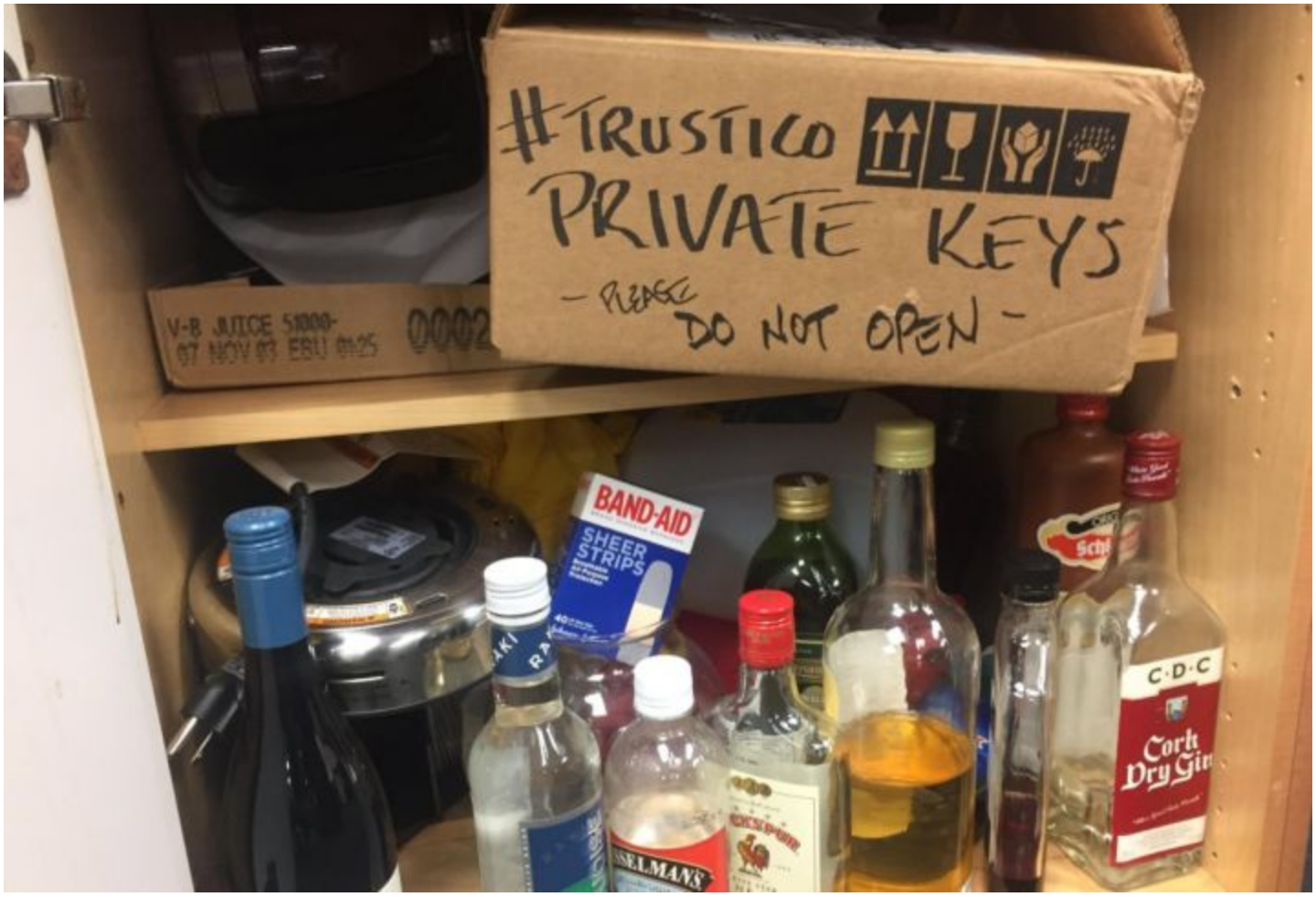
Security Now! #653 - 03-06-18

“MemCrashed” DDoS Attacks

This week on Security Now!

This week we discuss some very welcome microcode news from Microsoft, ten (yes, ten!) new 4G LTE network attacks, the battle over how secure TLS v1.3 will be allowed to be, the incredible Trustico certificate fiasco, the continually falling usage of Adobe Flash, a new and diabolical cryptocurrency-related malware, the best Sci-Fi news in a LONG time, some feedback from our terrific listeners... and a truly record smashing (and not in a good way) new family of DDoS attacks.

Our Picture of the Week



Security News

Microsoft ==WILL BE== updating Intel processor microcode for Windows users.
https://support.microsoft.com/en-us/help/4090007/intel-microcode-updates
http://www.catalog.update.microsoft.com/Search.aspx?q=KB4090007
https://blogs.windows.com/windowsexperience/2018/03/01/update-on-spectre-and-meltdown-security-updates-for-windows-devices/

- Currently only for 64 & 32 bit Windows 10 Version 1709.
- No word on Win8.1 or Win7
- Currently only:
  - Skylake H/S — 6th Generation Intel Core Processor Family — 506E3
  - Skylake U/Y & Skylake U23e — 6th Generation Intel Core m Processors — 406E3

InSpectre: I'll be updating InSpectre immediately to explain this to users, to show them their own CPU ID, and to provide a link to the Microsoft KB page to allow them to check for the availability of a processor-specific update for them.

Jump Over ASLR: Attacking Branch Predictors to Bypass ASLR
http://www.cs.ucr.edu/~nael/pubs/micro16.pdf
18 months ago…

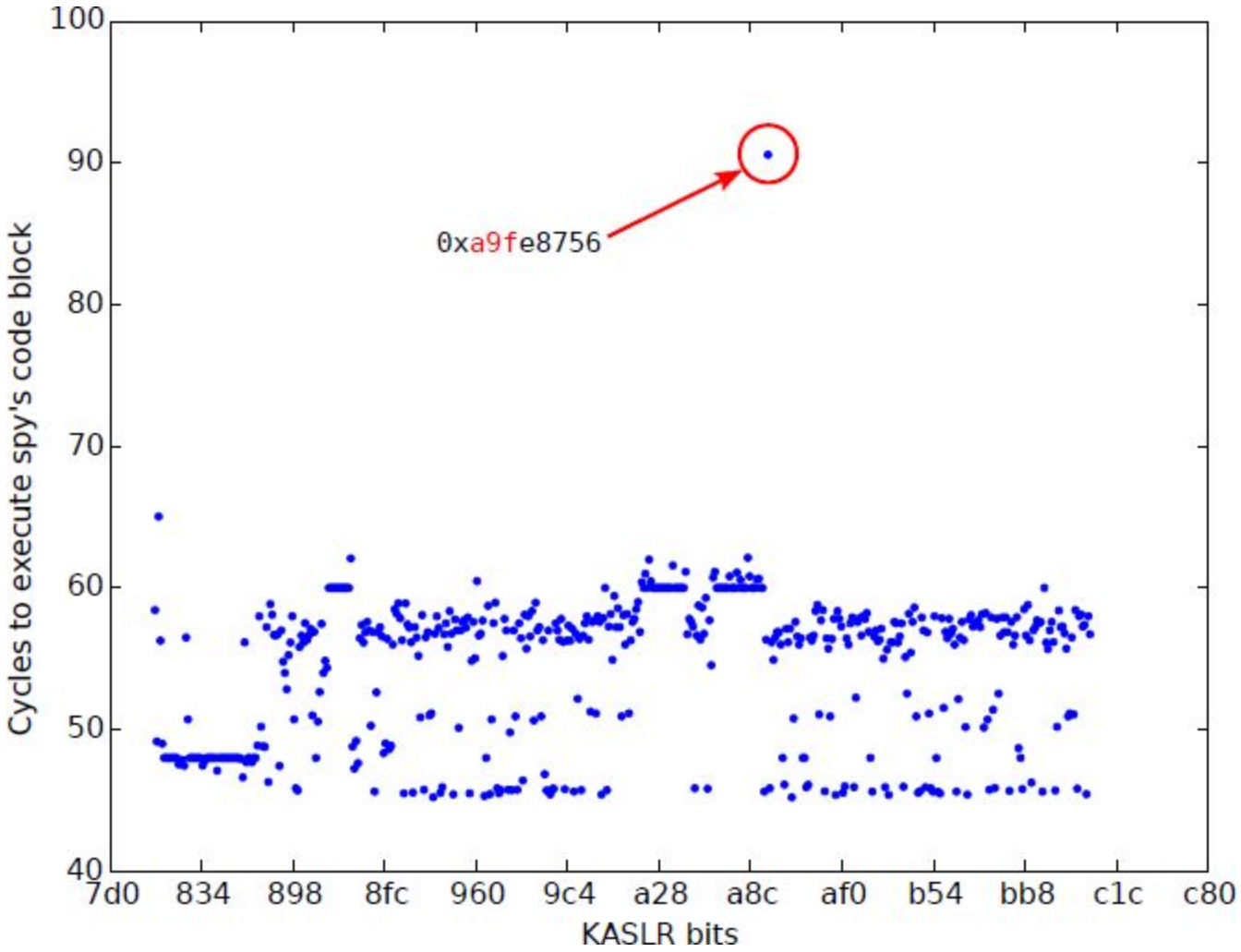
Two researchers at the State University of New York at Binghamton and another at UC Riverside have developed another side channel attack on Branch Target Buffer collisions.

<quote> Address Space Layout Randomization (ASLR) is a widely used technique that protects systems against a range of attacks. ASLR works by randomizing the offset of key program segments in virtual memory, making it difficult for an attacker to derive the addresses of specific code objects and consequently redirect the control flow to this code. In this paper, we develop an attack to derive kernel and user-level ASLR offset using a side-channel attack on the branch target buffer (BTB). Our attack exploits the observation that an adversary can create BTB collisions between the branch instructions of the attacker process and either the user-level victim process or on the kernel executing on its behalf. These collisions, in turn, can impact the timing of the attacker's code, allowing the attacker to identify the locations of known branch instructions in the address space of the victim process or the kernel. We demonstrate that our attack can reliably recover kernel ASLR in about 60 milliseconds when performed on a real Haswell processor running a recent version of Linux. Finally, we describe several possible protection mechanisms, both in software and in hardware.

[...then…]

In this paper, we demonstrate a new attack that can recover all random bits of the kernel addresses and reduce the entropy of user-level randomization by using side-channel information from the Branch Target Buffer (BTB). Our attack only requires the control of a user-level process and does not rely on any explicit memory disclosures. The key insight that makes the new BTB-based side-channel possible is that the BTB collisions between two user-level processes,

and between a user process and the kernel, can be created by the attacker in a controlled and robust manner. The collisions can be easily detected by the attacker because they impact the timing of the attacker-controlled code. Identifying the BTB collisions allows the attacker to determine the exact locations of known branch instructions in the code segment of the kernel or of the victim process, thus disclosing the ASLR offset.



Researchers uncover 4G LTE exploits that can be used to spy, spoof and cause panic
http://homepage.divms.uiowa.edu/~comarhaider/publications/LTE_NDSS18_paper.pdf

Three researchers from Purdue University and one from the University of Iowa took a good long hard look at our current 4G LTE network protocol... and found it wanting.

Abstract -- In this paper, we investigate the security and privacy of the three critical procedures of the 4G LTE protocol (i.e., attach, detach, and paging), and in the process, uncover potential design flaws of the protocol and unsafe practices employed by the stakeholders. For exposing vulnerabilities, we propose a model based testing approach: LTEInspector which combines a symbolic model checker and a cryptographic protocol verifier in the symbolic attacker model. Using LTEInspector, we have uncovered 10 new attacks along with 9 prior attacks, categorized into three abstract classes (i.e., security, user privacy, and disruption of service), in the three procedures of 4G LTE. Notable among our findings is the authentication relay attack that enables

an adversary to spoof the location of a legitimate user to the core network without possessing appropriate credentials. To ensure that the exposed attacks pose real threats and are indeed realizable in practice, we have validated 8 of the 10 new attacks and their accompanying adversarial assumptions through experimentation in a real test bed.

Their authentication relay attack allows an attacker to connect to an LTE network while spoofing another existing device's identity and location... without having legitimate credentials.

They write: "Through this attack the adversary can poison the location of the victim device in the core networks, thus allowing setting up a false alibi or planting fake evidence during a criminal investigation."

The researchers explain that the 4G LTE protocol is an "amalgamation of multiple critical procedures", each of which requires an "in-depth security and privacy analysis of its own."

Other exploits include the ability to track a victim device's location, intercept phone calls and messages and even inject fake emergency alerts. The researchers say this could create an "artificial emergency", much like the panic caused by that faulty missile alert that caused a mass scare in Hawaii in January -- though that one was, as we know, triggered by human error.

The paper presents their tool "LTEInspector" which they use to examine the LTE protocol flow. Since their paper's primary -- perhaps initial -- goal was to demonstrate how LTEInspector works, the researchers say that they don't dive into how to defend against the attacks they uncovered. However, they suggest that creating adequate defensive measures would be difficult without a significant overhaul of the 4G LTE infrastructure.

"We deliberately do not discuss defenses for the observed attacks as retrospectively adding security into an existing protocol without breaking backward compatibility often yields band-aid-like-solutions which do not hold up under scrutiny," the researchers say.

So... we've been mildly uncomfortable with the knowledge that our cellular communications system was not as robust as we would hope. It appears that it's actually much worse than that. For a determined actor, and certainly for any state-level actor, it appears to be readily attackable.

The battle over TLS v1.3 is heating up as it nears ratification

The Battle for "Forward Secrecy" in the forthcoming TLS v1.3 specification

BITS, the technology policy division of the Financial Services Roundtable.

TLS until v1.3 has not had perfect forward secrecy (PFS): Rather than computing a completely ephemeral symmetric cipher key, the server's static private key has always been used. And is today. This meant that if TLS traffic was captured and any server's private key which had been in use at the time was LATER disclosed, all of the TLS traffic that had been exchanged with the server could be retroactively decrypted.

TLS changes this so that the endpoints negotiate a fully ephemeral key for their connections which is no longer tied to the server's possibly-future-disclosed private key.

The trouble is, data centers -- and now apparently financial institutions -- are screaming that this will force the to spend money on new infrastructure to support this next level of security.

The big fight is whether TLS v1.3 should have a downgrade non-PFS option.

The purists absolutely refuse, saying that if it's there it will be used. And using it will weaken security. They also scream that this adds complexity which TLS v1.3 has deliberately removed.

(TLS v1.3 really is lovely and streamlined. It's faster since it reduces the number of startup handshakes and discards backward support for many older legacy protocols, ciphers and options.)

The big IETF meeting is this month. It's going to be interesting to see how this shakes out.

Trustico revokes 23,000 SSL certificates due to compromise

"Trustico" is (or perhaps was) a UK-based Symantec certificate reseller. This means that they do not have their own CA infrastructure. Running a CA is a big job and a big responsibility and a requires a robust network.

Just about a month ago on February 2, Trustico sent an email to DigiCert, asking DigiCert to revoke all certificates —around 50,000— it was managing after the Symantec CA business acquisition.

Trustico terminates its contract to resell Symantec (now DigiCert) certificates and initiates a partnership with Comodo.

DigiCert denies the request to mass-revoke 50,000 certificates explaining that CAB industry rules are not clear about whether a "certificate reseller" can revoke its customers' SSL certs, or only the end customer can do so alone.

DigiCert has stated that it told Trustico that they could mass-revoke certificates if there was evidence of a security incident during which the customers' private key were compromised.

Then, on February 27 -- BIZARRELY-- DigiCert receives an email from Trustico containing over 23,000 ==non-password protected== private keys for Trustico customers SSL certificates!

REMEMBER... no CA is ==ever== supposed to even HAVE their customer's private keys! The customer generates a public key pair on their end, typically on the server where it will reside, and sends ==ONLY== the PUBLIC key to the CA for the CA to sign.

But, now, in receipt of 23,000 disclosed private keys, in accordance with the CA industry rules that mandate that compromised certificates be revoked in 24 hours after a security incident, DigiCert starts the certificate revocation process for every one of the 23,000 compromised certs it received via email.

And... DigiCert sends eMails to over 23,000 Trustico customers stating that their certificates would be revoked.

Scott Helme, an information security consultant and an expert in the CA domain: “To arrive at the conclusion that Trustico have been anything other than grossly negligent here is rather difficult. Generation, storage and the apparent ease and willingness to further compromise the keys are all outrageously inappropriate. They could have trivially proven ownership of those keys without the need to zip 24k+ of them and send them via email. If these actions were motivated by business/politics as some suggest, it'd be ironic if their actions resulted in their removal as a reseller."

DigiCert reported the revocation incident on the Mozilla security mailing list, which is often used to discuss affairs of the CA/Browser Forum. Unsurprisingly, members of this mailing list had concerns regarding Trustico's access to the private keys.

Eric Mill, Senior Advisor at the U.S. General Services Administration's Technology Transformation Service posted: “Trustico doesn't seem to provide any hosting or CDN services that would make use of the private key, nor do they appear to explicitly inform users about the storage of this private key.” … thus suggesting that there was no apparent reason for Trustico to keep copies of the private keys.

Eric also wrote: “The storage of private keys appears to be done without the user's knowledge or consent. Given everything that's known, then regardless of who emailed whose customers when and why, I think it's clear that Trustico compromised those keys [...], and has been routinely compromising customer keys for years. Given that there's no evidence that Trustico [...] indicated any intent to change their business practices, then I believe it's appropriate for all CAs to immediately suspend or terminate their relationship with Trustico," Mill added.

DigiCert's Statement:

> Today, DigiCert issued the following statement regarding Trustico certificate revocation:
>
> “Trustico requested revocation of their Symantec, GeoTrust, Thawte and RapidSSL certificates, claiming the certificates were compromised. When we asked for proof of the “compromise,” Trustico did not provide details on why they were requesting the immediate revocation. Trustico’s CEO indicated that Trustico held the private keys for those certificates, and then emailed us approximately 20,000 certificate private keys. When he sent us those keys, his action gave us no choice but to act in accordance with the CA/Browser Forum Baseline Requirements, which mandate that we revoke a compromised certificate within 24 hours. As a CA, we had no choice but to follow the Baseline Requirements. Following our standard revocation process, we gave notice via email to each certificate holder whose private keys had been exposed to us by Trustico, so they could have time to get a replacement certificate.
>
> In communications today, Trustico has suggested that this revocation is due to the upcoming Google Chrome distrust of Symantec roots. That is incorrect. We want to make it clear that the certificates needed to be revoked because Trustico sent us the private keys; this has nothing to do with future potential distrust dates.

> The upcoming Chrome distrust situation is entirely separate. We are working closely to help customers with certificates affected by the browser distrust, and we are offering free replacement certificates through their existing customer portals. That process is well underway."

And remember that the certification revocation system is completely broken -- especially under Google's Chrome browser which implemented its own, never functional, CRLSET system. So suddenly we have at least 23,000 certificates whose private keys are of unknown and unknowable provenance… and no real ability to revoke them.

DigiCert is said to be planning to make the keys available to browser vendors so that they may proactively block their future use once they have been revoked.

And consider the mess this leaves those 23,000 individual customers in! Their certificate is about to be revoked due to what is clearly unethical behavior on the part of their Trustico CA who should never have had their private key in the first place and who should certainly never have mass-archived them.

My own personal wish would be that DigiCert might acquire the bulk of those users and provide them with top-tier world class CA services.

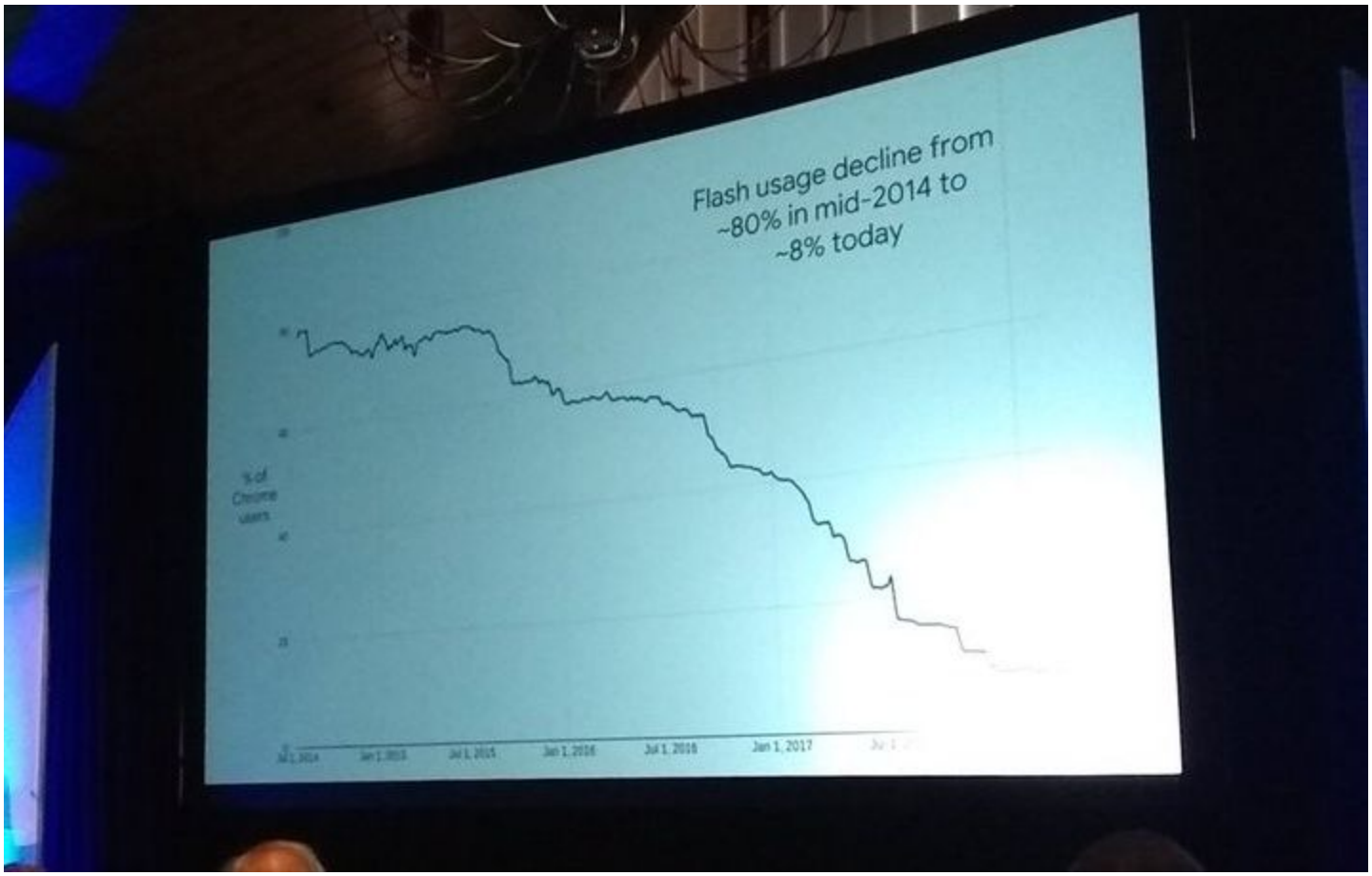
And to follow up... Trustico was performing the CSR/key generation on a JavaScript-based web page incorporating JavaScript from at least 5 or 6 different companies (including ad servers), which would potentially allow any of those third parties (intentionally or through compromise of their own) to capture generated keys.

Google Chrome: Flash Usage Declines from 80% in 2014 to Under 8% Today
https://www.bleepingcomputer.com/news/security/google-chrome-flash-usage-declines-from-80-percent-in-2014-to-under-8-percent-today/

During the keynote speech of the Network and Distributed System Security Symposium (NDSS) which was held in San Diego last week Google's Director of Engineering shared the slide on the next page… noting that the percentage of daily Chrome users who've loaded at least one page containing Flash content per day has gone down from around 80% in 2014 to under 8% in early 2018. The Keynote was focused upon the evolution of security features in Chrome. So Flash (or the lack thereof) was a topic of interest.

As we've covered previously, Adobe is planning to formally give up on Flash by the end of 2020. So only a few more years. And all of the industry's browsers -- Chrome, Firefox, Edge, etc. -- have already switched from "enabled and pray" to "click if you dare."

And, of course, since HTML5 can now play video, most advertising networks and video streaming portals have already moved away from Flash plug-ins to pure browser-based HTML5 for a superior user experience.



Someday it will just be a memory... and a big long lesson... thank goodness!

ComboJack Trojan Replaces Cryptocurrency Addresses Copied to Windows Clipboard
https://www.bleepingcomputer.com/news/security/combojack-trojan-replaces-cryptocurrency-addresses-copied-to-windows-clipboard/

ComboJack malware replaces a pasted cryptocurrency address with its own!

Researchers at Palo Alto Networks have dubbed the malware "ComboJack" because it hijacks the system clipboard, watching for a valid cryptocurrency address to be pasted... and then replaces it with its own on-the-fly. That's the "jack" part. The "Combo" part is that is can differentiate and detect the individual address formats for Bitcoin, Litecoin, Ethereum, Monero, Qiwi, Yandex Money, and WebMoney in either USD or Rubles.

The exploit chain is long and involved, but it works: The victim-to-be first receives an eMail supposedly containing a scan of a lost passport in an attached PDF file.

When the PDF is opened (please don't!) the PDF opens an RTF (rich text format) file that contains an embedded HTA object that attempts to exploit a known (CVE-2017-8579) DirectX vulnerability which then executes a series of PowerShell commands to download and execute a password protected self-extracting SFX file... which installs "ComboJack."

Miscellany

Netflix does "Lost in Space"

The year was 1965 and I was the perfect age of 10 years old.

- "Danger! Danger! Will Robinson!!"
- "Does Not Compute!!"

Friday the 13th -- April 13, 2018!!

3 Minute Trailer: https://youtu.be/fzmM0AB60QQ

SpinRite

Rick James
Location: Toronto, ON, Canada
Subject: Security Now ASA vulnerability follow-up
Date: 17 Feb 2018 10:10:52

As I'm sure you and your listeners are aware, Cisco requires its customers to have a support contract to get ASA/VPN software updates. I wonder if this is a circumstance where Cisco NEEDS to release a patch like Microsoft did last year to Windows XP. Are software/hardware vendors only responsible to fix their security issues if you're a current customer? I thought this might be an interesting discussion....since we can't talk to Cisco about it! lol

PS: I had a 2TB drive "die" in my Drobo Mini last week, and guess what, I ran a level 4 on it and viola, all is good! SpinRite saved the drive.

Love the show, and your efforts (cough, bootable Mac SpinRite, cough) :)

Closing The Loop

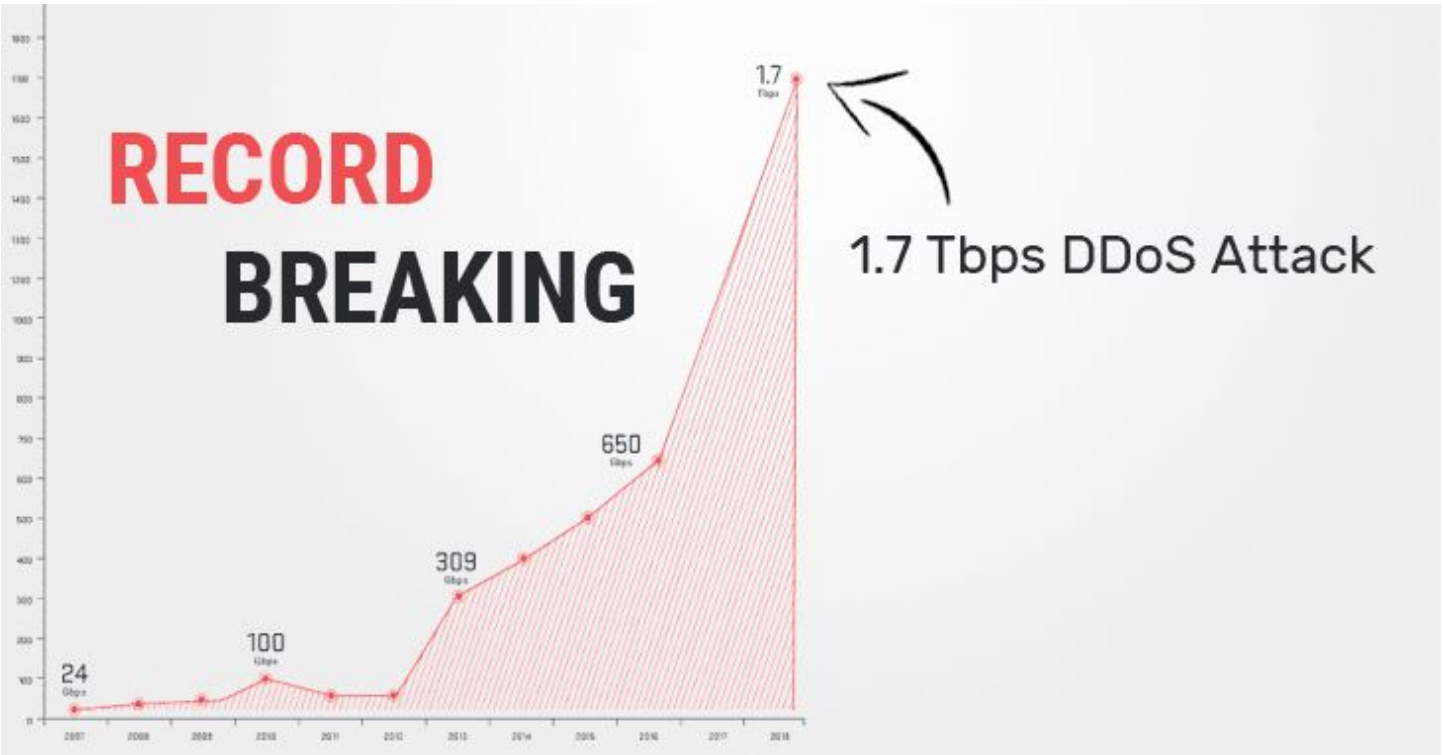
Joel Odom @joelodom
@SGgrc The bill in Georgia that you mentioned on Security Now is effectively dead, thanks to the hard work of EFF and the security community such as those of us at Georgia Tech. Thanks for bringing it to my attention via the podcast.
(Joel is a Georgia Tech computer security researcher.)

Ned Griffin @Ned_Griffin
Hey @SGgrc enabling @Quad9DNS in my router, but the few discussions I've looked are pointing people to changing in their computer setting. I've have control over all PC's on my network, so am I right assume the change at the router is OK?

Jared Komoroski @JaredKomoroski
@SGgrc / Your description of how Bluehost could securely use the last 4 of a password to verify account ownership is exactly correct. They store 2 hashes at password creation. I confirmed with some Bluehost Quality Assurance Engineers that I know. Keep up the good work!

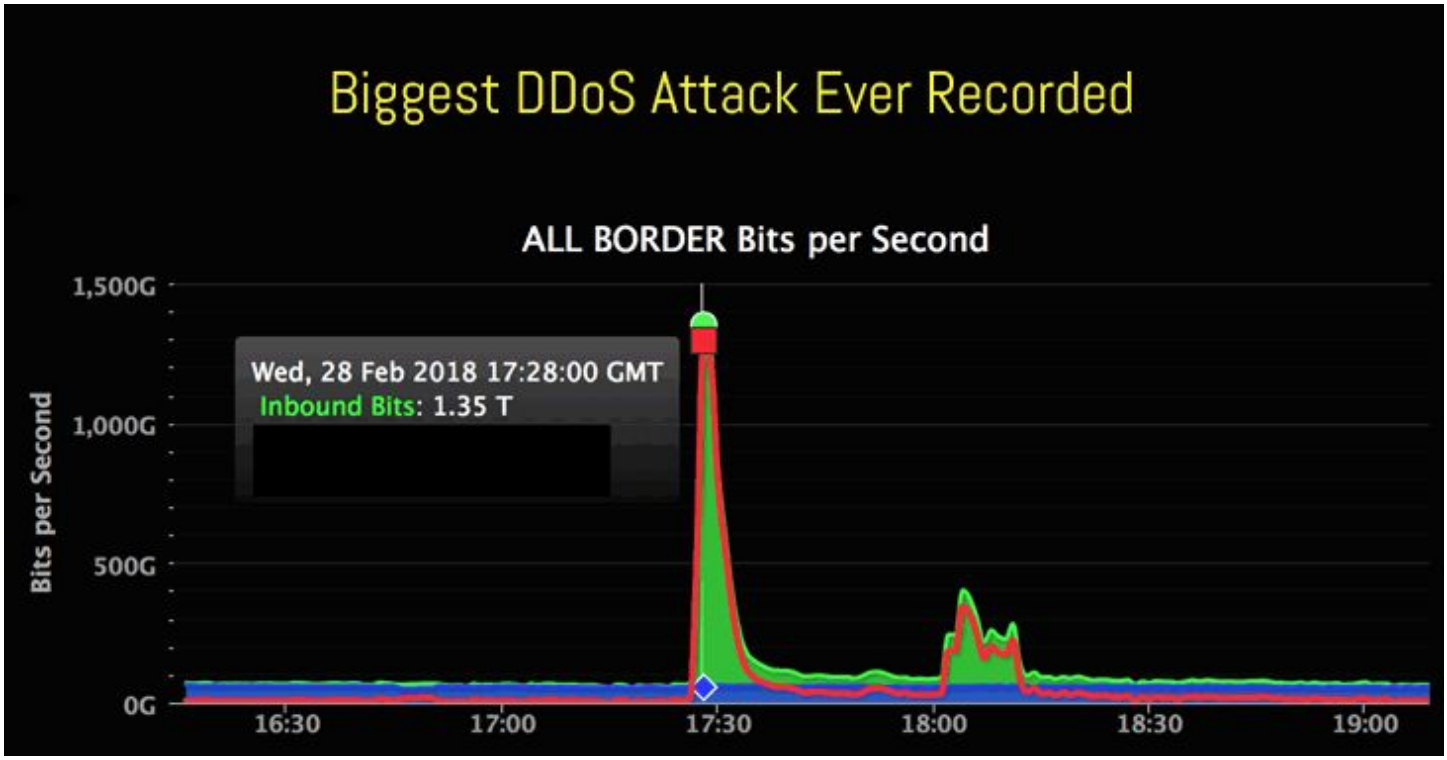
Martin Badke, CJL @lauxmyth
@SGgrc Speaking long ago with a German interpreter, she explained in German “When two vowels go walking, the second does the talking” as a teaching phrase. Diebold is said with the ‘e’. Consider Einstein and Siemens.

MidwestGuru @midwestguru
@SGgrc @leolaporte Diebold is a Germanic name. When e and i are together, pronounce the second one. Think Einstein v Dietrich.



“MemCrashed” DDoS Attacks

First there was the 1.35 TERABIT attack on Github…



In a post on its engineering blog, Github said, "The attack originated from over a thousand different autonomous systems (ASNs) across tens of thousands of unique endpoints. It was an amplification attack using the memcached-based approach that peaked at 1.35Tbps via 126.9 million packets per second."

Then, yesterday, Arbor Networks announced that record had been broken by a 1.7 TB attack!

"Memcached" ('d' is for daemon or server) - pronunciation: mem-cash-dee, mem-cashed

https://memcached.org/

What is Memcached?

Free & open source, high-performance, distributed memory object caching system, generic in nature, but intended for use in speeding up dynamic web applications by alleviating database load. Memcached is an in-memory key-value store for small chunks of arbitrary data (strings, objects) from results of database calls, API calls, or page rendering.

Memcached is simple yet powerful. Its simple design promotes quick deployment, ease of development, and solves many problems facing large data caches. Its API is available for most popular languages.

Memcached was first developed by Brad Fitzpatrick for his website LiveJournal, on May 22, 2003. It was originally written in Perl, then later rewritten in C and employed by LiveJournal.

Memcached is now used by many other systems, including YouTube, Reddit, Facebook, Twitter, and Wikipedia. Google App Engine, Microsoft Azure, IBM Bluemix and Amazon Web Services also offer a Memcached service through an API.

TCP or UDP listening on well-known port 11211

UDP can be spoofed.

Previous reflection / amplification attacks:

Meant for greased lightning speed, so NO AUTHENTICATION.

It's meant ONLY FOR INTERNAL USE and the MemCached server should normally be tightly bound to the localhost machine for internal-only access. But there are cluster application where the server might be unbound.

But NEVER public.
Shodan reports 87,811 open memcached servers!!