Security Now! #677 - 08-21-18

The Foreshadow Flaw

This week on Security Now!

As we head into our 14th year of Security Now!, this week we look at some of the research released during last week's USENIX Security symposium, we also take a peek at last week's Patch Tuesday details, Skype's newly released implementation of Open Whisper Systems' Signal privacy protocol, Google's Chrome browser's increasing pushback against being injected into, news following last week's observation about Google's user tracking, Microsoft's announcement of more spoofed domain takedowns, another page table sharing vulnerability, believe it or not... "Malicious Regular Expressions", some numbers on how much money CoinHive is raking in, flaws in browser and their add-ons that allow tracking-block bypasses, two closing-the-loop bits of feedback, and then a look at the details of the latest Intel Speculation disaster known as "The Foreshadow Flaw."

Security News

August's Windows Patch Tuesday retrospective
Microsoft Releases Patches for 60 Flaws—Two Under Active Attack

19 of last week's patched problems were rated critical encompassing Windows, Edge, IE, Office, the ChakraCore, .NET Framework, Exchange Server, Microsoft SQL Server and Visual Studio... and ALL of them allowed remote code execution when successfully exploited.

Two of the vulnerabilities patched were publicly known and being exploited in the wild at the time of release. So, this remains true for any still-unpatched machines.

The first vulnerability under active attack (IE Memory Corruption CVE-2018-8373) is a critical remote code execution vulnerability in IE versions 9, 10 and 11 across all versions of Windows that allows remote attackers to take control of the vulnerable systems by convincing users to view a specially crafted website through Internet Explorer.

In its advisory Microsoft noted that: "An attacker could also embed an ActiveX control marked ‘safe for initialization’ in an application or Microsoft Office document that hosts the IE rendering engine." and in that way indirectly invoke IE.

The second publicly known and actively exploited flaw resides in the Windows Shell as a consequence of improper validation of file paths. This is Windows Shell RCE CVE-2018-8414. This allows arbitrary code to be executed on targeted systems by convincing victims to open a specially crafted file received via an email or a web page.

The SQL Server problem affects SQL Server 2016 and 2017, both which are vulnerable to a buffer overflow vulnerability that could be exploited remotely by an attacker to execute arbitrary code in the context of the SQL Server Database Engine service account. It requires an attacker to be able to submit a query to an affected server which might be tricky to arrange. But we know that hackers are devilishly clever.

Windows 10 also has a PDF RCE (CVE-2018-8350). Win10 systems with Edge as the default browser can be compromised merely by a user viewing a website. Due to mistakes in handling objects in the memory, Windows 10's PDF library can be exploited by a remote attacker to execute arbitrary code on the targeted system. Microsoft further indicated that: "The attacker could take advantage of compromised websites or websites that accept or host user-provided content or advertisements, by adding specially crafted PDF content to such sites." So, in other words, under Win10 and Edge, an advertisement could execute arbitrary attacker-supplied code on the machine. Unless and until patched, Win10 systems using Edge can be compromised by viewing a website."

The Exchange Server vulnerability (Exchange 2010, 2013 & 2016) is worrisome since it would support attacks targeting specific enterprises. The memory corruption vulnerability (CVE-2018-8302) allows a remote attacker to run arbitrary code in the context of the System user just by sending a specially crafted email to the vulnerable Exchange server.

The Windows 7, 8.1 and 10 as well as Servers 2012 and 2016 graphics subsystems contain a vulnerability (CVE-2018-8344) in their font processing library due to improper handling of specially crafted embedded fonts. On any of those unpatched machines, attackers can take control simply by serving a maliciously embedded font via a deliberately crafted website or document.

And, believe it or not, we're still having problems (CVE-2018-8345) with those pesky .LNK (link) shortcut files. A remote code execution vulnerability exists in all versions of Windows that allows remote code execution if a .LNK file is processed. An attacker could present the user a removable drive, or remote share, that contains a malicious .LNK file and an associated malicious binary. When the user opens this drive (or remote share) in Windows Explorer, or any other application that parses the .LNK file, the malicious binary will execute code of the attacker’s choice, on the target system.

And, not be left out of a good Patch Tuesday... we have Adobe:
Last Tuesday released security patchs for 11 vulnerabilities, two which are rated as critical for Acrobat and Reader.

The other affected and patched products are, of course, Flash Player, but also Creative Cloud Desktop, and Experience Manager.

Happily, none of the 11 vulnerabilities patched were either publicly known nor known to be actively exploited in the wild.

Skype's implementation of the Signal protocol
Last week, Skype's announced and expected support for "Skype Private Conversations" quietly went live.
https://support.skype.com/en/faq/FA34824/what-are-skype-private-conversations

The "Chat +" button now offers this for the first time.

Back in January (on the 11th) Microsoft & Signal announced their partnership:
Signal.org: "Signal partners with Microsoft to bring end-to-end encryption to Skype"
https://signal.org/blog/skype-partnership/
"Microsoft joins a growing list of organizations including WhatsApp, Google, Facebook, and Signal itself that have integrated the open source Signal Protocol into their messaging platform."

On Android, SPC was added to v8.15.0.306 on August 2.
On Windows, SPC appeared in v8.28.0.41.

I'm always uncomfortable when key management is taken away from the user and is performed by the system. As we know, that's the weakness inherent in Apple's iMessage system. They manage the keys on behalf of the user. But, in reality, it's so trivial to capture keystrokes on the sending system, or to monitor the entire decrypted conversation at either end, that users should understand that

We now have bullet-proof Crypto. So crypto technology is not the problem. Now it's always the implementation and the hosting environment.

If I absolutely needed to exchange encrypted messages I'd use a dedicated and factory-fresh machine for only that one purpose.

Someone looking for secure messaging ought to obtain a iPhone, load nothing but Signal on it and use it for nothing -- ever -- other than secure messaging.

Signal clients now exist for the desktop, too. And once they become built into Linux distros, booting a Linux CD and running Signal from it would be strongly secure.

But in any event... Signal on Skype is a nice step forward and Microsoft likely doesn't want to be left with the only non-Signal secure messaging platform. But the only real assurance any of this provides is protection from eavesdropping on the network. The endpoints remain insecure and largely unsecurable.

Google's Chrome browser becoming more proactive against code injection instabilities
https://www.bleepingcomputer.com/news/google/google-chrome-showing-alerts-about-incompatible-applications/

Last year we noted that Google had announced to developers that it was going to be moving to make Chrome less tolerant of having code and hooks injected into its browser processes.

Software is soft, and hooking is a longstanding practice used when some sort of add-on, like an A/V facility, desires access to some of a system's internals which that system is not explicitly exposing via an API. So the add-on just digs in and gets what it wants. But this has consequences...

In April of this year Google began alerting users after browser crashes caused by third-party injected code: "Update or remove problem applications" / The following application could be preventing Chrome from working properly..."

With release #68 of Chrome last month, the next phase of this gradual tightening continued. The "advisory" nature of the previous warnings has been elevated to demands:
"Update or remove incompatible applications."

In samples of the Chrome notices, "MalwareBytes" and "BitDefender Total Security" were shown.

The next phase of this tightening is slated for January 2019 with the release of Chrome 72 which will begin automatically blocking third-party code injection.

BleepingComputer writes: Since this feature was enabled in July, there have been an increasing number of reports about antivirus software being listed as incompatible applications by Chrome. Some of the antivirus applications that we have seen reported as incompatible applications include Malwarebytes, Bitdefender, Eset, Emsisoft, and AVG, IOBit, Avast.
Strangely, there are many other programs that are also being listed as incompatible applications

such as TortoiseGit 2.6.0.0 (64 bit), TortoiseSVN 1.10.0.28176 (64 bit), Stardock, Acronis True Image, Dropbox, FileZilla, Acer Power Manager software, and RocketDock. While antivirus software I [Lawrence Abrams] can understand being listed, some of these programs are a bit surprising.

According to a Google dev who posted to a Google help forum question regarding these alerts, they have no way of determine programs which are innocently injecting code or maliciously, and will continue to report on all of them.

"Chrome dev here. This is related to a new feature that aims to prevent third party software from injecting into Chrome's processes and interfering with its code. This type of software injection is rampant on the Windows platform, and causes significant stability issues (crashes). The Microsoft Edge browser already does this kind of blocking, and we are in the process of making Chrome behave similarly. What you are seeing is the warning phase of year-long effort to enable blocking, originally announced in November 2017.

Since it is effectively impossible for Chrome to automatically determine whether any particular piece of software is innocently injecting or purposefully injecting and interfering with Chrome code. To keep things simple we warn about all injected software, without making value judgments. Note that soon we will actually start blocking software from injecting, at which point this warning will cease to show. Note that you should only be seeing these crashes if you manually navigate to the chrome://settings/incompatibleApplications page, or on a startup after the Chrome browser has crashed. Additionally, this feature is currently considered experimental so not all users will see these warnings."

I'm now running Win7 and I'm glad to have Windows Defender's built-in real-time protection watching my back. IMO, Chrome SHOULD block modification of its internals, but if it's going to do that it should also provide an officially supported means for third-party A/V tools to obtain the access they believe they need so that hooking and code injection will no longer be needed.

Speaking of Google, there's news on the Tracking issue front...
Following up on our discussion last week of the AP research, which I noticed Jeff and Stacy concurred with on last week's "This Week in Google", Google has been hit with the first class-action lawsuit from a man in San Diego and the Electronic Privacy Information Center (EPIC) sent a three-page letter to the US Federal Trade Commission (FTC), noting that Google's "deceptive trade practice" is in clear violation of the 2011 settlement with the agency. EPIC wrote:

"Google is not permitted to track users after they have made clear in their privacy settings that they don't want to be tracked. The FTC's failure to enforce its Consent Orders places American consumers at risk. The Commission's inactions have made the Internet less safe and less secure for users and consumers."

As part of its 2011 settlement with the FTC, Google agreed to not misrepresenting its practices related to (1) the purposes for which it collects and uses covered information, and (2) the extent to which consumers may exercise control over the collection, use, or disclosure of covered information.
Microsoft shuts down six Russian "Fancy Bear" fraudulent spoofing web domains:

I caught an interview this morning of Brad Smith, Microsoft's President and Chief Legal Counsel. He was being interviewed due to Microsoft's just-announced takedown of six spoofing domains:

This morning Microsoft said that the APT28 hacking group - also known as Strontium, Fancy Bear, Sofacy, Sednit, and Pawn Storm, which is believed to be tied to the Russian government - had established at least six fake websites related to US Senate and conservative organizations to trick its visitors and hack into their computers.

my-iri.org
hudsonorg-my-sharepoint.com
senate.group
adfs-senate.services
adfs-senate.email
office365-onedrive.com

Microsoft's Digital Crimes Unit disabled the fake websites last week after obtaining court approval last year. This allowed Microsoft to seize the fake domains created by APT28 before they were used in any successful attacks. Microsoft has used the courts a dozen times since 2016 to shut down a total of 84 fake websites created by APT28.

During the Aspen Security Forum last month, Microsoft VP Tom Burt said the company also took down a fake domain registered by APT28, after discovering that it was established for phishing attacks against at least three congressional candidates.

"Turning [the page] Tables" on OS kernel protections
https://blog.ensilo.com/bypassing-kernel-mitigations
https://cdn2.hubspot.net/hubfs/487909/Turning%20(Page)%20Tables_Slides.pdf

Meanwhile, during the recent "BSides" privacy and security conference in Las Vegas, researchers with Ensilo Cybersecurity revealed a new responsibly-disclosed hack against Windows virtual memory page table management which may ring a few bells...

Several years ago we covered one of the RowHammer attacks which cleverly leveraged the "page coalescing" that virtual machine environments use to huge advantage.
<<Explain/Remind about page page coalescing>>

Windows was built in a memory-scarce environment and that legacy continues today. What was once a clever way of reusing memory among processes now creates new vulnerabilities.

ole32.dll, user32.dll -- find and exploit a vulnerability in code that's mapped into your own process and you have a foothold into all other processes which have the same DLL mapped into their processes!

Using a Regular Expression to freeze the web
During last week's 27th USENIX Security Symposium, held in Baltimore, Maryland, a group of researchers updated the world on the state of the art in "ReDos" attacks against servers.

Though the potential for these attacks have been known and appreciated for about six years, since 2012, the significant increased use of Server-Side JavaScript on web sites over the past six years has made their impact much more significant.

Believe it or not, "ReDos" stands for "Regex Denial of Service"... and involves remote attackers deliberately submitting super-hairy strings which present "worst case" complexity for Regular Expression parsers.

Here's what WikiPedia has to say about ReDos attacks: https://en.wikipedia.org/wiki/ReDoS

"The regular expression denial of service (ReDoS) is an algorithmic complexity attack that produces a denial-of-service by providing a regular expression that takes a very long time to evaluate. The attack exploits the fact that most regular expression implementations have exponential time worst case complexity: [which is to say that] the time required can grow exponentially relative to the input size. An attacker can thus cause a program to spend an unbounded amount of time processing by providing such a regular expression, either slowing down or becoming unresponsive.

Welcome to "Malicious Regular Expressions" !!

<<what are regular expressions?>>

Freezing the Web: A Study of ReDoS Vulnerabilities in JavaScript-based Web Servers
http://mp.binaervarianz.de/ReDoS_TR_Dec2017.pdf

Freezing the Web: A Study of ReDoS
Vulnerabilities in JavaScript-based Web Servers

Abstract
Regular expression denial of service (ReDoS) is a class of algorithmic complexity attacks where matching a regular expression against an attacker-provided input takes unexpectedly long. The single-threaded execution model of JavaScript makes JavaScript-based web servers particularly susceptible to ReDoS attacks. Despite this risk and the increasing popularity of the server-side Node.js platform, there is currently little reported knowledge about the severity of the ReDoS problem in practice. This paper presents a large-scale study of ReDoS vulnerabilities in real-world web sites. Underlying our study is a novel methodology for analyzing the exploitability of deployed servers. The basic idea is to search for previously unknown vulnerabilities in popular libraries, hypothesize how these libraries may be used by servers, and to then craft targeted exploits. In the course of the study, we identify 25 previously unknown vulnerabilities in popular modules and test 2,846 of the most popular websites against them. We find that 339 web sites (11% of the ones that use Express, a popular server-side JavaScript framework) suffer from at least one ReDoS vulnerability and some even suffer from multiple ones. A single request can block a vulnerable site for several seconds, and sometimes even much longer, enabling denial of service attacks that pose a serious threat to the availability of these sites. We also show that the

fact whether a website is vulnerable is independent of its popularity, indicating that the problem requires attention across a wide spectrum of web providers. Our results are a call-to-arms for developing techniques to detect and mitigate ReDoS vulnerabilities in JavaScript.

///

Using this methodology, we identify 339 websites that suffer from at least one ReDoS vulnerability. Based on experiments with locally installed versions of the vulnerable server-side libraries, attacking these websites with crafted inputs can cause a web server to remain unresponsive for several seconds or even minutes. These problems are due to a very small number of vulnerabilities, with a single vulnerability that causes 241 sites to be exploitable. While this is encouraging from a mitigation point of view, it also implies that an attacker aware of a single, previously unknown vulnerability can cause serious harm to vulnerable websites.

Some of the vulnerabilities we identify are more serious that the others. For one of them, 50 characters of carefully created input can block the main server thread for 10 minutes, while for most of the others at least 10,000 characters are needed to trigger one second of slowdown. Since we use HTTP header values to transport payloads, limiting the size of the header can serve as a first line of defense. We find that many websites implement this defense mechanism: 85% of the websites reject headers longer than 25,000 characters and 3% even reject headers longer than 500 characters. However, limiting the header size alone is insufficient to defend against ReDoS because even millisecond-level matching times can be used to severely degrade the performance of a website, and because there are other ways to transport payloads.

In 2012, researcher first identified ReDoS as a threat for the Node.js platform. Davis et al. confirm that such problems exist in popular modules and report that 5% of the security vulnerabilities identified in Node.js are ReDoS.

There's money in offering unwanted in-browser mining...
https://arxiv.org/pdf/1808.00811.pdf

Four German researchers recently published a paper titled "Digging into Browser-based Crypto Mining"

What they found was interesting because they were able to put some numbers to what we've been talking about in various forms for the past year...

ABSTRACT
Mining is the foundation of blockchain-based cryptocurrencies such as Bitcoin rewarding the miner for finding blocks for new transactions. Monero, a recent alternative currency enables mining with standard hardware in contrast to special hardware (ASICs) as often used in Bitcoin, paving the way for browser-based mining as a new revenue model for website operators. In this work, we study the prevalence of this new phenomenon. We identify and classify mining websites on a large corpus of websites and present a new fingerprinting method which finds up to a factor of 5.7 more miners than publicly available block lists. Our work identifies and dissects Coinhive as the major browser-mining stakeholder. Further, we present a new method to associate mined blocks in the Monero blockchain to mining pools and uncover that Coinhive

currently contributes 1.18% of mined blocks turning over Moneros worth a quarter of a million USD per month.

The hardware imbalance and the consequential high difficulty to mine Bitcoin renders its browser-based mining inefficient and motivates the use of, e.g., Monero as an alternative currency that can be efficiently mined on CPUs and thus browsers. Given its design, Monero has been adopted by websites (e.g., The Piratebay or a video streaming service with subsequent media exposure) and even among botnets to mine the currency on millions of compromised hosts. To ease browser mining, APIs exist, e.g., for in-game financing, link forwarding, captchas, during video streaming or even as an entry fee for parties.

Our work identifies Coinhive as a widely used service which provides a framework for embedding a Monero miner into a website. While these frameworks enable to mine without the users' knowledge, other services (Authedmine) actively ask users for their consent to do so as an alternative to displaying ads. Besides media reports, little is known about the ubiquity and use of browser-based mining.

Given these new possibilities, we provide a first in-depth study of the prevalence and economics of browser-based mining as a new web business model. We base this perspective on crawls of the set of .com/.net/.org domains and the Alexa Top 1M list to first identify sites using browser-based mining enabling to create a new fingerprinting method to identify mining code. Second, we dissect the short link service of the largest web-mining stakeholder Coinhive and screen their market power and profits.

To the tune of $250,000 per month!

Bypassing anti-Tracking and Ad Blockers
https://www.usenix.org/system/files/conference/usenixsecurity18/sec18-franken.pdf
https://wholeftopenthecookiejar.eu/static/tpc-paper.pdf

Also last week diring the USENIX Security Symposium, three researchers from KU Leuven, the University in the town of Leuven, Flanders, Belgium presented their paper titled: "Who Left Open the Cookie Jar? A Comprehensive Evaluation of Third-Party Cookie Policies."

Abstract
Nowadays, cookies are the most prominent mechanism to identify and authenticate users on the Internet. Although protected by the Same Origin Policy, popular browsers include cookies in all requests, even when these are cross-site. Unfortunately, these third-party cookies enable both cross-site attacks and third-party tracking. As a response to these nefarious consequences, various countermeasures have been developed in the form of browser extensions or even protection mechanisms that are built directly into the browser.

In this paper, we evaluate the effectiveness of these defense mechanisms by leveraging a framework that automatically evaluates the enforcement of the policies imposed to third-party requests. By applying our framework, which generates a comprehensive set of test cases covering various web mechanisms, we identify several flaws in the policy implementations of the 7 browsers and 46 browser extensions that were evaluated. We find that even built-in protection

mechanisms can be circumvented by multiple novel techniques we discover. Based on these results, we argue that our proposed framework is a much-needed tool to detect bypasses and evaluate solutions to the exposed leaks. Finally, we analyze the origin of the identified bypass techniques, and find that these are due to a variety of implementation, configuration and design flaws.

A bit more...

As a direct response to the privacy threat imposed by third-party trackers and associated intrusive advertisements, a wide variety of efforts have been made. Most prominently is the emergence of dozens of browser extensions that aim to thwart their users from being tracked online. These extensions make use of a designated browser API to intercept requests and either block them or strip sensitive information such as headers and cookies.

Correspondingly, several browsers have recently introduced built-in features that aim to mitigate user tracking. For instance, Firefox in its private browsing mode will by default block third-party requests that are made to online trackers. It is important to note that the effectiveness of these anti-tracking mechanisms fully relies on the ability to intercept or block every type of request, as a single exception would allow trackers to simply bypass the policies.

In this paper, we show that in the current state, built-in anti-tracking protection mechanisms as well as virtually every popular browser extension that relies on blocking third-party requests to either prevent user tracking or disable intrusive advertisements, can be bypassed by at least one technique.

GRC's 3rd-party "cookie forensics" auditing system.

Closing The Loop

Kurt in Singapore
Subject: Lithium-Ion: Slow charge vs. fast charge
Date: 19 Aug 2018 21:46:15
:
Hi Steve,

I've been following your advice for years to always plug in my devices and keep them charged up.

One thing I couldn't get a consensus online is whether slow charging, regular charging, and fast charging has different affects on the battery. There are conflicting statements.

What would be your answer?

Sheldon T Prodanuk in Canada
Subject: Crystal disk and spinrite
Date: 20 Aug 2018 09:50:22
:
Hi Steve and Leo

Been a long time fan of security now. I have been listening for 10 yrs and have listened to every podcast from day 1 honey monkeys.

I had an old laptop that was in the process of dying and so decided to take the hard drives out, put them in my PC and use then for backup storage. Dual 500 gig 7200 drives. I use Crystaldisk info to monitor the healh of my drives since some of them are very old. Crystal disk gives me a CAUTION about relocated Sectors count being 83 and the threshold is 36.

I thought now is a good time run SpineRite on lv 4 and and see what it does. So I ran a lv4 overnight and reran CrystalDisk which still throws a caution with same warning.Is this normal and would it be wise to raid the 2 disks at Raid 1 ?

Sheldon Prodanuk

The Foreshadow Flaw

https://foreshadowattack.eu/
https://foreshadowattack.eu/foreshadow.pdf
https://foreshadowattack.eu/foreshadow-NG.pdf

Tenable:
A flaw in Intel's Software Guard Extensions implementation allows an attacker to access data stored in memory of other applications running on the same host, without the need for privilege escalation.

TrendMicro:
Foreshadow/L1TF Intel Processor Vulnerabilities: What You Need to Know

L1TF - L1 (cache) Terminal Fault

SonicWall:
Foreshadow Vulnerability (L1TF) Introduces New Risks to Intel Processors

The Hacker News:
Foreshadow Attacks — 3 New Intel CPU Side-Channel Flaws Discovered

PC World
Foreshadow attacks Intel CPUs with Spectre-like tactics (but you're probably safe)

WIRED
Spectre-Like Flaw Undermines Intel Processors' Most Secure Element

Believe it or not... Intel's SPECULATION even applies to the virtual memory management paging tables!

WikiPedia:
https://en.wikipedia.org/wiki/Foreshadow_(security_vulnerability)

Foreshadow is a vulnerability that affects modern microprocessors that was first discovered by two independent teams of researchers in January 2018, but was first disclosed to the public on 14 August 2018. [1][2][3][4][5][6][7][8][9][10][11][12][13][14][15][16]

The vulnerability is a speculative execution attack on Intel processors that may result in the loss of sensitive information stored in personal computers, or third party clouds. There are two versions: the first version (original/Foreshadow) (CVE-2018-3615) targets data from SGX enclaves; and the second version (next-generation/Foreshadow-NG) (CVE-2018-3620 and CVE-2018-3646) targets Virtual Machines (VMs), hypervisors (VMM), operating system (OS) kernel memory, and System Management Mode (SMM) memory. Intel considers the entire class of speculative execution side channel vulnerabilities as "L1 Terminal Fault" (L1TF). A listing of affected Intel hardware has been posted.

Foreshadow is similar to the Spectre security vulnerabilities discovered earlier to affect Intel and AMD chips, and the Meltdown vulnerability that also affected Intel. However, AMD products, according to AMD, are not affected by the Foreshadow security flaws. According to one expert, "[Foreshadow] lets malicious software break into secure areas that even the Spectre and Meltdown flaws couldn't crack".] Nonetheless, one of the variants of Foreshadow goes beyond Intel chips with SGX technology, and affects "all [Intel] Core processors built over the last seven years".

Foreshadow may be very difficult to exploit, and there seems to be no evidence to date (15 August 2018) of any serious hacking involving the Foreshadow vulnerabilities. Nevertheless, applying software patches may help alleviate some concern(s), although the balance between security and performance may be a worthy consideration. Companies performing cloud computing may see a significant decrease in their overall computing power; individuals, however, may not likely see any performance impact, according to researchers. The real fix, according to Intel, is by replacing today's processors. Intel further states, "These changes begin with our next-generation Intel Xeon Scalable processors (code-named Cascade Lake), as well as new client processors expected to launch later this year [2018]."

On 16 August 2018, researchers presented technical details of the Foreshadow security vulnerabilities in a seminar, and publication, entitled "Foreshadow: Extracting the Keys to the Intel SGX Kingdom with Transient Out-of-Order Execution" at a Usenix Security conference.