Betting BIOS Bugs Won't Bite Y'er Butt?

Xeno Kovah

Corey Kallenberg

About us

- MITRE researchers for the past 7.5 years
- As of *today*, full time at LegbaCore!
- Focused on low level x86 security at the kernel level and below (kernel, VMM, SMM, BIOS, peripheral firmware, etc)
- Papa Legba is the voodoo spirit who serves as an access control mechanism between the human world and spirit world. Or meatspace and cyberspace

About this talk

- Discuss the threats against BIOS
- Go through *a few* of the numerous vulnerabilities we and others have found over the past 2 years
- Explain how you can start to protect yourself

- BIOS and UEFI (Unified Extensible Firmware Interface) will be used as synonyms in this talk, because almost all modern BIOSes use (U)EFI

What, me worry? About BIOS security?

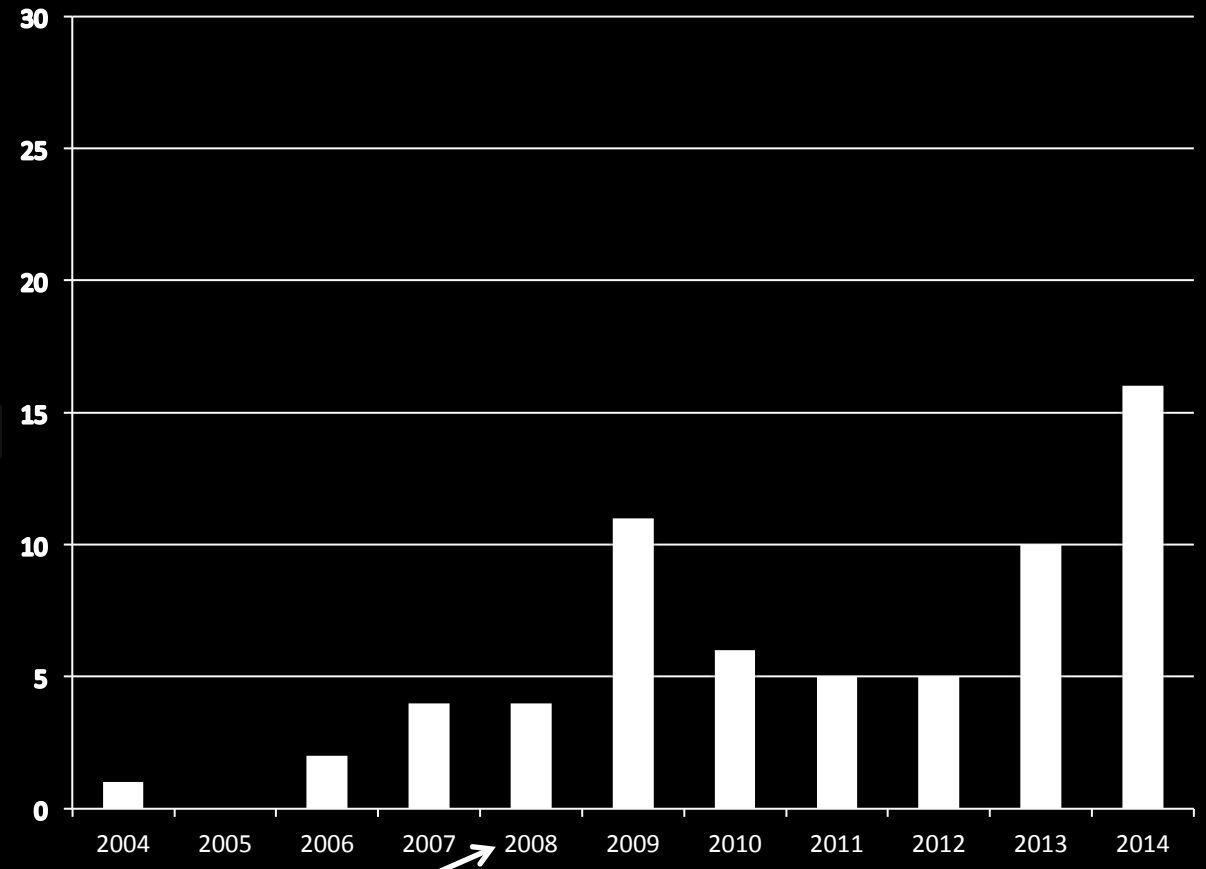
- If your BIOS is compromised everything is compromised
- If your BIOS is corrupted your system is bricked
- Attacker does not need physical presence to attack BIOS
- BIOS passwords do not protect BIOS itself, only some configuration settings
- No enterprise-grade machines use update jumpers

Threats

- In Sept. 2011 the first crimeware (Mebromi) was founding using BIOS infection [13]
- In Dec 2013 NSA IAD director said other states are developing BIOS attack capabilities [14]
- In Dec 2013 Snowden leaks said NSA SID has a catalog of offensive capabilities that includes BIOS/SMM implants [15]
- 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) [16]

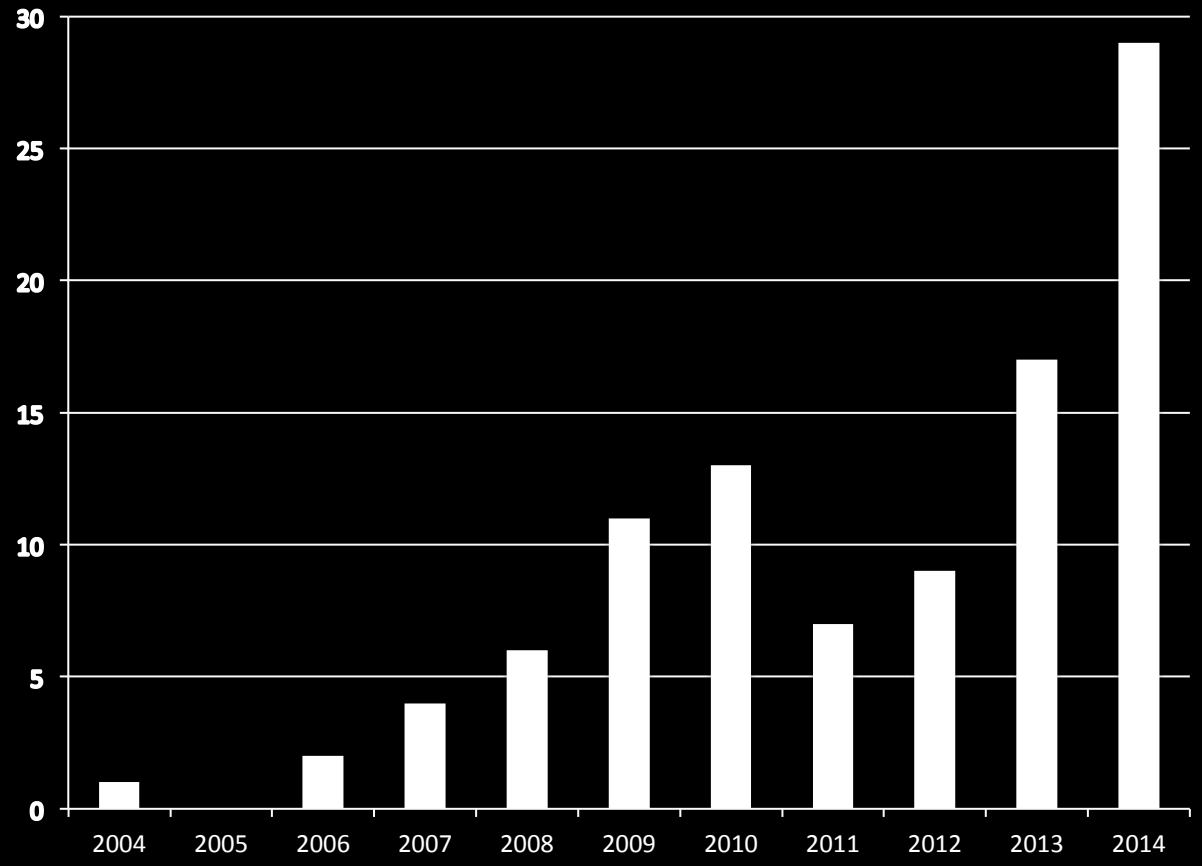
BIOS/SMM/OROM/DMA/ACPI/ME/TXT/Firmware Attack Talks

(from bit.ly/1bvusqn)



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

2015: Know of at least 4 vulns under disclosure not yet publicly talked about

Top ~~10~~ 5 attacks of 2013-2014

- See [17] for a dedicated survey paper of attacks
- Wanted to cover, but due to time constraints, they were moved to backup
  - CSW13 – Evil Maid Got Angrier [1]
  - NSC2013 – Tick & Flea [2]
  - BH2013 – Defeating Win 8 Secure Boot [3]
  - CSW14 – VU#758382 Setup UEFI variable [4]
  - CSW14 – All Your Boot are Belong to Us [5]
- Let's talk about these
  - BH2013 – VU#912156 (Ruy Lopez) [6]
  - 31C3 – Thunderstrike [7][8]
  - 31C3 – VU#766164 (Speed Racer) [9]
  - BH2014 – VU#552286 (King & Queen's Gambit) [10]
  - 31C3 – VU#976132 (Darth Plagueis) [9]

Compromise Key/Legend

X	√	X	√	X	√	X	X	X
SPI	Brick	SMM	SecBoot	TpmBoot	AuthVars	BLE	BWP	PRR

- √ = attack can defeat protection, X = attack can't defeat protection
- SPI = Able to write to the SPI flash chip, where the BIOS lives. Uber-win, grants all other capabilities
- Brick = able to render the system inoperable
- SMM = Able to infect SMM. Grants AuthVars, Brick, SecBoot, BWP, BLE. Usually TpmBoot
- SecBoot = Able to defeat UEFI and/or Win8 Secure Boot
- TpmBoot = Able to defeat a "measured boot" that stores measurements in TPM
- AuthVars = Able to write to UEFI Authenticated Variables without needing a signature. Grants SecBoot. *May* imply SPI depending on BIOS update mechanism
- BLE = defeats BIOS_CNTL.BLE (BIOS Lock Enable) security bit
- BWP = defeats BIOS_CNTL.SMM_BWP (SMM BIOS Write Protect) security bit
- PRR = defeats Protected Range Registers

- If BLE and BWP and PRR are defeated or not used, it implies SPI
- Just remember "the more checkmarks, the more security is defeated" and then come back and read the slides later :)

Defeating Signed BIOS Enforcement[6]

Kallenberg et al. – MITRE (now LegbaCore)

- Presented the 2nd ever public BIOS exploit
- Corrupted SMM during malicious BIOS update. Takes control before any protections are set. Attacker can write anywhere in SPI, or just hang out in SMM.
- Found on Dell Latitude E6400. Thanks to code reuse, affected 22 total models.

√	√	√	√	√	√	√	√	√
SPI	Brick	SMM	SecBoot	TpmBoot	AuthVars	BLE	BWP	PRR

Thunderstrike[8]

Trammell Hudson – Two Sigma

- Improvement over existing work[7] that used PCI Option ROMs (OROMs) to bootkit Mac OS X
- Determined that OROMs still get loaded even when firmware updates are taking place. That's a major design flaw, assigned CVE-2014-4498
  - And of course the SPI has to be completely unlocked during update, therefore the OROM can write to it
- Confirmed that there is no special flash chip protection as [7] hypothesized
- Requires malicious thunderbolt dongle plugged in

	SPI	Brick	SMM	SecBoot	TpmBoot	AuthVars	BLE	BWP	PRR
[7]	X	X	X	N/A	N/A	√?	X	X	X
[8]	√	√	√	N/A	N/A	√	√	√	√

Speed Racer[9]

Butterworth & Cornwell et al. & Wojtczuk – MITRE/Bromium

- Independently hypothesized by Sam Cornwell and John Butterworth of MITRE and Rafal Wojtczuk of Bromium. Proved through implementation by Corey
- ***Architectural*** hardware race condition flaw that completely takes BLE off the table! It can no longer provide any protection
- Made a sampling of enterprise machines go from 30% vulnerable to 80%.

X/√(**d**epends)	√	X	√	X	√	√	X	X
SPI	Brick	SMM	SecBoot	TpmBoot	AuthVars	BLE	BWP	PRR

Extreme Privilege Escalation[10]

Kallenberg et al.– MITRE (now LegbaCore)

- 3rd and 4th ever public BIOS exploits. The most wide-spread BIOS memory corruption vulns ever. *Don't need kernel privilege to launch on either Windows or Linux.*
- Vulnerabilities found in UEFI open source reference implementation for the BIOS update code. Attacker gains arbitrary code execution in SMM before signatures are checked and before SPI is locked.
- Trickled down to *a lot* of systems (Phoenix vulnerable, AMI vulnerable, HP 33 enterprise/470+ consumer models, Dell 39 enterprise, Lenovo 45 enterprise). Most vendors (Acer/ASUS/Sony/Panasonic/Samsung/ Toshiba, etc) never reported what all is vulnerable.

√	√	√	√	√	√	√	√	√
SPI	Brick	SMM	SecBoot	TpmBoot	AuthVars	BLE	BWP	PRR

Venamis[11]

Wojtczuk – Bromium & Intel ATR

- Independently discovered Rafal Wojtczuk of Bromium and the Intel ATR team. Proved by implementation by Corey
- When you reboot, some BIOS protections become unlocked. Turns out also to be true when you go into a low power state (see VU#577140 if it ever gets published)
- The first code that executes at resume is supposed to lock the system back down. An “EFI boot script” gets interpreted to quickly configure and lock everything back down before handing off to the OS
- This script can be hijacked, and the attacker can get code execution at a time when the flash chip is unlocked, and play DMA games to get code execution directly in SMM

√	√	√	√	√	√	√	√	√
SPI	Brick	SMM	SecBoot	TpmBoot	AuthVars	BLE	BWP	PRR

Shout outs to my co-founder, Corey

- Corey has single-handedly taken down every BIOS security mechanism that he's gone up against…repeatedly. And we're all better for it, since he disclosed all the vulnerabilities

- Next up: Intel Management Engine (undefeated since 2009…)

√	√	√	√	√	√	√	√	√	???
SPI	Brick	SMM	SecBoot	TpmBoot	AuthVars	BLE	BWP	PRR	ME/AMT

Defense: What can you do ***today*** for ***free***

- Z0MG please go apply BIOS patches/updates!!!1!
  - They're just EXEs! I'm sure your patch management system can handle pushing and running EXEs!
- It does no one any good for us to work hard to discover and disclose firmware vulnerabilities and then have the vendors issue advisories and patches *if no one ever patches their BIOS!*
- Nor does it do *you* any good to act like you don't have exploitable vulnerabilities waiting to be targeted

Defense: What can you do ***today*** for ***free***

- Run MITRE Copernicus – long URL, just google "MITRE Copernicus" Designed for enterprise deployment
  - Vulnerability check && integrity check
  - Supports 32/64bit
  - Windows >= 7 Intel
  - Distributed as pre-signed binary, "Code for data" source code license available
  - Contact jbutterworth@mitre.org if you want the source and are willing to pilot on > 1k systems
- Run Intel ChipSec – http://github.com/chipsec/chipsec
  - Designed for modularity for security researchers
  - Vulnerability check. No integrity check
  - Very prominent warning.txt says not to run on production systems ;)
  - Supports Windows/Linux/UEFI Shell
  - Distributed as source, Requires you have a Windows kernel driver signing key to run on Windows

Defense: What can you do ***today*** for ***free***

- If you're trying to build something where you have some evidence that you're not owned right from startup (aka secure boot), you need to demand a "DRTM Measured Boot *with an STM*"
- "DRTM Measured Boot" today mostly means "Use Intel's open source 'tboot' as your boot loader"
  - It'll be a dramatic improvement over your current security posture, but unfortunately it's been known since 2009 that it's *architecturally* vulnerable, until we start getting STMs (SMM Transfer Monitors) embedded in our BIOSes.
    - So we're working to get STMs embedded BIOSes
      - But it would help us help you if Intel and the OEM's hear from customers that they *do* actually want their systems to be secure

Conclusion

- All state-sponsored attackers now know with 100% certainty that BIOS attacks are feasible, weaponizable, and used by other states in the wild
  - Welcome to the age of "me too!" for BIOS attacks
- The past couple years has seen a massive spike in vulnerabilities being responsibly disclosed
  - But you're not patching your BIOSes, are you?
  - Our data indicates that *99.95%* of un-patched enterprise machines contain at least one low level vulnerability that allows for bricking or backdooring the system
- Free options exist for you to start assessing your vulnerability (or infection) posture and start remediating it

Contact

- Twitter: @xenokovah, @coreykal, @legbacore
- Email: {xeno, corey}@legbacore.com
- http://legbacore.com/Contact.html for our GPG keys

- As always, go check out OpenSecurityTraining.info for the free classes from Corey and I on x86 assembly & architecture, binary executable formats, stealth malware, and exploits.
- Then go forth and do cool research for us to hear about!

References

[1] Evil Maid Just Got Angrier: Why Full-Disk Encryption With TPM is Insecure on Many Systems – Yuriy Bulygin – Mar. 2013
http://cansecwest.com/slides/2013/Evil%20Maid%20Just%20Got%20Angrier.pdf

[2] BIOS Chronomancy: Fixing the Core Root of Trust for Measurement – Butterworth et al., May 2013
http://www.nosuchcon.org/talks/D2_01_Butterworth_BIOS_Chronomancy.pdf
http://dl.acm.org/citation.cfm?id=2516714

[3] A Tale of One Software Bypass of Windows 8 Secure Boot – Bulygin et al.– Jul. 2013
http://blackhat.com/us-13/briefings.html#Bulygin

[4] 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

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

References

[6] 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 (not yet released)

[7] DE MYSTERIIS DOM JOBSIVS Mac EFI Rootkits - Loukas K (snare), Jul. 2012
https://media.blackhat.com/bh-us-12/Briefings/Loukas_K/BH_US_12_LoukasK_De_Mysteriis_Dom_Jobsivs_Slides.pdf

[8] Thunderstrike – Trammell Hudson, Dec. 2014 https://trmm.net/Thunderstrike_31c3 CVE-2014-4498

[9] 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

[10] 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

[11] 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

[12] See all the rest of stuff here: http://timeglider.com/timeline/5ca2daa6078caaf4

References

[13] “Mebromi: the first BIOS rootkit in the wild” http://www.webroot.com/blog/2011/09/13/mebromi-the-first-bios-rootkit-in-the-wild/

[14] “NSA Speaks Out on Snowden Spying” http://www.cbsnews.com/news/nsa-speaks-out-on-snowden-spying/

[15] "To Protect And Infect” https://www.youtube.com/watch?v=vILAIhwUgIU (contains leaked classified NSA documents)

[16] “U.S. Gas, Oil Companies Targeted in Espionage Campaigns” http://threatpost.com/u-s-gas-oil-companies-targeted-in-espionage-campaigns/103777

[17] “Summary of Attacks Against BIOS and Secure Boot” https://www.defcon.org/images/defcon-22/dc-22-presentations/Bulygin-Bazhaniul-Furtak-Loucaides/DEFCON-22-Bulygin-Bazhaniul-Furtak-Loucaides-Summary-of-attacks-against-BIOS-UPDATED.pdf also worth a read, even though it’s incomplete and they don’t include all our work ;)

Backup

- “Should you worry when the skullhead is in front of you? Or is it worse because it’s always waiting, where your eyes don’t go?”
  – They Might Be Giants

Evil Maid Just Got Angrier [1]

Bulygin et al. - Intel

- Using TPM support with MS BitLocker FDE is supposed to not allow the system to boot if the measurements of pre-boot code like the BIOS change. This is to prevent a present attacker from stealing the FDE password or key
- Established that *SPI* may imply *TpmBoot*
- Didn't present a new way to break into SPI, just assumed it was unlocked

√	√	√	√	√	√	√	√	√
SPI	Brick	SMM	SecBoot	TpmBoot	AuthVars	BLE	BWP	PRR

BIOS Chronomancy [2]

Butterworth et al. - MITRE

- The "Tick" Showed how all implementations of TpmBoot at the time were *fundamentally* incapable of providing security guarantees in the presence of a BIOS level attacker
- Established that *SPI* always implies *TpmBoot* (not just in the presence of implementation bugs)
- The "Flea" showed an attacker can persist across BIOS updates (so you can't use that as a valid cleaning strategy)
- But this work also enabled a stronger TpmBoot

√	√	√	√	√	√	√	√	√
SPI	Brick	SMM	SecBoot	TpmBoot	AuthVars	BLE	BWP	PRR

A Tale of One Software Bypass of Windows 8 Secure Boot [3]

Bulygin et al. - Intel

- Concretely established what we already know to be architecturally true. If SPI is unlocked, and the attacker can infect it, they get to run first, and Secure Boot won't work.
- Presented 5 specific places to target to defeat Secure Boot once you can write to SPI.
- Just assumed SPI is unlocked, but that's an unfortunately common occurrence on anything unpatched and sold a couple years ago

√	√	√	√	√	√	√	√	√
SPI	Brick	SMM	SecBoot	TpmBoot	AuthVars	BLE	BWP	PRR

Setup for Failure[4]

Bulygin et al. & Kallenberg et al. – Intel/MITRE (now LegbaCore)

- Independently discovered by Bulygin's team and Corey (though only Corey realized it could be used to brick a system)
- There exists a UEFI non-volatile variable "Setup" that is crucial to secure boot but is not marked as *authenticated*. Therefore anyone who understands its structure can modify it and turn of Secure Boot a couple different ways

X	√	X	√	X	X	X	X	X
SPI	Brick	SMM	SecBoot	TpmBoot	AuthVars	BLE	BWP	PRR

All your boot are belong to us[5]

Bulygin et al. - Intel

- Intel team publicly disclosed 5 new BIOS configuration vulnerabilities that they had been privately disclosing to vendors under NDA for almost a year
- Vendors were for the most part not fixing quickly, and we doubt some of them ever issued patches to fix their old models, and only let their new models be fixed when they got increased protection from their IBV

X	√	X	√	X	X	X	X	X
SPI	Brick	SMM	SecBoot	TpmBoot	AuthVars	BLE	BWP	PRR