complex, dynamic and require time to conduct properly. In investigations of cyber incidents, the details matter, and we are committed to communicating appropriately when we have what we believe is credible and actionable information.

While our investigation is ongoing, based on what we know to date, it appears that the hackers may have accessed and downloaded business documents. The specific documents that may have been accessed, however, are currently unknown. At this time, there is no indication that the security of any Citrix product or service was compromised.

While not confirmed, the FBI has advised that the hackers likely used a tactic known as password spraying, a technique that exploits weak passwords. Once they gained a foothold with limited access, they worked to circumvent additional layers of security.

Citrix deeply regrets the impact this incident may have on affected customers. Citrix is committed to updating customers with more information as the investigation proceeds, and to continuing to work with the relevant law enforcement authorities.

I was a bit curious about that second-to-last sentence: "Citrix deeply regrets the impact this incident may have on affected customers." Huh? The posting said: "... it appears that the hackers may have accessed and downloaded business documents." I assumed that meant Citrix business documents?

So that was Wednesday. But then two days later, last Friday, NBC News posted the story with a somewhat different take. NBC's headline and lead-in read: "Iranian-backed hackers stole data from major U.S. government contractor. The hackers are believed to have penetrated the software giant Citrix years ago and have remained inside the company's computer network ever since."

Ouch.

NBC: "Iranian-backed hackers have stolen vast amounts of data from a major software company that handles sensitive computer projects for the White House communications agency, the U.S. military, the FBI and many American corporations, a cybersecurity firm told NBC News.

Citrix Systems Inc. came under attack twice, once in December and again Monday, according to Resecurity, which notified the firm and law enforcement authorities.

Employing brute force attacks that guess passwords, the assault was carried out by the Iranian-linked hacking group known as Iridium, which was also behind recent cyberattacks against numerous government agencies, oil and gas companies and other targets, Charles Yoo, Resecurity's president, said.

The hackers extracted at least six terabytes of data and possibly up to 10 terabytes in the assault on Citrix, Yoo said. The attackers gained access to Citrix through several compromised employee accounts, he said. "So it's a pretty deep intrusion, with multiple employee compromises and remote access to internal resources."

So between 6 to 10 terabytes of... something... was exfiltrated from Citrix. This sort of sounds as though it might be their customer's business documents, and that, yeah... Citrix would likely regret the impact this incident may have on their affected customers.

The security company was "Resecurity" and in their own reporting of this they indicated that Citrix was first notified of the presence of an Iridium APT (advanced persistent threat) last December. Resecurity wrrote: "Friday, December 28, 2018 at 10:25 AM – Resecurity has reached out to Citrix and shared early warning notification about targeted attack and data breach. Based on the timing and further dynamics, the attack was planned and organized specifically during Christmas period."

https://resecurity.com/blog/supply-chain-the-major-target-of-cyberespionage-groups/

What's odd is that Citrix's CSIO stated in his blog that "On March 6, 2019, the FBI contacted Citrix to advise they had reason to believe that international cyber criminals gained access to the internal Citrix network." ... Yet Resecurity makes it very clear that they notified Citrix of this nearly 3 months earlier. One wonders whether Citrix just ignored the first report? But in any event, a multi-year targeted 6 to 10 terabyte data exfiltration makes this a major breach or a major cyber services company.

Pen Test Partners blog posting is titled: "Gone in six seconds? Exploiting car alarms"
https://www.pentestpartners.com/security-blog/gone-in-six-seconds-exploiting-car-alarms/

The two most popular aftermarket car alarm systems in the world open their owners to hacking. (Who could have guessed that? :)

Pen Test Partners blog begins: "Key relay attacks against keyless entry vehicles are well known. Many 3rd party car alarm vendors market themselves as solutions to this. We have shown that fitting these alarms can make your vehicle EVEN LESS SECURE! These alarms can expose you to hijack, may allow your engine to be stopped while driving, and it may even be possible to steal vehicles as a result."

The car alarm systems are those by Pandora and Viper (Viper is "Clifford" in the UK). They have been found to be vulnerable to remote exploitation, enabling attackers to hijack the vehicles they're installed on and to spy on their owners.

The exploitable software flaws were found in the smartphone apps used to control the alarm systems developed by Pandora and Viper (known as Clifford in the UK), two of the most popular smart car alarms worldwide.

The controlling smartphone app has been downloaded more than 3 million times.

Pen Test Partners who poked that the smartphone app aid that "the vulnerabilities are relatively straightforward insecure direct object references in the API." They said that "simply by tampering with parameters, one can update the email address registered to the account without authentication, send a password reset to the now-modified attacker's eMail address, and take

over the account."

This would allow attackers to:

• Geo-locate the car in real time
• Disable the alarm
• Unlock the car
• Enable or disable the vehicle immobiliser
• In some cases, the car engine could be ‘killed’ while driving
• One of the two alarm brands allows drivers to be ‘snooped’ on through a microphone
• And, depending upon the alarm, it may also be possible to steal vehicles

Moreover, the flaws they observed in the car alarm APIs exposed huge amounts of personally identifiable information.

They also noted that it's not necessary to purchase either of these alarm systems to obtain an account on the system. Both products allow anyone to create a trial test/demo account. From that demo account it’s possible to access any genuine account and retrieve it user's details.

<quote>

Killing car engines to order: This part is crazy!

We discovered we could kill the engine on the Viper equipped car whilst it was in motion. Promotional videos from Pandora indicate this is possible too, though it doesn’t appear to be working on our car.

The intention is to halt a stolen vehicle. Except, using the account takeover vulnerability in the mobile app, one could kill the engine of any car fitted with these alarms.

…

Audio snooping on drivers. Yes, really

The Pandora alarm has the ability to make SOS calls. A microphone is fitted in order to enable this.

The microphone can be accessed and enabled remotely owing to the authorisation flaw in the API.

Therefore all cars and drivers with the alarm fitted can be silently listened to. Millions of drivers snooped on.

…

CAN control. OMG!

Both the Viper and Pandora have the ability to send custom CAN messages. This is where things

get a bit scary…

In recent years car alarms have had the ability to interface directly with the CAN. This is necessary given the level of complexity of modern vehicles, this also helps to reduce alarm wiring and installation time required.

Higher-end alarms can automatically detect the vehicle type they are being fitted to and customise their command set to the CAN messaging being used. This speeds up installation significantly.

However, when the alarm doesn't correctly recognise the vehicle, or it isn't automatically supported, the installer will need to programme the alarm manually.

As far as we can determine, alarm programming has to be done locally using a laptop app or (more interesting) a mobile phone using Bluetooth. Whilst there appear to be methods to programme over the air from the API, they aren't documented fully and we haven't been able to fully reverse engineer them yet.

We are still working on this area, but each requires a different vehicle with an alarm fitted to prove it.

BUT

Start/Stop functionality is already enabled.

After analysing the firmware, manuals and related changelogs, we found some scary functionality. It is very vehicle-specific so we've been unable to test it all.

If you own one of these vehicles with the relevant Pandora alarm, we would love to know if it works. Obviously take great care and we don't advise doing this on public roads.

Mazda 6, Range Rover Sport, Kia Quoris, Toyota Fortuner, Mitsubishi Pajero, Toyota Prius 50 and RAV4 – these all appear to have undocumented functionality present in the alarm API to remotely adjust cruise control speed!

Some workarounds for stop/start functionality also require a false brake pedal message to be sent, simulating the driver pressing the pedal and starting the vehicle.

</quote>

We've covered the CAN bus -- and its power -- in detail. There's an explicit and deliberate firewall which separates the "infotainment" side from the critical vehicle operations side. It's clear that these extremely insecure alarm systems are being attached to the critical vehicle operations side in order to give them control they require. So, essentially, anyone adding this aftermarket alarm system is bypassing the CAN bus firewall and attaching a highly insecure and very poorly designed system to the inner guts of their car.

Fearing that bad guys might already know of these vulnerabilities and be exploiting them in the

wild, Pen Test Partners gave both companies a very short 7 days to fix the security issues. Both Pandora and Viper responded and patched them immediately.

Pen Test Partners wrote: "Pandora's UK representative responded in about 48 hours and had their Moscow-based HQ take action quickly. The insecure direct object references were fixed overnight and we confirmed that the following morning. Viper responded faster, but took a little longer to fix the vulnerability. That one is also confirmed as fixed."

They also wrote: "It's important to note that we didn't carry out a full test of the APIs; doing that would have required further authorisation which we didn't have. We have no idea if there are other vulnerabilities in the API.

In other words, they found the first big horrific flaw in the alarms and notified the vendors. What they did NOT do was perform a full in-depth security audit.

Everyone listening, who uses the Alt-Tab Windows key combination, raise your hand...
https://www.bleepingcomputer.com/news/microsoft/microsoft-asks-users-to-call-windows-10-devs-about-alt-tab-feature/

Yeah... That's about what I thought. Definitely some of us use it. But a definite minority. I mostly use it to quickly ping-pong between a pair of apps. Sometimes I'll use it to find an app that's deeper down in the MRU (most recently used) stack. But I mostly use it to toggle between a pair where I'm interacting. It's my "go back to the previous app" jump. For example, I first create the outline for each week's podcast in an outliner, ThoughtManager Desktop.

(ThoughtManager was an outliner for the Palm Pilot and it had a Windows add-on component. The Palm obviously died (aside from the collection that I still have chilling in my fridge), but ThoughtManager Desktop is the cleanest and simplest outliner I've found for Windows.)

Anyway... while researching topics, news and stories for the podcast I'll rapidly switch back and forth between Firefox and ThoughtManager grabbing bits and pieces from here and there and transporting them into the growing podcast outline.

Anyway, my point here is that Bleeping Computer had an intriguing piece about a new practice that Microsoft is exploring: Having 5 to 10 minute conversations, by appointment, with Windows 10 users about their usage of specific Windows features.

Lawrence Abrams writes:

Microsoft has started to display notifications in the Windows 10 Action Center asking users to have a phone call with Microsoft developers and provide direct feedback about the ALT+TAB feature in Windows.

While using a Windows 10 Insider build today, I was shown a Feedback Hub notification stating

that "Microsoft wants to hear your opinions! To set up a phone call with Windows engineers, go to: http://www.aka.ms/alttab". This link then redirects to a web page at https://ux.microsoft.com/?AltTab.

It is not known if this is only being shown to Windows Insiders users at this time.

When users visit this link they will be shown a Microsoft User Research page stating that a Windows 10 product team is looking to "understand our customer needs" and would like to have an anonymous 5-10 minute phone call with the user.

In this particular case, the phone call will be with Microsoft engineers to discuss how users use the ALT+TAB feature to switch between apps. Microsoft states they are performing these calls in order to get a better understanding of how a feature is being used while they are in development.

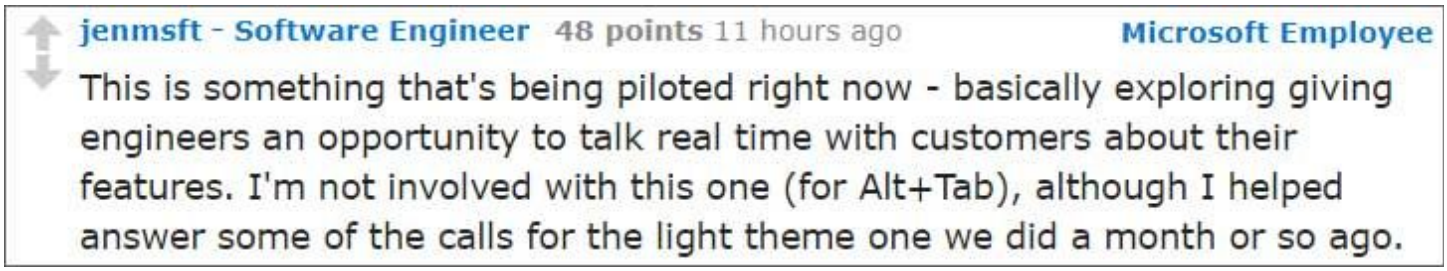
<quote>
"Your feedback is important to us. As we develop new software and services, it's critical that we get feedback from customers who use our products.

Hearing from you early in the development cycle helps us make changes to our products and test them before release.

Early customer interactions ensure we hit the mark with features. The time you spend with us today can improve our products for users around the world."
</quote>

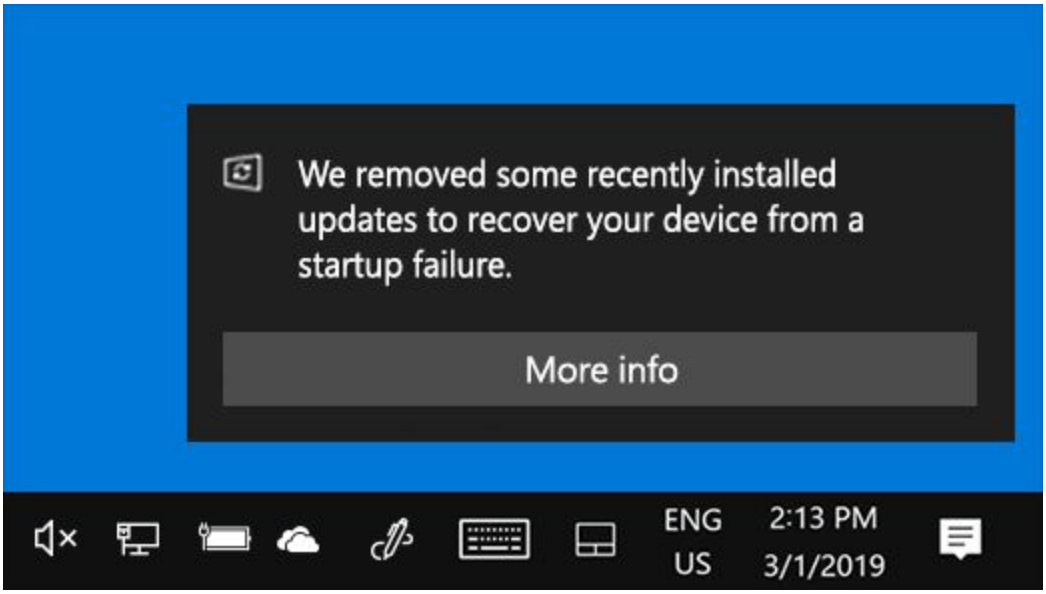
According to the web site, Windows engineers will be available on 3/11/2019 between 11:15 AM and 1:00 PM PST and on 3/12/2019 between 9:30 AM and 11:30 AM PST to schedule a call. The page goes on to say that users can expect a 5-10 minute call, but that it could last longer if there is more to discuss. They also state that the calls are not being recorded, are anonymous, and the content of the call will not be stored.

When researching this notification, I ran into a Reddit thread where a user received a similar notification over the weekend. Jen Gentleman, a Community Manager and Software Engineer at Microsoft, stated that these notifications are a new program being piloted that gives engineers the ability to talk to users in real time about features they are working on.



This is not the first time that Microsoft has asked Windows 10 users to call engineers. In February 2019, Microsoft also asked used to contact them regarding the Windows Update feature.

And in other Win10 news - self-uninstalling updates!:
Windows 10 will soon acquire the ability to autonomously uninstall updates that cause problems.



According to a support document published yesterday, Windows 10 will begin automatically uninstalling automatically installed Windows updates that cause startup failures due to incompatibility. Updates will be automatically removed when Windows detects that it has recovered from a startup failure after all other "automatic recovery attempts have been unsuccessful."

Furthermore, and to prevent any immediate repeat, Windows will blacklist them from installing for the following 30 days.

Microsoft wrote:
To ensure that your device can start up and continue running as expected, Windows will also prevent problematic updates from installing automatically for the next 30 days. This will give Microsoft and our partners the opportunity to investigate the failure and fix any issues. After 30 days, Windows will again try to install the updates.

Users who still want to install them because they believe the removed Windows updates weren't the cause behind the startup failures can do so by downloading them from the Windows Update Catalog or, in the case of Device Drivers, using the system's Device Manager.

So this feels like a useful and practical CYA move for helping to deal with the past run of troubles Windows 10 users have experienced after Windows updates messed up their systems. (And it's why I have next system set to wait a month in any event.)

Marriott CEO shares post-mortem on last year's hack
Today we know more about what happened with the breach at Marriott's Starwood properties Marriott investigators found Mimikatz and a remote access trojan (RAT) on hacked Starwood IT system.
https://www.zdnet.com/article/marriott-ceo-shares-post-mortem-on-last-years-hack/

Last Thursday, Marriott International's CEO Arne Sorenson testified in front of a US Senate subcommittee. His testimony revealed previously unknown details about the infamous breach.

Speaking in front of the Senate Committee on Homeland Security & Governmental Affairs Permanent Subcommittee on Investigations, Sorenson apologized to the company's customers but also shot down rumors that China was behind the hack.

In his prepared statement, Sorenson said that Marriott learned something might be wrong on September 8th of 2018 when they were contacted by Accenture, the IT company responsible for managing the Starwood guest reservation database.

As we know, Marriott had acquired the Starwood hotels properties two years before in September of 2016. Work was underway to migrate Starwood's customers over to Marriott's own guest reservation system, but at that time the Starwood system was still separate from the rest of the Marriott network.

So on September 8th, Accenture told Marriott's IT staff that a database monitoring system known as IBM Guardium had detected an anomaly on the Starwood guest reservation database the day before, on September 7th.

Sorenson said that "The Guardium alert was triggered by a query from an administrator's account to return the count of rows from a table in the database."

Such queries are considered dangerous because the software that runs on top of a database doesn't usually need to make them. This meant that a human operator was making this type of very specific query by hand.

Sorenson continued... "As part of our investigation into the alert, we learned that the individual whose credentials were used had not actually made the query."

At that point, the Marriott staff realized they were dealing with a probable breach, although they didn't know if it was something big or just the beginning of a hack that could be very easily contained before the attackers accessed any user data.

It brought in third-party forensic investigators on September 10th, to help its IT staff look into a possible breach. Within a week the forensic firm's investigation uncovered malware on the Starwood IT systems.

Sorenson testified that: "The investigators uncovered a Remote Access Trojan ('RAT'), a form of malware that allows an attacker to covertly access, surveil, and even gain control over a computer. I was notified of the ongoing investigation that day, and our Board was notified the following day."

Significant forensic work was required to reveal the full scope of the attack and despite the RAT's presence on the Starwood IT system, at that point, there was no evidence that unauthorized parties had accessed customer data from Starwood's guest reservation database.

But the investigation continued. A month later, in October, the forensic firm discovered Mimikatz, a well known penetration testing tool which searches a device's memory for usernames and passwords. The tool was most likely used to help hackers acquire passwords for other Starwood systems and help them move laterally to other parts of the IT network.

Still, however, investigators found no direct evidence that hackers had accessed customer data.

A month later the hack turned from "probably bad" to "bad" when, in November, investigators discovered that the hackers had been active on Starwood's IT network since July 2014. This meant that a long-term persistent attack had been underway for more than two years without ever being detected. This meant that the forensic firm had to dig through years of logs.

And, still, there was no actual evidence of attackers accessing customer data.

In mid-November the evidence was found. Sorensen's statement reads:

"On November 13, our investigators discovered evidence that two compressed, encrypted files had been deleted from a device that they were examining. The files were encrypted and the actual content was unknown. There was also evidence to suggest that those two files had potentially been removed from the Starwood network. Six days later, on November 19, 2018, investigators were able to decrypt the files, and found that one contained an export of a table from the Starwood Guest Reservation Database containing guest data, while the other contained an export of a table holding passport information."

The hotel chain then notified authorities and went public with its data breach disclosure on November 30, revealing a breach that impacted around 500 million customers.

According to Sorenson's testimony and an update on the Starwood breach notification website, the latest stats surrounding the Marriott breach, which have been updated several times as more was learned, are:

- 383 million guest records
- 18.5 million encrypted passport numbers
- 5.25 million unencrypted passport numbers (663,000 from the US)
- 9.1 million encrypted payment card numbers
- 385,000 card numbers that were still valid at the time of the breach

Sorenson stated that investigative efforts have yet to uncover evidence to suggest that hackers gained access to the encryption key used to encrypt the 9.1 million payment card numbers.

So it's likely that some Starwood employee, somewhere, fell victim to a phishing attack which allowed something malicious to get into their machine. That might have been a remote access Trojan that phoned home to report its success and then gave some remote attacker visibility into

that machine. Form there the attacker likely perused the connected network and may have moved laterally into other machines. By downloading, installing and running MimiKatz in various machines vestigial credentials could have been recovered from RAM and then used to further penetrate the organization's network, servers and services.

Miscellany

From Elaine, According to Merriam-Webster...
While it's often maintained that the word/doozy/derives from the "Duesenberg" in the name of the famed Duesenberg Motor Company, this is impossible on chronological grounds./Doozy/was first recorded (in the form/dozy/) in eastern Ohio in 1916, four years before the Duesenberg Motor Company began to manufacture passenger cars; the related adjective/doozy,/meaning "stylish" or "splendid," is attested considerably earlier, in 1903. So where did/doozy/come from? Etymologists believe that it's an altered form of the word/daisy,/which was used especially in the late 1800s as a slang term for someone or something considered the best.

SpinRite

Jamie
Location: Sydney, Australia
Subject: Spinriting my iPod - working on one spot for hours and hours and hours
Date: 28 Feb 2019 20:52:48
:
Steve,

A bit of a story - there is a question in there somewhere. I promise!

I've had a copy of Spinrite for ages. I purchased it many years ago soon after having discovered the SecurityNow podcast. (At the time, I went back and started from number 1, so ... I've heard 'em all!)

I've used Spinrite quite a few times around the house, I've even had to re-download it once or twice over the years as it is a very small exe/iso that is very easy to lose.

Recently I inherited an iPod. (That's me justifying a short iPod Classic 'phase' I went through where I picked up a few on eBay ;-)) I didn't realise at the time but the hard drive had issue. I mean, how are you to know, really?

Oh well ... I figured I'd 'fix it with Spinrite'. Didn't seem too much of an issue...

https://kevinstreet.co.uk/2017/09/11/running-spinrite-6-0-on-macos/

Following this guide, I managed to get Spinrite running in VirtualBox, connecting to the iPod in Disk Mode. Very cool that we can do that at all!

I kicked off a Level 2 scan. As it progressed, it came up with an estimate of 33 minutes to complete. Things were chugging along so I left it running and went on with other stuff and sort of forgot about it.

Many hours later (5+ at the moment) it is still running. Spinrite is fine and I can navigate around the different screens so the VM is still going, but we are still sitting on 14%. The time estimate has not moved (we're 5 minutes in with the estimated Time Remaining is sitting at 33 minutes).

If I listen to the iPod, I can hear the hard drive in what seems like a loop. It's doing the same thing over and over again

chuck chunk chunk chunk ... squwiggle ... berrrrrp ... squeak

Over and over and over and over...

I guess this the drive trying and trying to read one troublesome spot?

And so, onto the question.

When connecting to an iPod like this (or any USB drive), do all of the special data recovery techniques you've mentioned over the years (coming at the spot from all angles etc.) still work? On the website you say "It contains and deploys an extensive arsenal of data recovery techniques and technologies designed to pull a drive's data back from oblivion."

Do you still have the same low level control of the drive as you would with a direct IDE connection? Are you able to do everything to a drive connect in this way as you can to something directly connected to a physical PC via IDE/SATA?

By the time you get this (and if it were to air on SecurityNow) I may well have an answer of my own. I plan to just leave this running - at least overnight, maybe the weekend.

Thanks for any answer - via SecurityNow or not. (Happy to share on SN if you're running low on Spinrite stories.)

Long time listener, enjoying every episode.

Thanks
Jamie

P.S. Could we bring back a few of the 'deep dive' style episodes where you 'go deep' and tell us all about something?

SPOILER:
Speculative Load Hazards Boost Rowhammer and Cache Attacks

https://arxiv.org/pdf/1903.00446.pdf

Researchers from the Worcester Polytechnic Institute in Worcester, Massachusetts, USA and the University of Lübeck in Lübeck, Germany named their work "SPOILER", and I saw a comment somewhere that they were too tired afterward even bother trying to reverse engineer that to make it into some clever abbreviation.

So "SPOILER" is "Speculative Load Hazards Boost Rowhammer and Cache Attacks" and, as the buzz words embedded in that title suggest, it successfully combines processor architecture speculation vulnerabilities with DRAM row hammering cache attacks.

Their Abstract is going to make everyone's eyes cross and is guaranteed to stop the propellers spinning on the top of anyone beanies. But that's also kind of the point I want to make. I want to show how the subtly and nuance of attacks such as Rowhammer and Speculation evolve over time and how far out into the weeds academic security research can and does go...

Abstract

Modern microarchitectures incorporate optimization techniques such as speculative loads and store forwarding to improve the memory bottleneck. The processor executes the load speculatively before the stores, and forwards the data of a preceding store to the load if there is a potential dependency. This enhances performance since the load does not have to wait for preceding stores to complete. However, the dependency prediction relies on partial address information, which may lead to false dependencies and stall hazards. In this work, we are the first to show that the dependency resolution logic that serves the speculative load can be exploited to gain information about the physical page mappings. Microarchitectural side-channel attacks such as Rowhammer and cache attacks rely on the reverse engineering of the virtual-to-physical address mapping. We propose the SPOILER attack which exploits this leakage to speed up this reverse engineering by a factor of 256. Then, we show how this can improve the Prime+Probe attack by a 4096 factor speed up of the eviction set search, even from sandboxed environments like JavaScript. Finally, we improve the Rowhammer attack by showing how SPOILER helps to conduct DRAM row conflicts deterministically with up to 100% chance, and by demonstrating a double-sided Rowhammer attack with normal user’s privilege. The later is due to the possibility of detecting contiguous memory pages using the SPOILER leakage.

What that just said was that these guys came up with yet another exploit of speculation, not in the execution path of branch and jump prediction, but in a modern processor's intrinsic memory fetching pipeline which is always attempting to aggressively prefetch as much main memory into the processor's local caches as possible. In other words, "prefetching", which every modern processor does in order to keep its execution units fed, can be viewed -- and exploited -- as yet another form of Speculation.

<quote>
Spectre attacks on the speculative branch prediction unit imply that side channels such as caches can be used as a primitive for more advanced attacks on speculative engines. Speculative engines predict the outcome of an operation before its completion, and they enable execution of the following dependent instructions ahead of time based on the prediction. As a result, the pipeline can maximize the instruction level parallelism and resource usage. In rare cases where the prediction is wrong, the pipeline needs to be flushed resulting in performance penalties. However, this approach suffers from a security weakness, in which an adversary can fool the predictor and introduce arbitrary mispredictions that leave microarchitectural footprints in the pipeline. These footprints can be collected through the cache side channel to steal secrets.
</quote>

<quote>
During the speculative execution of the load, false dependencies may occur due to the unavailability of physical address information. These false dependencies need to be resolved to avoid computationon invalid data. The occurrence of false dependencies and their resolution depend on the actual implementation of the memory subsystem. Intel uses a proprietary memory disambiguation and dependency resolution logic in the processors to predict and resolve false dependencies that are related to the speculative load. In this work, we discover that the dependency resolution logic suffers from an unknown false dependency independent of the 4K aliasing. The discovered false dependency happens during the 1 MB aliasing of speculative memory accesses which is exploited to leak information about physical page mappings.
</quote>

Our Contribution
We have discovered a novel microarchitectural leakage which reveals critical information about physical page mappings to user space processes. The leakage can be exploited by a limited set of instructions, which is visible in all Intel generations starting from the 1st generation of Intel Core processors, independent of the OS and also works from within virtual machines and sandboxed environments. In summary, this work:

1. Exposes a previously unknown microarchitectural leakage stemming from the false dependency hazards during speculative load operations.

2. Proposes an attack, SPOILER, to efficiently exploit this leakage to speed up the reverse engineering of virtual-to-physical mappings by a factor of 256 from both native and JavaScript environments.

3. Demonstrates a novel eviction set search technique from JavaScript and compares its reliability and efficiency to existing approaches.

4. Achieves efficient DRAM row conflicts and the first double-sided Rowhammer attack with normal user-level privilege using the contiguous memory detection capability of SPOILER.

5. Explores how SPOILER can track nearby load operations from a more privileged security domain right after a context switch.
SPOILER from JavaScript

Microarchitectural attacks from JavaScript have a high impact as drive-by attacks in the browser can be accomplished without any privilege or physical proximity. In such attacks, co-location is automatically granted by the fact that the browser loads a website with malicious embedded JavaScript code.

The browsers provide a sandbox where some instructions like clflush and prefetch and file systems such as procfs are inaccessible, limiting the opportunity for attack. Genkin et al. showed that side-channel attacks inside a browser can be performed more efficiently and with greater portability through the use of WebAssembly. Yet, WebAssembly introduces an additional abstraction layer, i.e. it emulates a 32-bit environment that translates the internal addresses to virtual addresses of the host process (the browser). WebAssembly only uses addresses of the emulated environment and, similar to JavaScript, it does not have direct access to the virtual addresses. Using SPOILER from JavaScript opens the opportunity to puncture these abstraction layers and to obtain physical address information directly.

From the end of their conclusions:

SPOILER can be executed from user space and requires no special privileges. While speculative execution enables SPOILER and Spectre and Meltdown, our newly found leakage stems from a completely different hardware unit, the Memory Order Buffer. We exploited the leakage to reveal information on the 8 least significant bits of the physical page number, which are critical for many microarchitectural attacks such as Rowhammer and cache attacks.

Broadly put, the leakage described in this paper will enable attackers to perform existing attacks more efficiently, or to devise new attacks using the novel knowledge.

Responsible Disclosure

We have informed the Intel Product Security Incident Response Team of our findings on December 1st, 2018, and they have acknowledged the receipt.
</>

Bruce Schneier's words should be ringing in our ears: "Attacks never get worse, they only ever get better."

ARM and AMD processors are not affected, at least not by this. This is purely a side effect of the specific design Intel uses in all of its CORE processor microarchitectures.

~30~