RSAConference2015

San Francisco | April 20-24 | Moscone Center

CHANGE

Challenge today's security thinking

SESSION ID: HTA-F02

# Are you giving firmware attackers a free pass?

**Xeno Kovah**

CEO & Co-Founder
LegbaCore, LLC
@XenoKovah

**Corey Kallenberg**

CTO & Co-Founder
LegbaCore, LLC
@CoreyKal

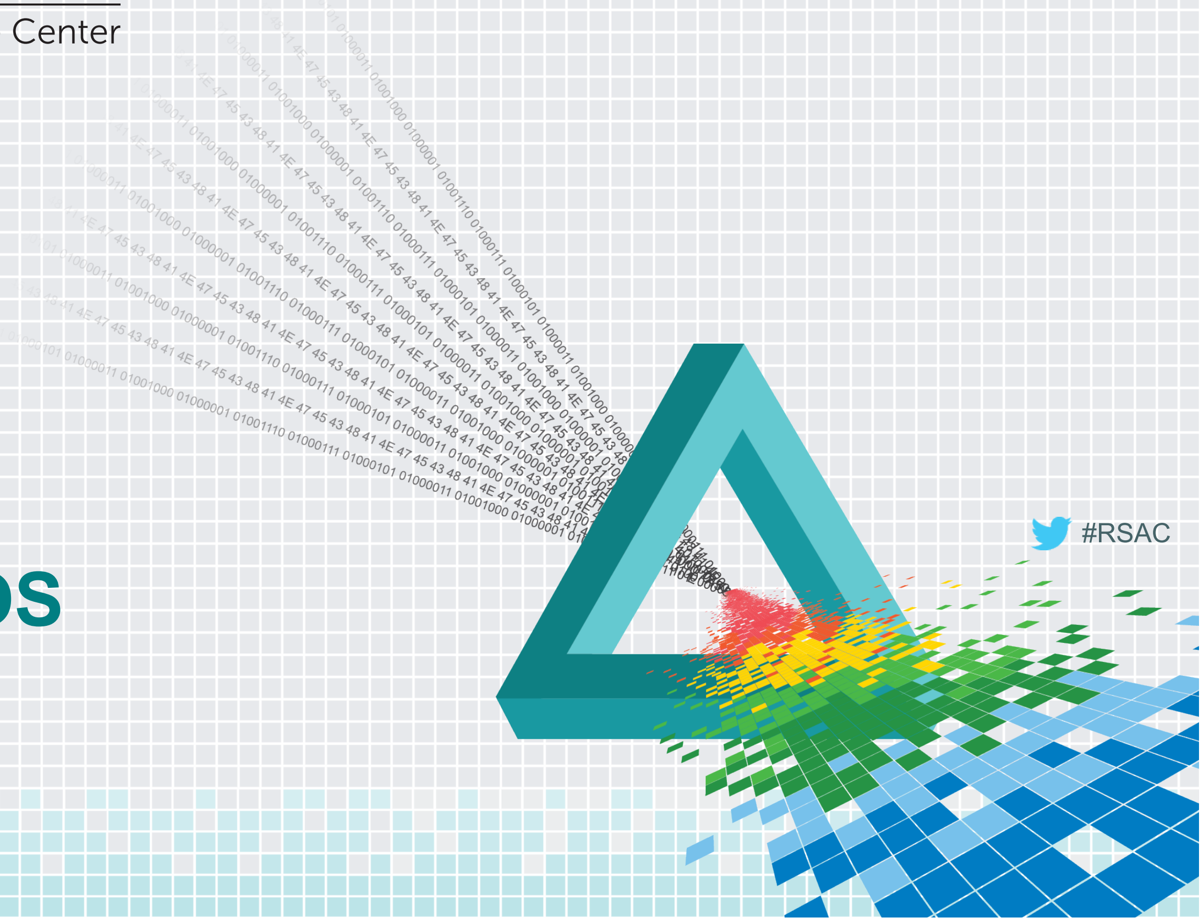
#RSAC

# YES

RSAConference2015

San Francisco | April 20-24 | Moscone Center

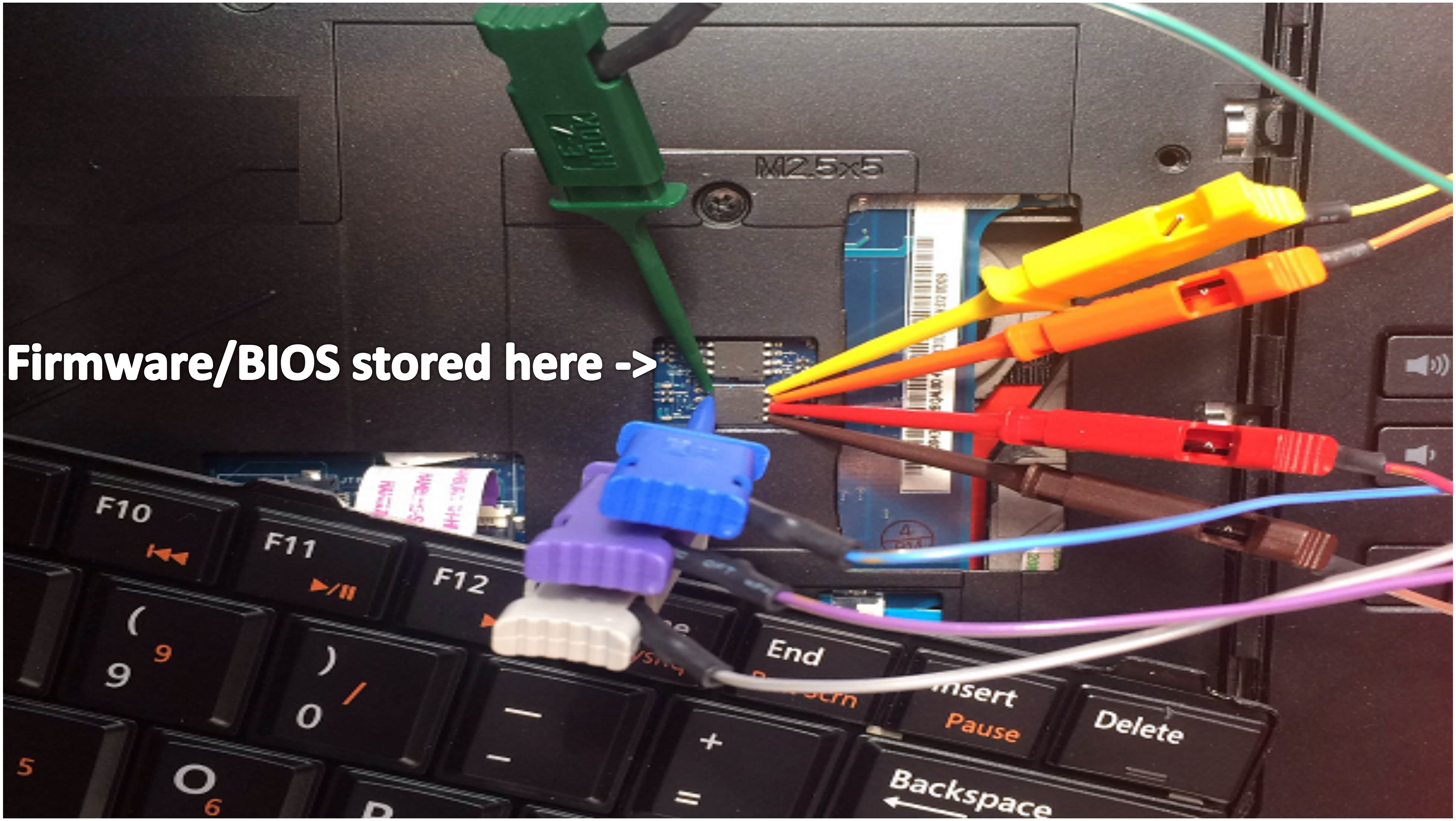
# Better know a BIOS



#RSAC

# What do we mean when we say…

- **Firmware** is the first *software* run by a system
  - It is not hardware, though it's job is usually to configure hardware
  - It is only called "firm" because it is typically stored in a non-volatile flash chip, soldered to a circuit board somewhere

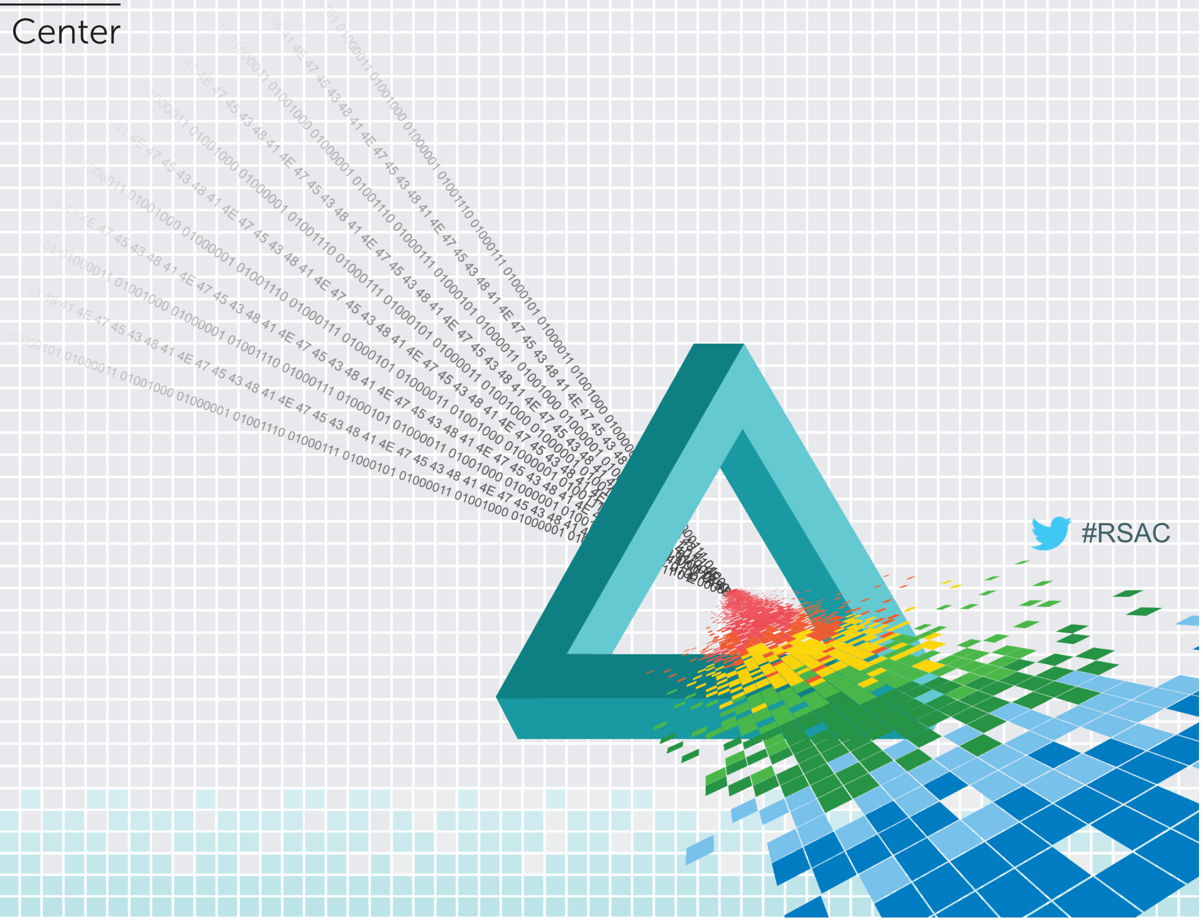

Firmware/BIOS stored here ->

# What do we mean when we say…

- **Firmware** is the first *software* run by a system
  - It is not hardware, though it's job is usually to configure hardware
  - It is only called "firm" because it is typically stored in a non-volatile flash chip, soldered to a circuit board somewhere
- Since the first IBM x86 PCs, an Intel x86 CPU's firmware has been referred to as the **BIOS** (Basic Input/Output System)
- The new industry standard for BIOS is to comply with the Unified Extensible Firmware Interface (**UEFI**) specification
  - An open source UEFI reference implementation is publicly available
- System Management Mode (**SMM**) is the most privileged CPU execution mode on an x86 system

RSAConference2015

San Francisco | April 20-24 | Moscone Center

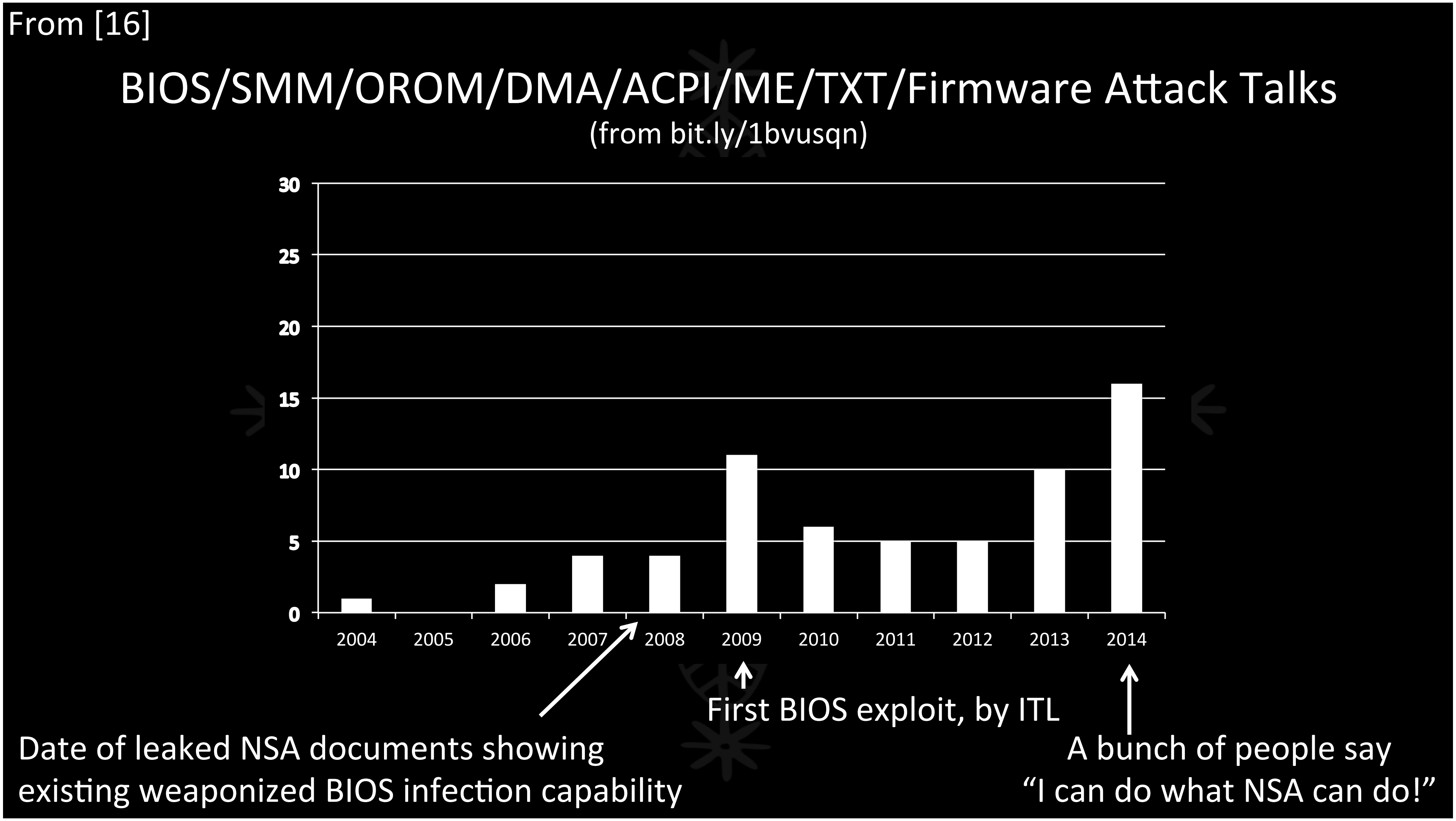


# The recent past

#RSAC

# Triumph & Tragedy

- Over the last 2 years we have researched, found, and responsibly disclosed numerous vulnerabilities that would defeat SecureBoot or allow infection of the BIOS or SMM
  - CERT VU#s 912156[1](“Ruy Lopez”), 255726[1](“The Sicilian”), 758382[2] (“Setup bug”), 291102[4] (“Charizard”), 552286[5](“King & Queen’s Gambit”), 533140[6] (“noname”), 766164[7] (“Speed Racer”), 976132[8] (“Venamis”), 577140[9](“Snorlax”)
- Other groups like the Intel Advanced Threat Research team, and Invisible Things Lab before them have also found and disclosed many vulnerabilities



From [16]

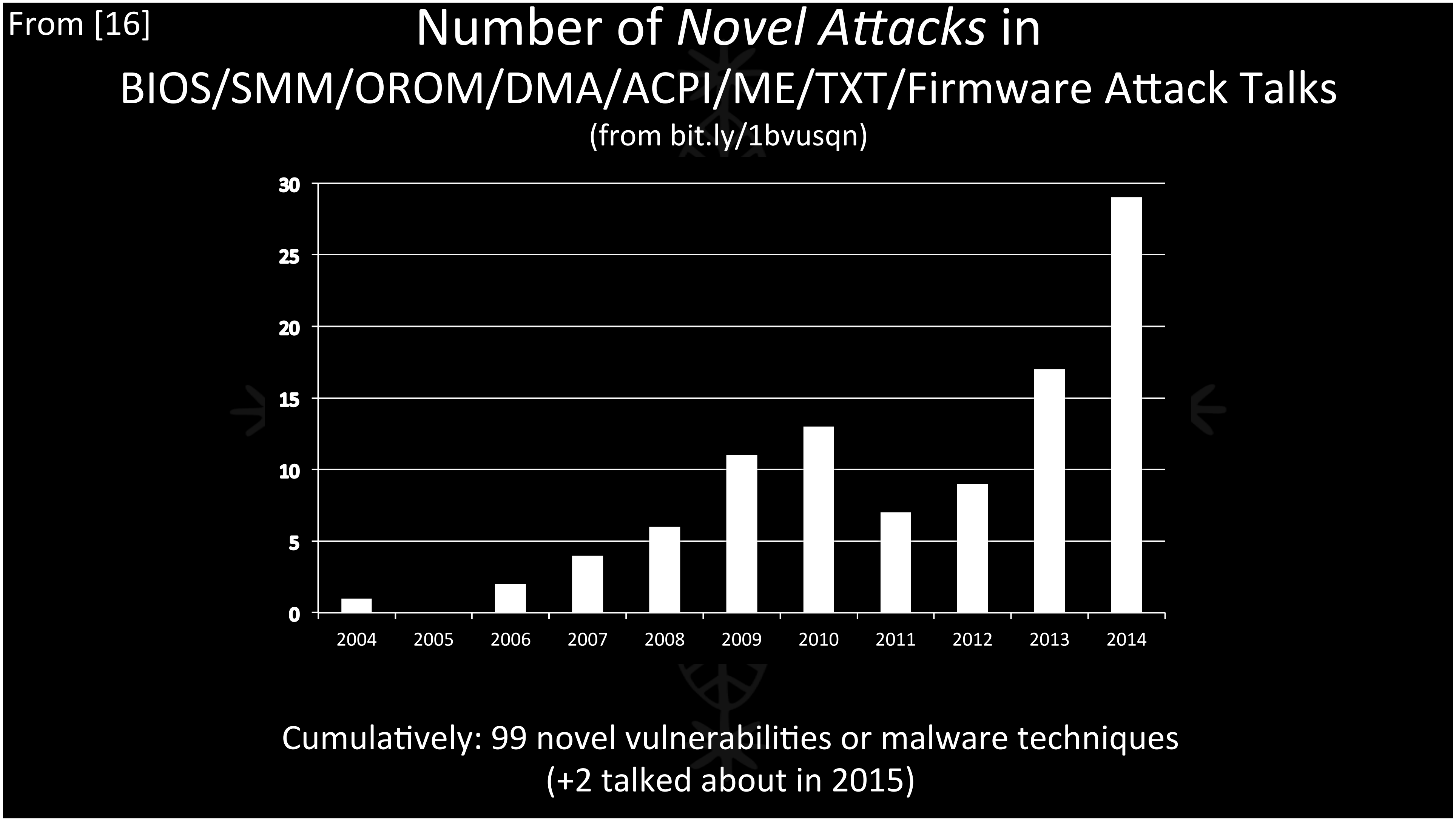
BIOS/SMM/OROM/DMA/ACPI/ME/TXT/Firmware Attack Talks

(from bit.ly/1bvusqn)

First BIOS exploit, by ITL

Date of leaked NSA documents showing existing weaponized BIOS infection capability

A bunch of people say “I can do what NSA can do!”



From [16]

# Number of *Novel Attacks* in BIOS/SMM/OROM/DMA/ACPI/ME/TXT/Firmware Attack Talks

(from bit.ly/1bvusqn)

Cumulatively: 99 novel vulnerabilities or malware techniques
(+2 talked about in 2015)

# Triumph & Tragedy

- The top OEMs issued patches for most vulnerabilities
  - Many smaller OEMs *never released patches!*
- Even the top OEMs will often only issue patches the last N models
  - We’re trying to get them to make N public

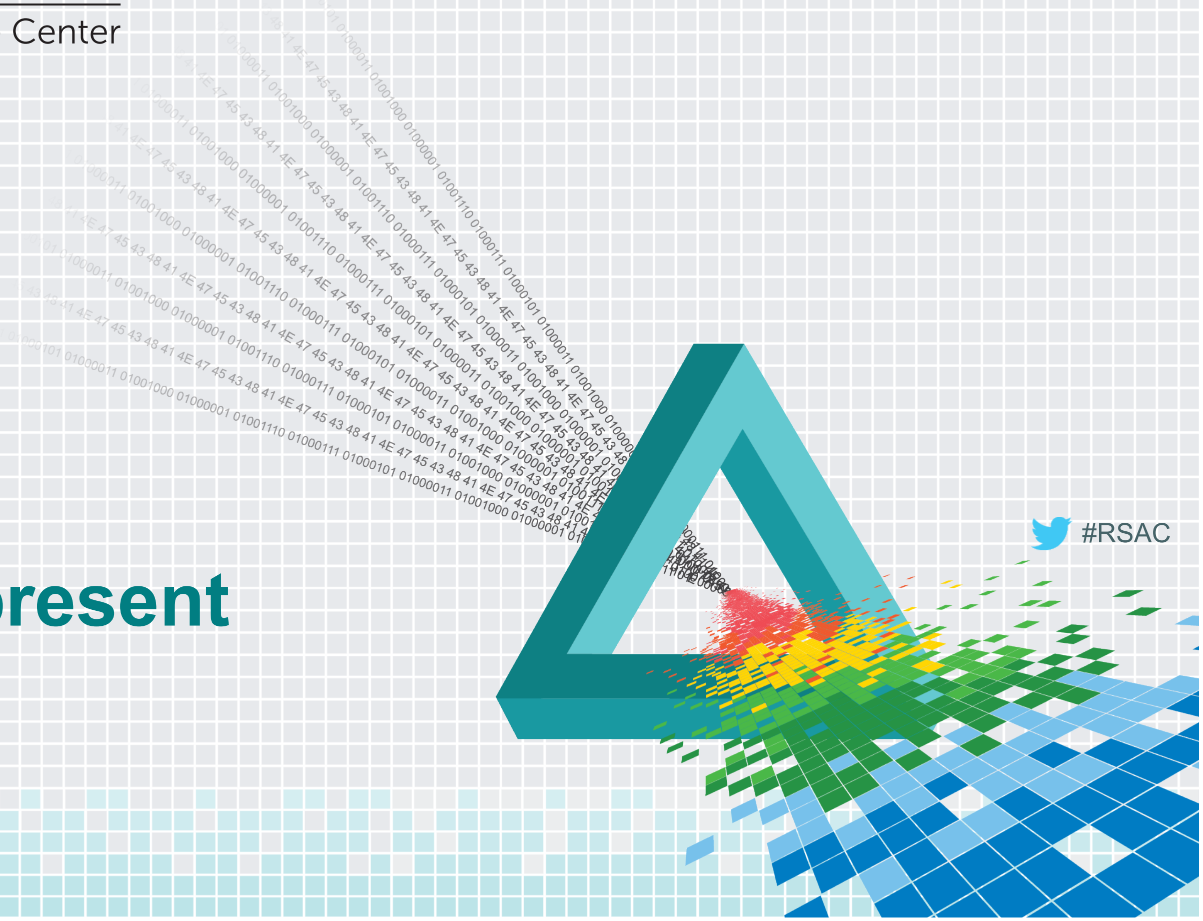
# Triumph & Tragedy

- From our conversations with companies and individuals, there has been no significant uptick in BIOS patch management becoming part of corporate best practices
- We did the right thing, and were counting on companies to do the same, but it never happened
- This talk will hopefully convince you why this is important

RSAConference2015

San Francisco | April 20-24 | Moscone Center

# The unfortunate present



#RSAC

# Threats

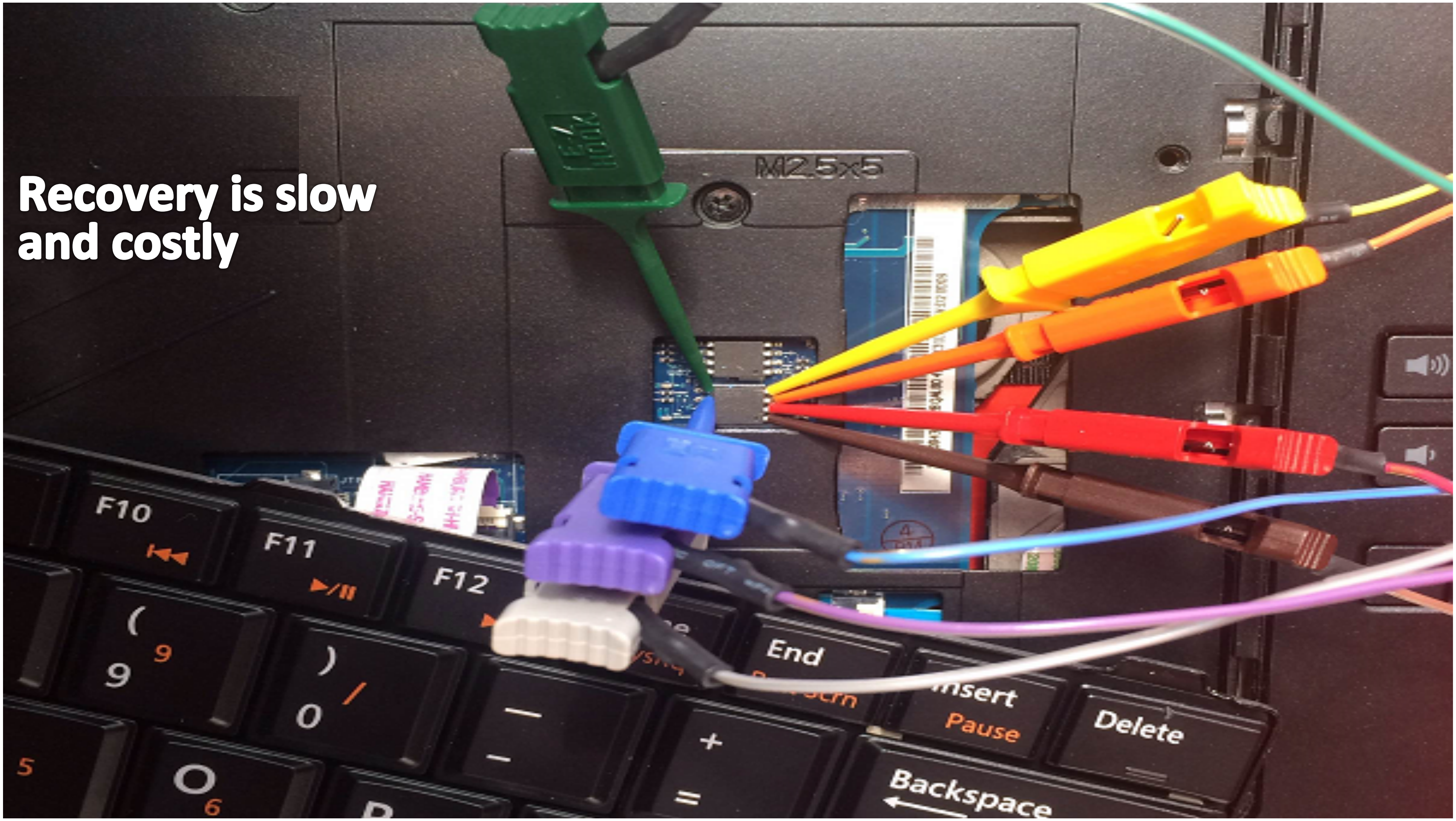
- In Sept. 2011 the first crimeware using BIOS infection (Mebromi) was found [10]
- In Dec 2013 NSA defensive director said other states are developing BIOS attack capabilities [11]
- In Dec 2013 Snowden leaks said NSA offensive has a catalog of offensive capabilities that includes BIOS/SMM implants [12]
- In Jan 2014 CrowdStrike said that some malware they attributed to Russia is collecting BIOS version info (but they didn't say they had seen BIOS infection itself) [13]

# The world post-Snowden

- Every country in the world now knows that firmware attacks are *unequivocally* the way to reliably persist on target networks, unseen, for years at a time
- All the world's intelligence agencies are saying: "Me too! Me too!"

# Destructive attacks can get worse

- Some nation state actors have shown the will to exercise destructive HD-wiping attacks
  - Iran against Saudi Aramco
  - North Korea against South Korean banks / Sony
- Firmware-wiping attacks are just as possible, but far more devastating, and far more difficult to recover from



Recovery is slow and costly

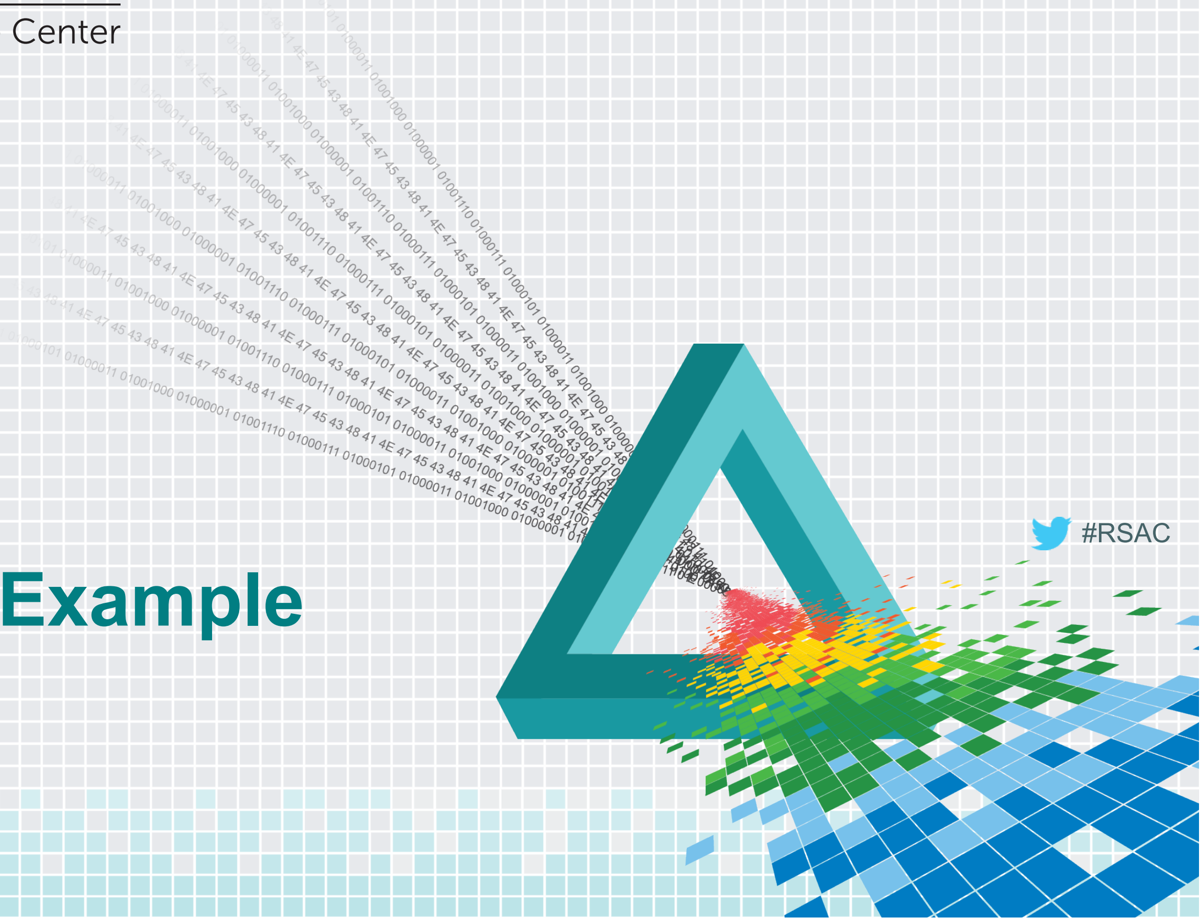
# We hold these truths to be non-obvious

- Because almost no one applies BIOS patches, almost every BIOS in the wild is affected by *at least* one vulnerability, and can be infected
- The high amount of code reuse across UEFI BIOSes means that BIOS infection is automatable and reliable (see [9] for details)

# 3 paths to infection

- Remote interaction
- Physical interaction
- Supply chain

# Remote Infection Example



# LightEater

Hello my friends.
Welcome to my home
in the Deep Dark

# What can a LightEater do?

- LightEater lives in SMM
- SMM is the most privileged CPU execution mode
- Therefore LightEater trumps all security systems
- And LightEater can perform *any attack* that a lesser-privileged (e.g. hypervisor, kernel, userspace) attacker can perform

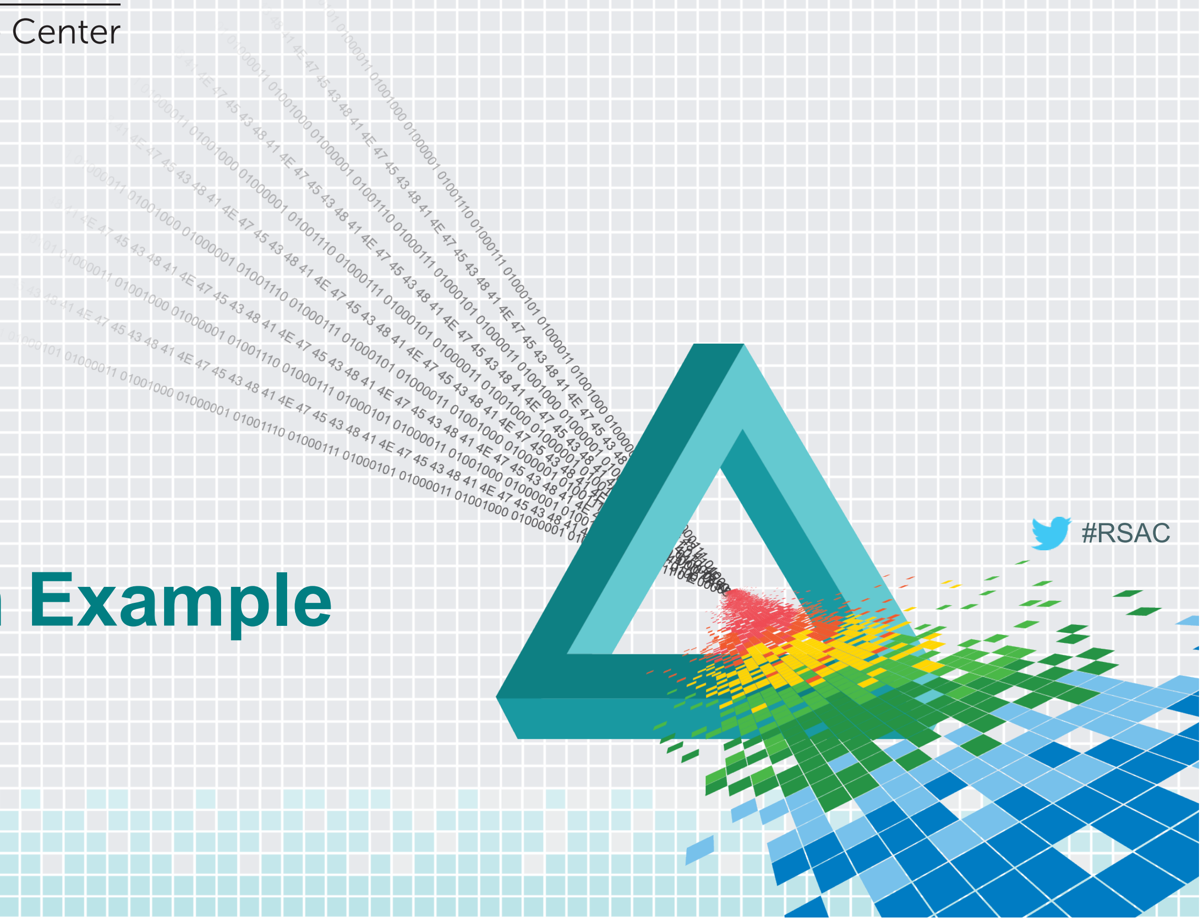
# LightEater on ASUS

- We chose to show a typical kernel-mode rootkit behavior
  - But instigated from infected SMM
- LightEater will hook into the OS internals to be notified every time a new process starts
  - It can then choose to hack that process or not

# Demo

RSAConference2015

San Francisco | April 20-24 | Moscone Center



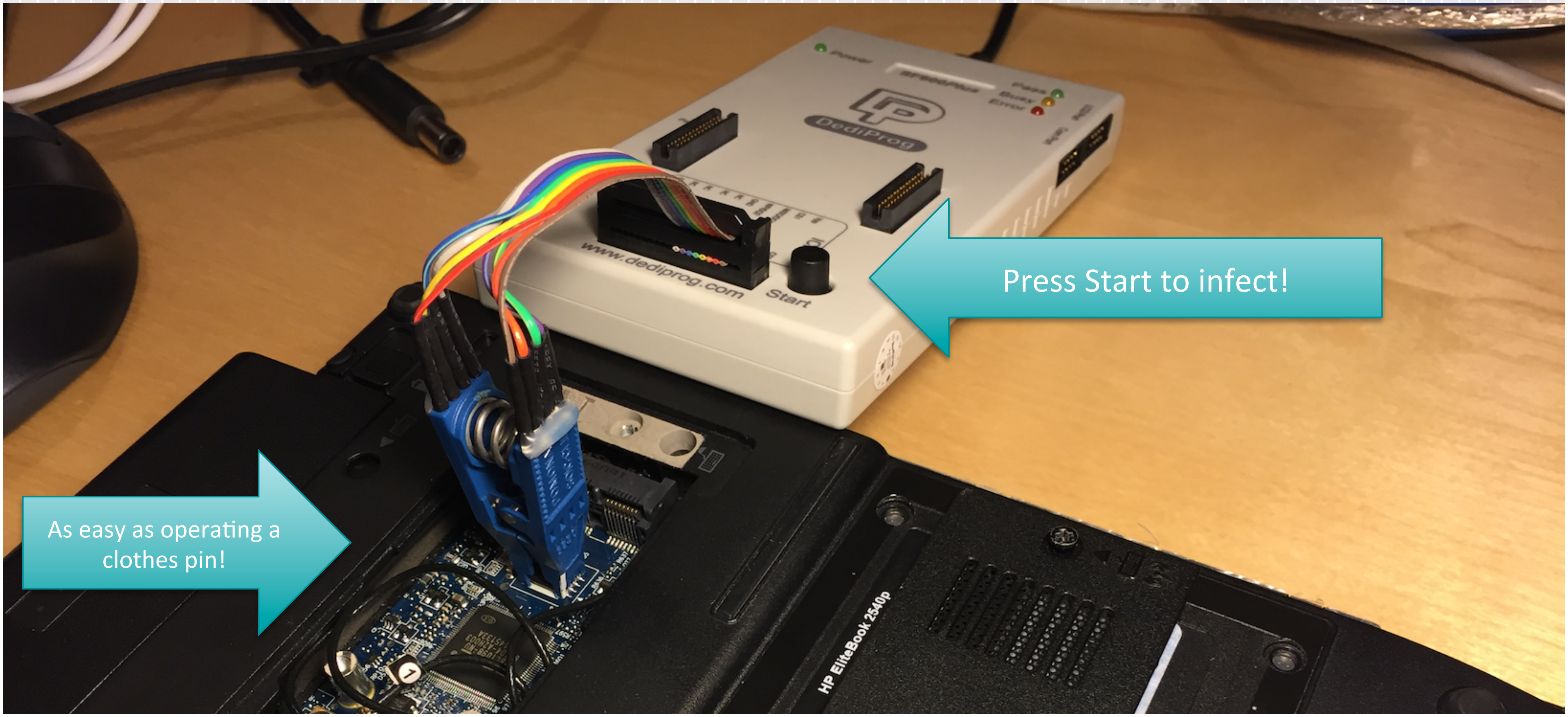
#RSAC

# Physical Infection Example

# Possible touchpoints

- “Evil Maid” attacks when you leave your laptop in your hotel room, or when your cleaners come into the office for the night
- “Border Guard” attacks when you’re crossing international borders

# It’s easier for an unskilled accomplice than you think: unscrew 1 screw, clip, press “start”, wait 50 seconds, done



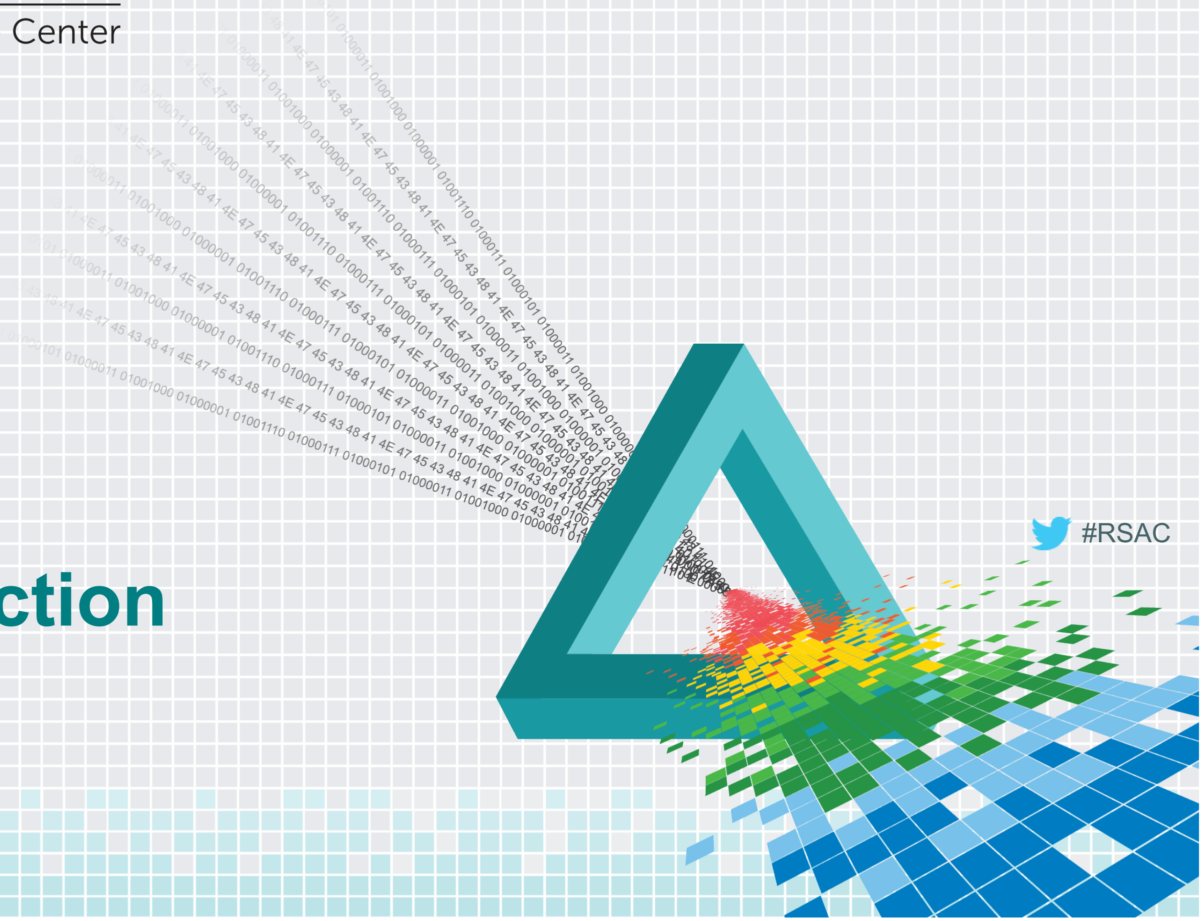
# LightEater on HP

- In this case LightEater will exfiltrate data over the network using Intel Serial-Over-Lan
  - A legitimate capability found in many enterprise-grade systems
  - No low level driver needed. Write data to a port, packets come out
- Has an option to “encrypt” data with bitwise rot13 to thwart network defenders ;)

# Demo

RSAConference2015

San Francisco | April 20-24 | Moscone Center



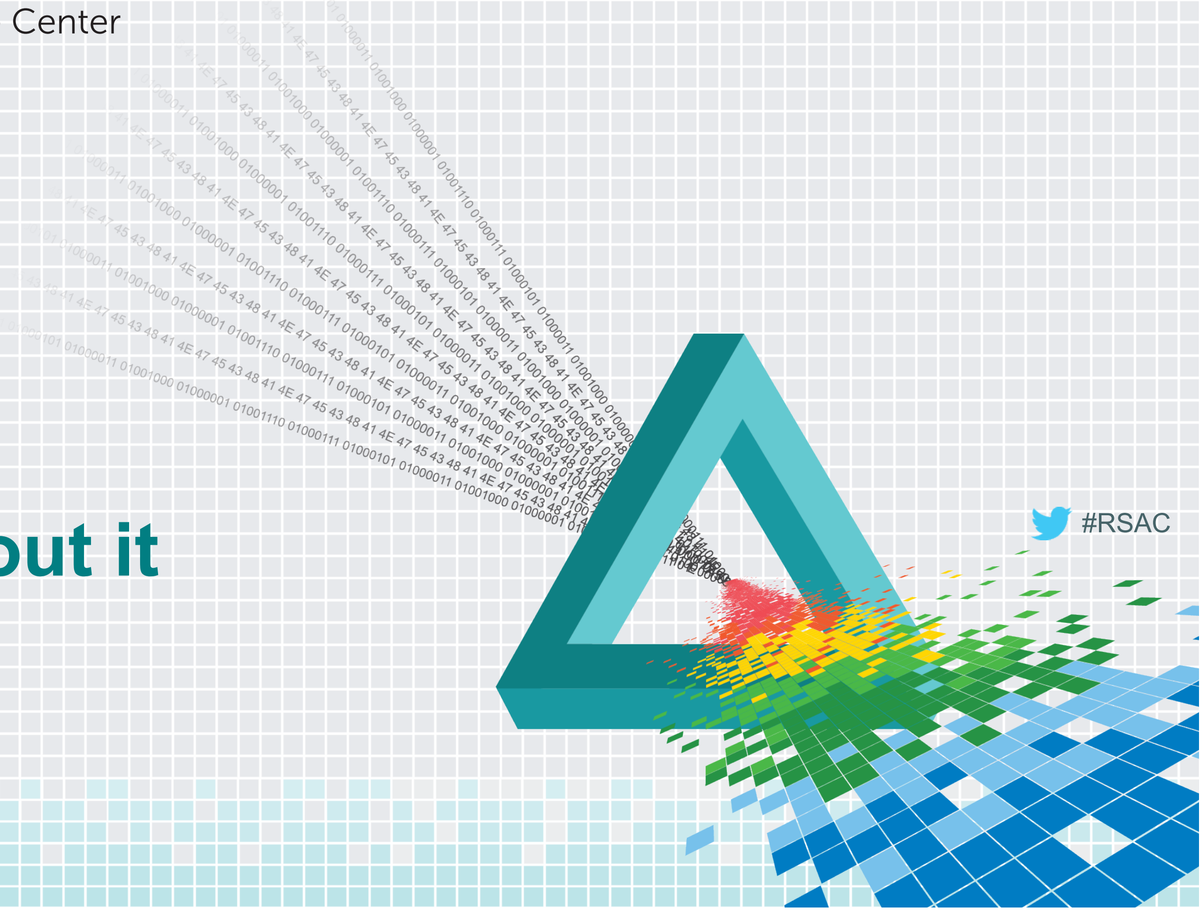
#RSAC

# Supply chain infection

RSAConference2015

San Francisco | April 20-24 | Moscone Center

## Do something about it *TODAY*



#RSAC

# BIOS problems are detectable, if you only look!

- 2 kinds of problems we want to look for:
- Vulnerabilities
  - “Can this system be hacked?”
- Infections
  - “Has this system been hacked?”

# Can this system be hacked?

- Copernicus [14]
  - Xeno ran this project at previous employer
  - Designed for enterprise deployment
    - Run on ~10k systems in production environments
  - Supports Intel CPUs on Windows >= 7 32/64bit
  - Previously freely distributed as signed binary
    - After we left, they added a requirement to fill out a “FastLicense request” form to get a copy of the binary

# Can this system be hacked?

- Intel ChipSec – http://github.com/chipsec/chipsec
  - Designed for modularity – excellent for security researchers
  - Meant to run on single test systems which are representative of a broader population
    - Very prominent warning.txt says not to run on production systems
  - Supports Windows/Linux/UEFI Shell
  - Distributed as source, it requires you to sign it yourself to run on Windows (usually use a self-signed key on non-production system)

# Example vulnerability assessment scenarios

- Representative sample audit
  - Collect one of each model that is in your corporate lifecycle program
  - Update BIOS on representative systems to latest
  - Run ChipSec on each model
  - If it shows no vulnerabilities, then you should update all Models in your environment to that version
  - If it shows vulnerabilities, then you should contact the vendor and contact us so we can help work with the vendor to fix the vulnerabilities

# Example vulnerability assessment scenarios

- Full enterprise audit
  - Push Copernicus kernel driver and a script to run it to all endpoints, using your patch management system
  - Collect Copernicus output to central server
  - Use Copernicus' protections.py with the "per-version" option to create a summary document that shows which Vendor/Model/Revision BIOSes in your environment are currently vulnerable

  - This has been done on ~10k production systems

# BIOS problems are detectable, if you only look!

- 2 kinds of problems we want to look for:
- Vulnerabilities
  - “Can this system be hacked?”
- Infections
  - “Has this system been hacked?”

# Has this system been hacked?

- Use Copernicus
  - Copernicus, ChipSec, and Flashrom can dump the contents of the flash chip which contains the BIOS
  - But only Copernicus includes an integrity check mechanism
  - bios_diff.py compares two UEFI BIOSes' firmware filesystem and prints any differences

# Example integrity checking scenarios

- Evil Maid scenario
  - Dump the BIOS before a system travels abroad
  - Dump the BIOS after, and diff against the before
- Enterprise audit - acceptable scenario
  - Bucket all your BIOSes according to Vendor/Model/Revision
  - Treat one BIOS as golden, and diff all others against it
- Enterprise audit - best case scenario
  - Extract a known clean BIOS image from a BIOS update that the vendor provides on their website
  - Diff all matching Vendor/Model/Revision BIOSes against that gold copy

# BIOS integrity check failures

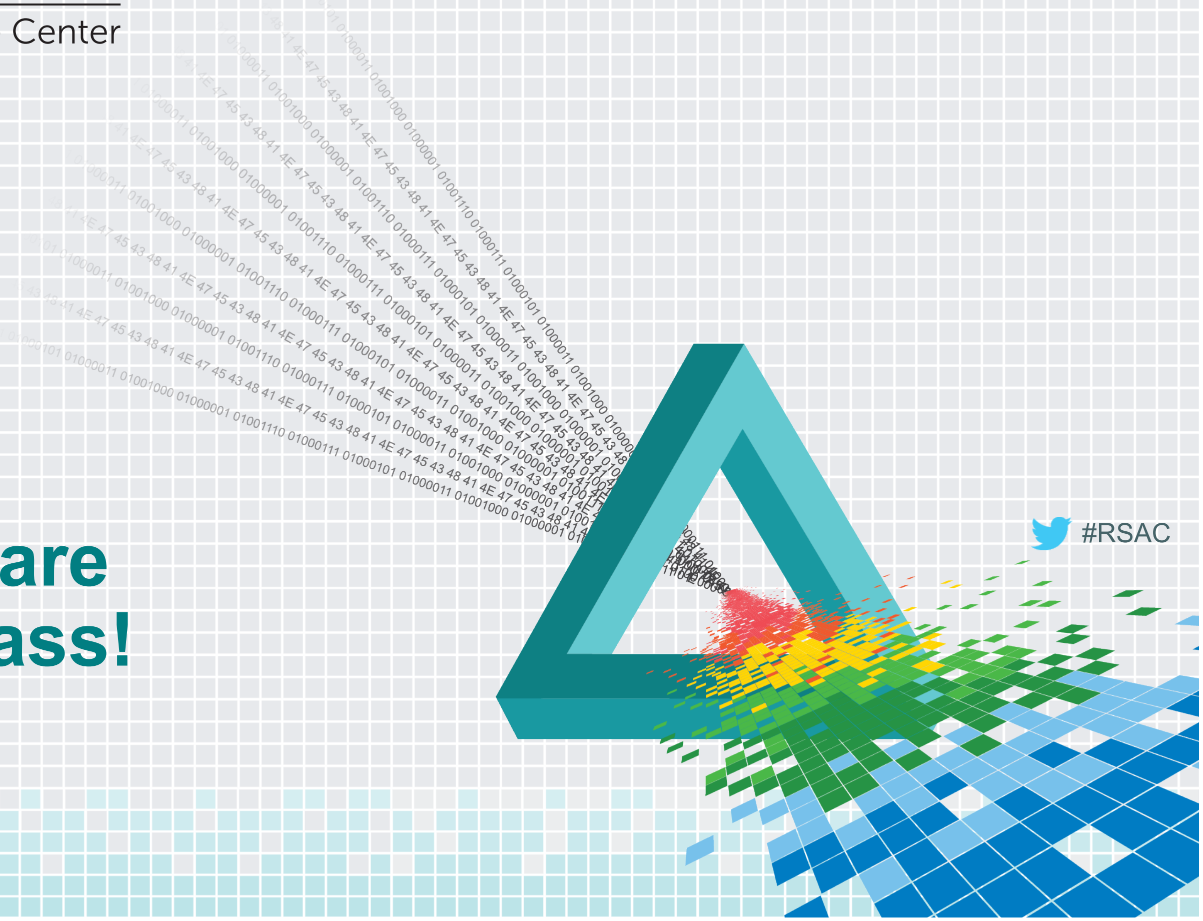
- If an integrity failure is found, you have a few options to determine if it is a genuine malware detection, or a tool problem

  1. Insource the analysis by sending your malware analysts/forensics experts to our BIOS security training
  2. Ask your friendly neighborhood intelligence agency
  3. Ask the OEM
  4. Ask us :)

RSAConference2015

San Francisco | April 20-24 | Moscone Center

# Stop giving firmware attackers a free pass!



#RSAC

# Apply – NEXT WEEK

- Find out if your asset management software collects information about hardware models' BIOS revisions.
  - If not, tell your vendor you want that capability
  - If so, build a histogram of your most common hardware models for prioritization
- Have IT patch the BIOS and run ChipSec or Copernicus on the small collection of "representative machines"
  - If no vulnerabilities, prepare BIOS patch management procedures
  - If vulnerabilities, let us know so we can talk to the OEM

# Apply – 3 MONTHS

- Patch the BIOS for at least the single model of PC that is most common in your environment
- Push Copernicus/Flashrom through your patch management system to collect vulnerability & integrity information for all your systems
- Institute a loaner-laptop policy for traveling employees & perform integrity checks on the firmware with Copernicus upon return

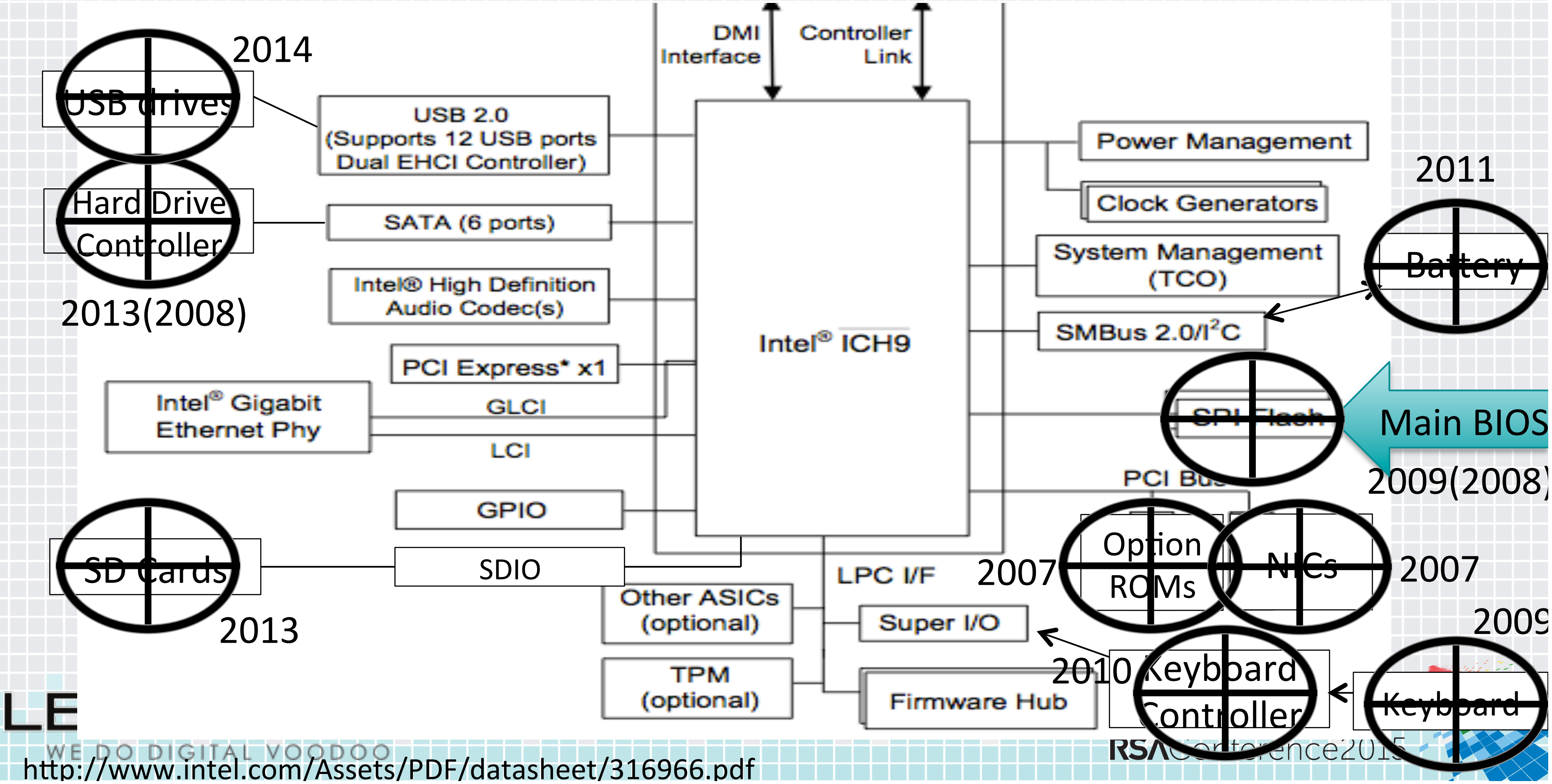
# Apply – 12 MONTHS

- Be collecting BIOS version information incorporated into your asset management product of choice
- Make BIOS patch management for all models in your environment part of your standard procedures
- Analyze vulnerability/integrity data returned by Copernicus/Flashrom
- Utilize our services to do a more trustworthy audit on systems you think are potential high value/mission critical targets
- Provision your Trusted Platform Modules (TPMs) to enable more trustworthy assessment technologies (sorry, Macs are out of luck)
- Ask your OEM if they utilize "Dual Monitor Mode" to stop SMM from being able to completely compromise the system

# Then you're done with firmware, right?

- Today we've only talked about BIOS
- There are many other firmware blobs in your x86 system that have been the target of attack research…

# Other publicly targeted firmware[0]:



http://www.intel.com/Assets/PDF/datasheet/316966.pdf

# Conclusion

- Stop giving firmware attackers a free pass! Start patching!
- Checking UEFI BIOS for vulnerabilities and infections is no longer a research problem. It’s something you can start doing TODAY!

# Questions?

- Contact: {xeno,corey}@legbacore.com
- http://legbacore.com/Contact.html for our GPG keys
- http://legbacore.com/Research.html for the latest slides

- Go check out OpenSecurityTraining.info for the free classes from Xeno and Corey on x86 assembly & architecture, binary executable formats, stealth malware, and exploits. As well as lots of good classes from others

# References

[0] Low level PC attack Papers Timeline by Xeno Kovah http://timeglider.com/timeline/5ca2daa6078caaf4

[1] Defeating Signed BIOS Enforcement – Kallenberg et al., Sept. 2013
http://conference.hitb.org/hitbsecconf2013kul/materials/D1T1%20-%20Kallenberg,%20Kovah,%20Butterworth%20-%20Defeating%20Signed%20BIOS%20Enforcement.pdf

http://www.kb.cert.org/vuls/id/912156

http://www.kb.cert.org/vuls/id/255726 (CERT hasn't posted yet despite request)

[2] All Your Boot Are Belong To Us (MITRE portion) – Kallenberg et al. – Mar. 2014, delayed from publicly disclosing potential for bricking until HITB at Intel's request https://cansecwest.com/slides/2014/AllYourBoot_csw14-mitre-final.pdf

http://www.kb.cert.org/vuls/id/758382

[3] All Your Boot Are Belong To Us (Intel portion) – Bulygin et al. – Mar. 2014 https://cansecwest.com/slides/2014/AllYourBoot_csw14-intel-final.pdf

[4] Setup for Failure: Defeating UEFI Secure Boot - Kallenberg et al., Apr. 2014
http://syscan.org/index.php/download/get/6e597f6067493dd581eed737146f3afb/SyScan2014_CoreyKallenberg_SetupforFailureDefeatingSecureBoot.zip

http://www.kb.cert.org/vuls/id/291102 (CERT hasn't posted yet despite request)

# References

[5] Extreme Privilege Escalation on UEFI Windows 8 Systems – Kallenberg et al., Aug. 2014
https://www.blackhat.com/docs/us-14/materials/us-14-Kallenberg-Extreme-Privilege-Escalation-On-Windows8-UEFI-Systems.pdf

http://www.kb.cert.org/vuls/id/766164

[6] Attacks against UEFI Inspired by Darth Venamis and Speed Racer – Wojtczuk & Kallenberg, Dec. 2013
https://bromiumlabs.files.wordpress.com/2015/01/attacksonuefi_slides.pdf http://www.kb.cert.org/vuls/id/533140

[7] Speed Racer: Exploiting an Intel Flash Protection Race Condition – Kallenberg & Wojtczuk, Dec. 2013
https://frab.cccv.de/system/attachments/2565/original/speed_racer_whitepaper.pdf

http://www.kb.cert.org/vuls/id/912156

[8] Attacking UEFI Boot Script – Wojtczuk & Kallenberg, Dec. 2013

https://frab.cccv.de/system/attachments/2566/original/venamis_whitepaper.pdf

http://www.kb.cert.org/vuls/id/552286

[9] “Snorlax” bug – Cornwell, et al., Dec. 2013

https://frab.cccv.de/system/attachments/2566/original/venamis_whitepaper.pdf

http://www.kb.cert.org/vuls/id/577140 (CERT hasn’t posted yet despite request)

# References

[10] “Mebromi: the first BIOS rootkit in the wild”

http://www.webroot.com/blog/2011/09/13/mebromi-the-first-bios-rootkit-in-the-wild/

[11] “NSA Speaks Out on Snowden Spying”, Dec 2012

http://www.cbsnews.com/news/nsa-speaks-out-on-snowden-spying/

[12] "To Protect And Infect” - Jacob Applebaum, Dec. 2012

https://www.youtube.com/watch?v=vILAlhwUgIU (contains leaked classified NSA documents)

[13] “U.S. Gas, Oil Companies Targeted in Espionage Campaigns”, Jan. 2013

http://threatpost.com/u-s-gas-oil-companies-targeted-in-espionage-campaigns/103777

[14] Copernicus: Question Your Assumptions about BIOS Security,  John Butterworth, Jul. 2013

https://www.mitre.org/capabilities/cybersecurity/overview/cybersecurity-blog/copernicus-question-your-assumptions-about

[15] Betting BIOS Bugs Won’t Bite Y’er Butt? – Kovah & Kallenberg, Jan. 2015

http://legbacore.com/Research_files/2015_ShmooCon_BIOSBugs.pdf