Advanced x86:
BIOS and System Management Mode Internals
Motivation

Xeno Kovah && Corey Kallenberg

LegbaCore, LLC

All materials are licensed under a Creative Commons “Share Alike” license.

http://creativecommons.org/licenses/by-sa/3.0/

You are free:

to Share — to copy, distribute and transmit the work

to Remix — to adapt the work

Under the following conditions:

Attribution — You must attribute the work in the manner specified by the author or licensor (but not in any way that suggests that they endorse you or your use of the work).

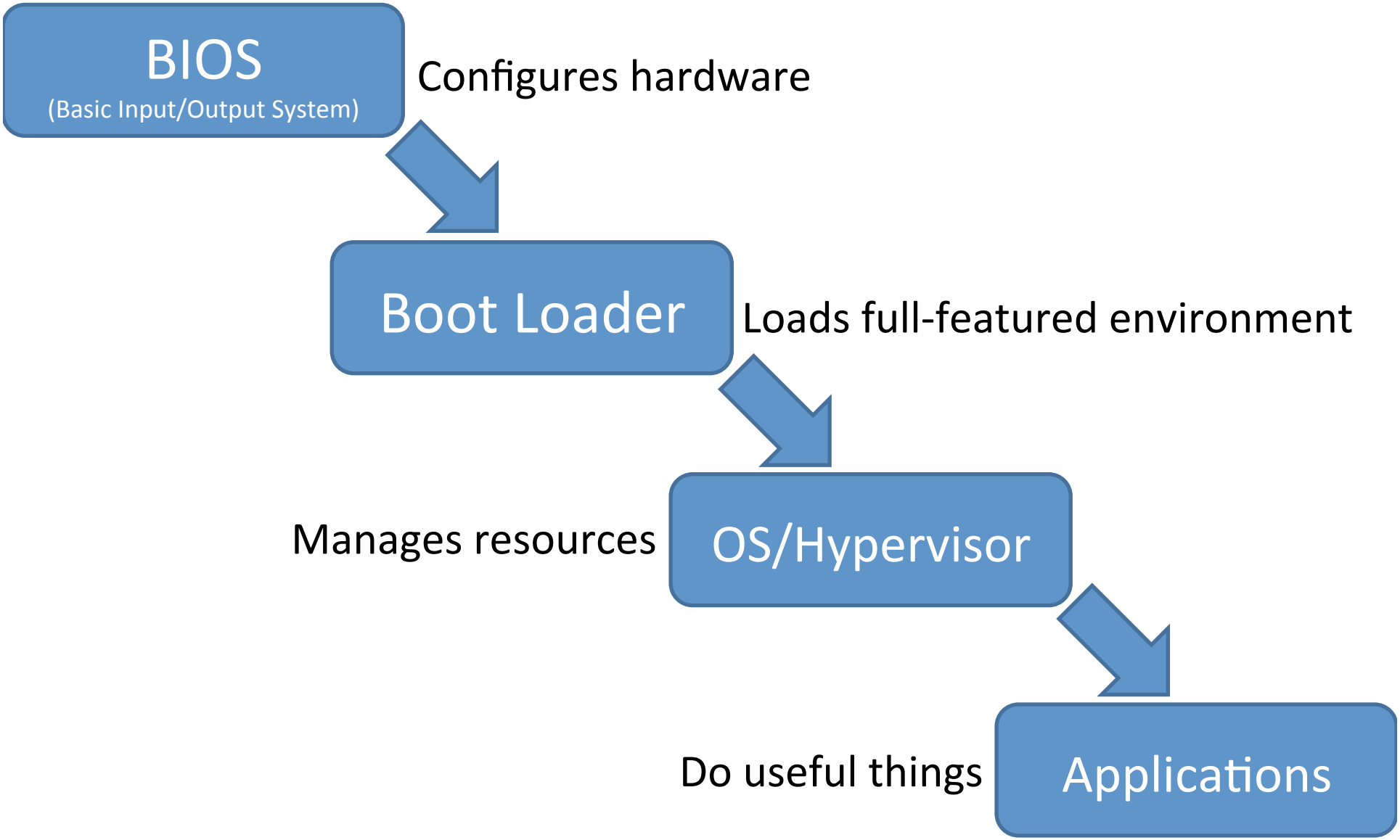
Share Alike — If you alter, transform, or build upon this work, you may distribute the resulting work only under the same, similar or a compatible license.

Attribution condition: You must indicate that derivative work
"Is derived from John Butterworth & Xeno Kovah’s ’Advanced Intel x86: BIOS and SMM’ class posted at http://opensecuritytraining.info/IntroBIOS.html”

Background

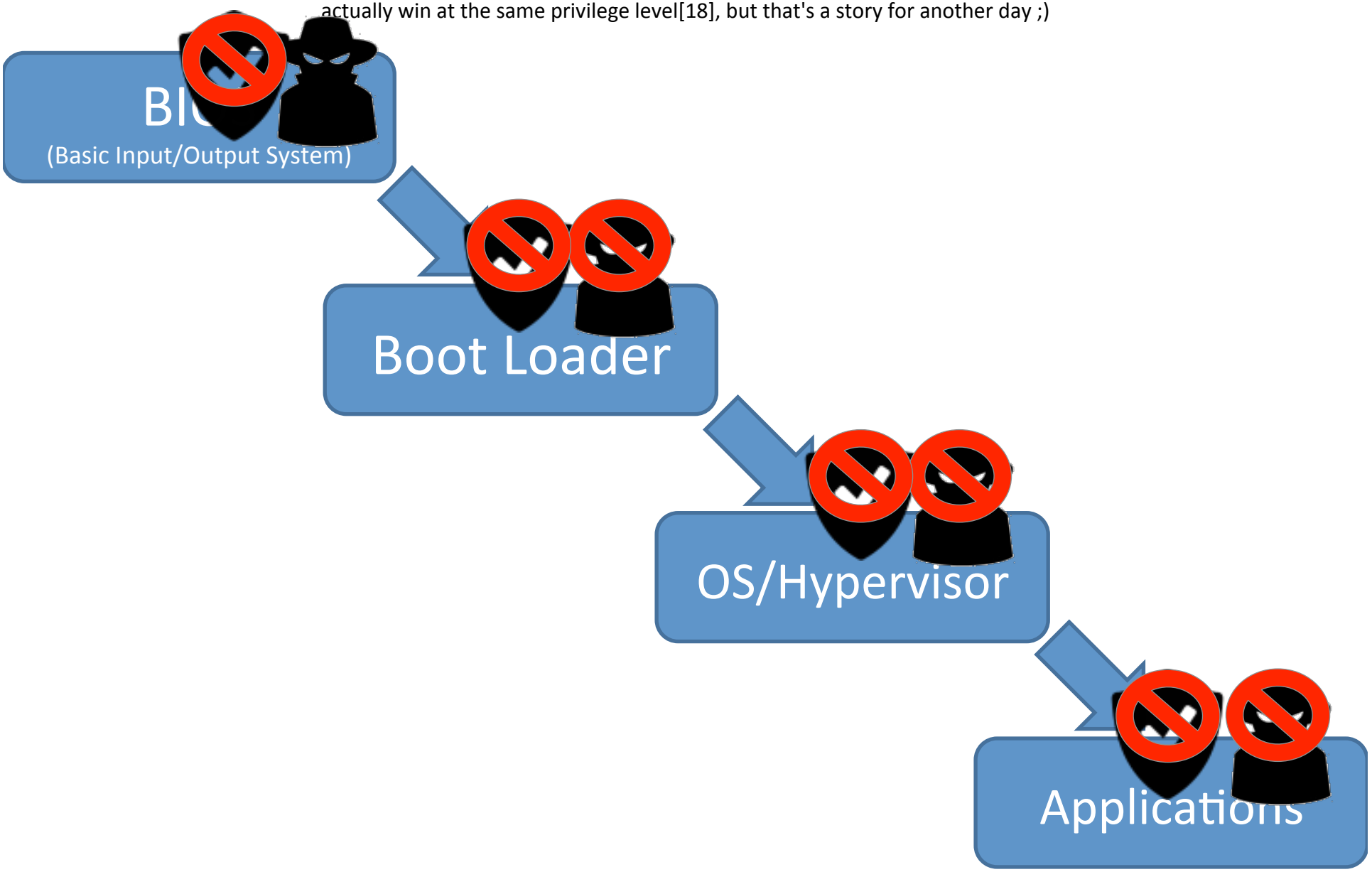
- Things you should *ideally* know before you start to learn BIOS attack/defense:
  - x86 assembly
    - http://OpenSecurityTraining.info/IntroX86.html
  - x86 architecture (execution modes, segmentation, virtual vs. physical addresses, port IO)
    - http://OpenSecurityTraining.info/IntermediateX86.html
  - Reverse engineering
    - http://opensecuritytraining.info/IntroductionToReverseEngineering.html
  - Portable Executable binary format
    - http://OpenSecurityTraining.info/LifeOfBinaries.html
  - All the different way to find and exploit vulnerabilities (except the ways to get around anti-exploit techniques, because there are none :))
    - http://OpenSecurityTraining.info/Exploits1.html

How computers do useful things



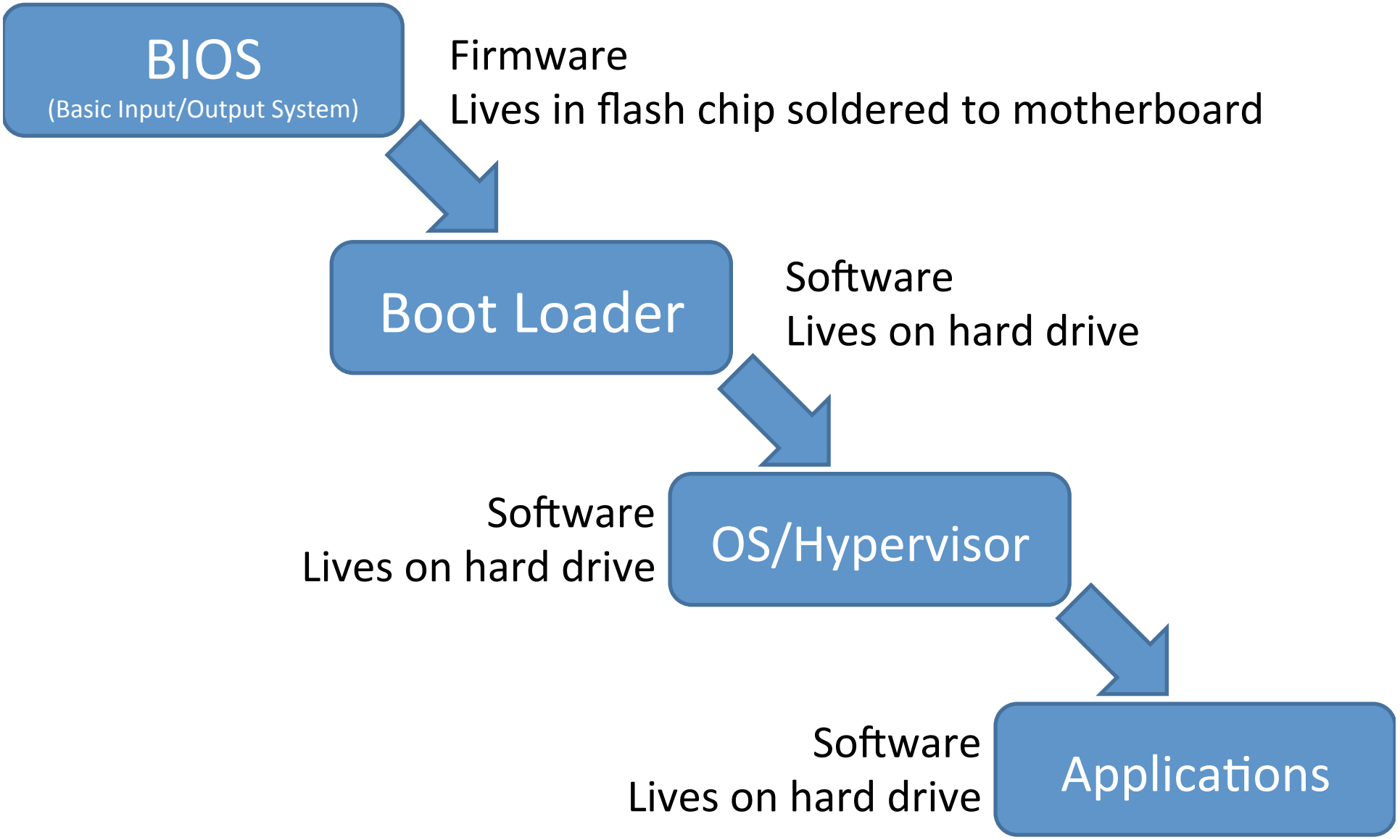
A brief history of the world('s insecurity)

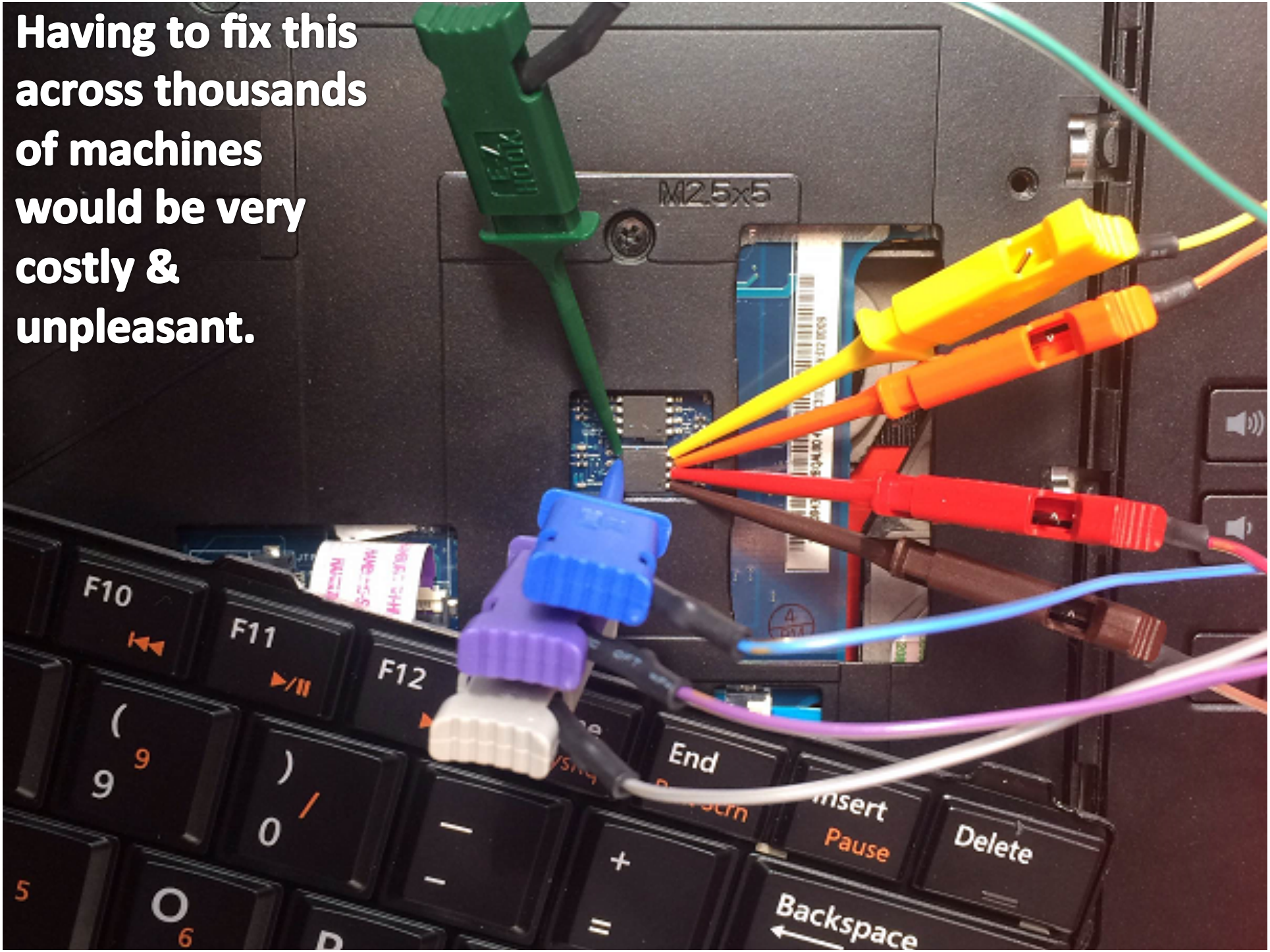
"Insanity: doing the same thing over and over again and expecting different results." – Albert Einstein
This is why I think people need to put more research effort in my hobby-horse area of Timing-Based Attestation…which can actually win at the same privilege level[18], but that's a story for another day ;)



One of these things is not like the others

One of these things is not well understood




Having to fix this across thousands of machines would be very costly & unpleasant.
M2.5x5
F10
F11
F12
End
Insert
Pause
Delete
Backspace

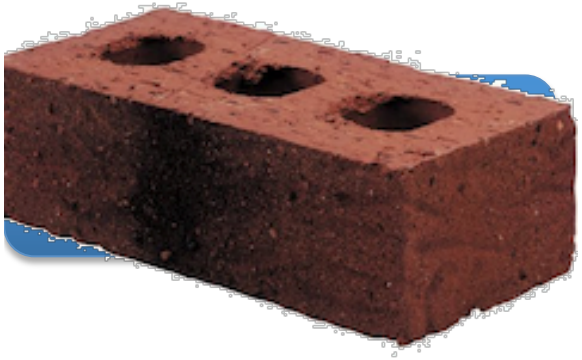
Why Hack the BIOS?

- Why would someone want to hack the BIOS?
  - "Because it is a free pass for persistence forever."
  - Because very few organizations, are checking the BIOS
  - If you get a foothold there, it's likely to be there for years
  - Low level code initializes the system so the lower the attacker goes the further up the malware food chain he is

- Is hacking the BIOS itself the end-goal?
  - Likely not, it's a stepping stone to get into SMM or some other portion of the system
  - And to keep the foothold for years and years
  - Likely one target will be SMM, since this provides the attacker nigh-maximum privileges (aka "God Mode"), but there are other targets too

- What can attackers who have infiltrated the BIOS do?
  - "Anything the hardware lets them!"
  - If the processor can reach out and touch it, then infiltrating the BIOS gives you power over it
  - He who runs first, runs best

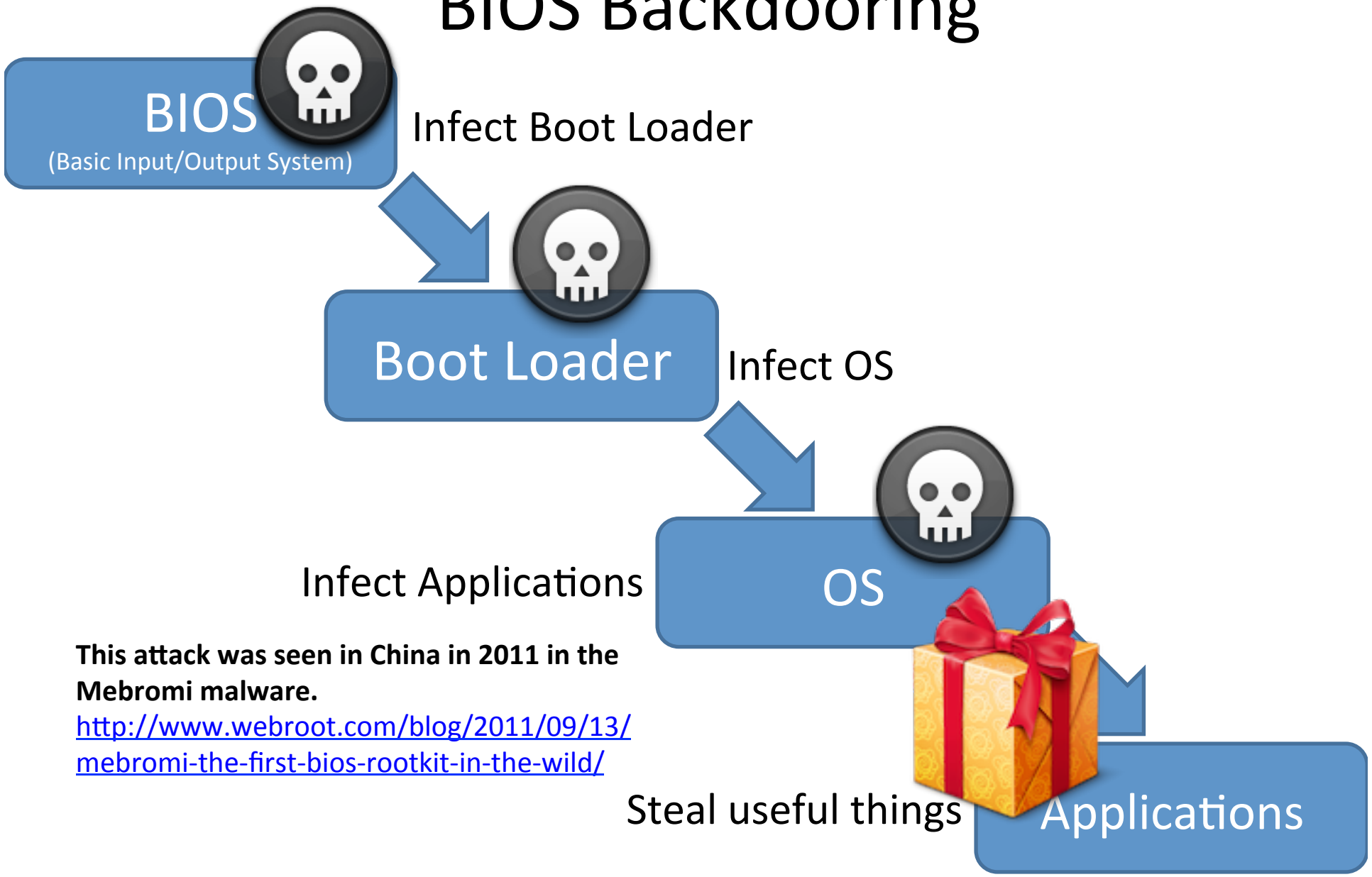
Example Attacks
BIOS "Bricking"



Firmware is corrupted (1 byte is all that's needed)
System will not boot

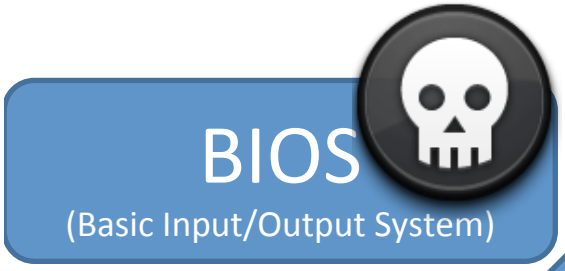
The CIH virus did this as a time-bomb attack on (*supposedly* 60 million) computers in 1998
http://en.wikipedia.org/wiki/CIH_(computer_virus)

Example Attacks
BIOS Backdooring

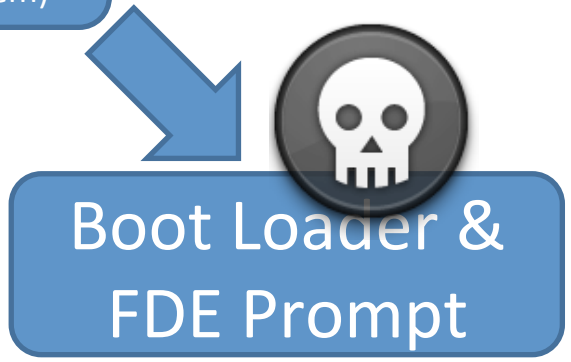


This attack was seen in China in 2011 in the Mebromi malware.
http://www.webroot.com/blog/2011/09/13/mebromi-the-first-bios-rootkit-in-the-wild/

Example Attacks
Uber Evil Maid



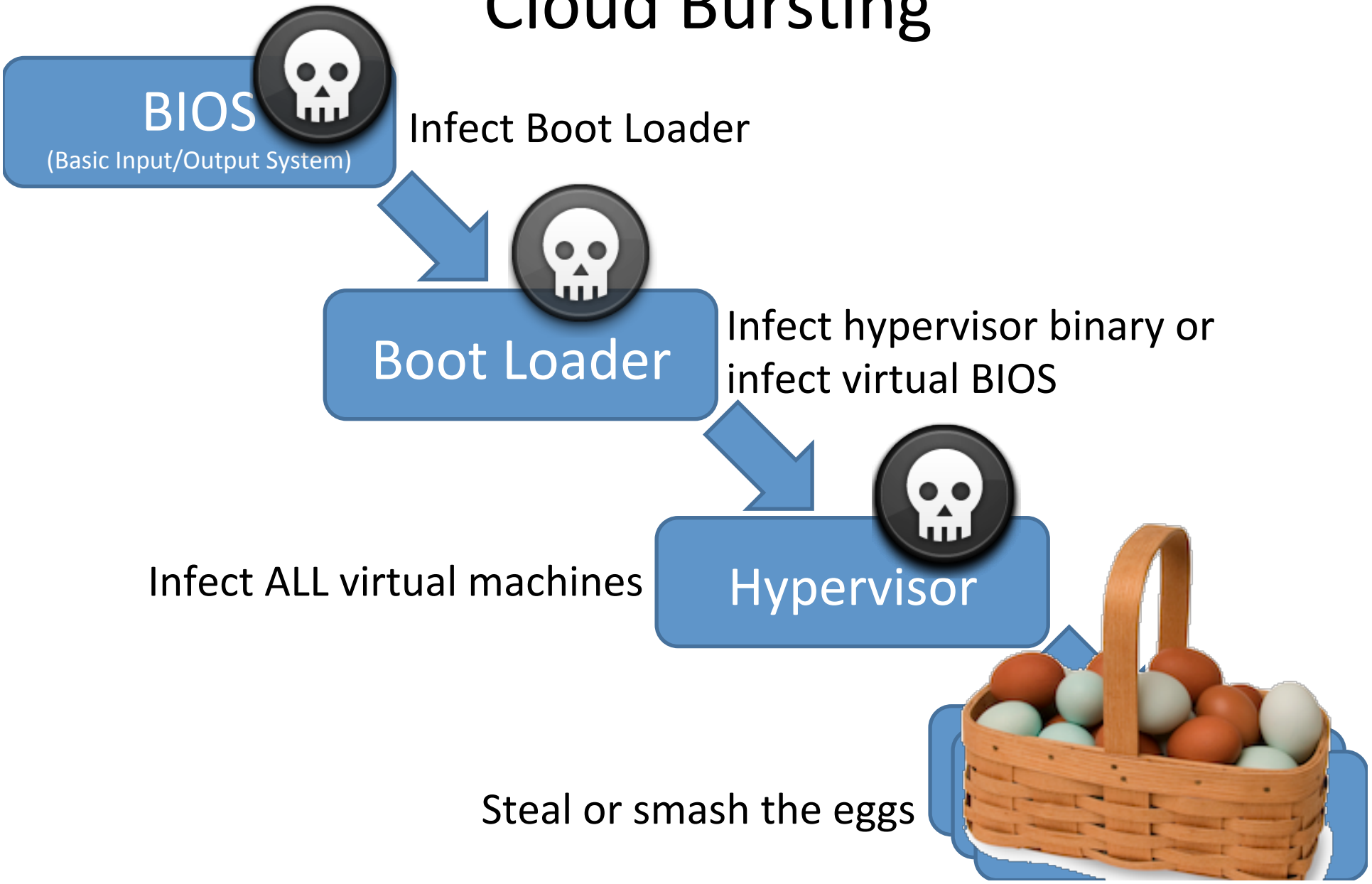
Hook Full Disk Encryption (FDE) software in memory



Capture FDE password
Store in BIOS or unused HD space

Steal full HD contents at one's leisure

Example Attacks
Cloud Bursting



One Stealth Malware Taxonomy

Your view of the world?

- Ring 3 – Userspace-Based
- Ring 0 – Kernel-Based

Yeah, I'm pretty fuckin badass…
I totally broke into the web browser (userspace application) remotely

Nah bro, I'm the hot shit here
I totally broke into the kernel and am now running my uber1337 r00tk4t

One Stealth Malware Taxonomy

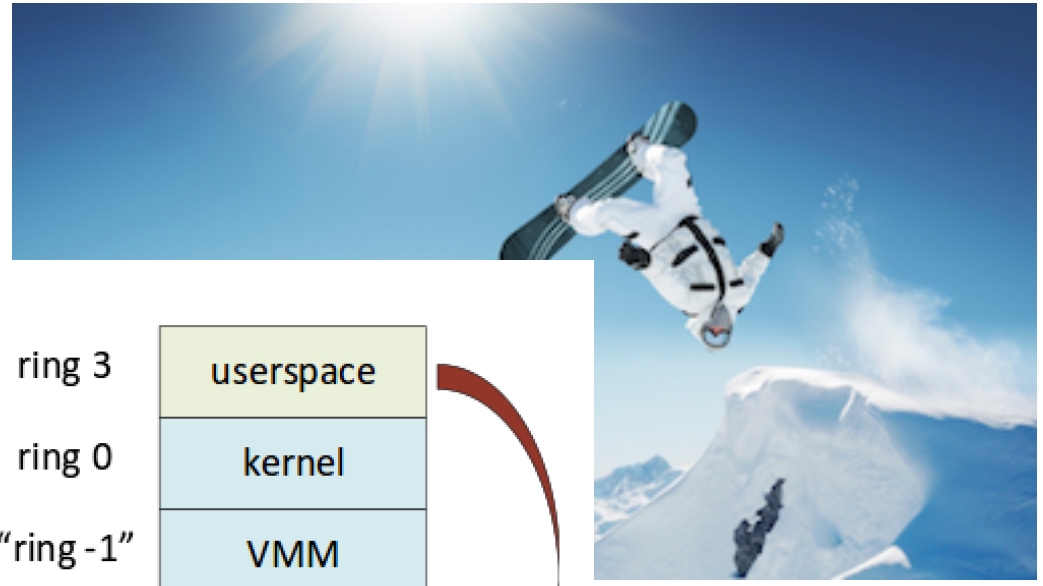
Welcome to the Deep Dark!

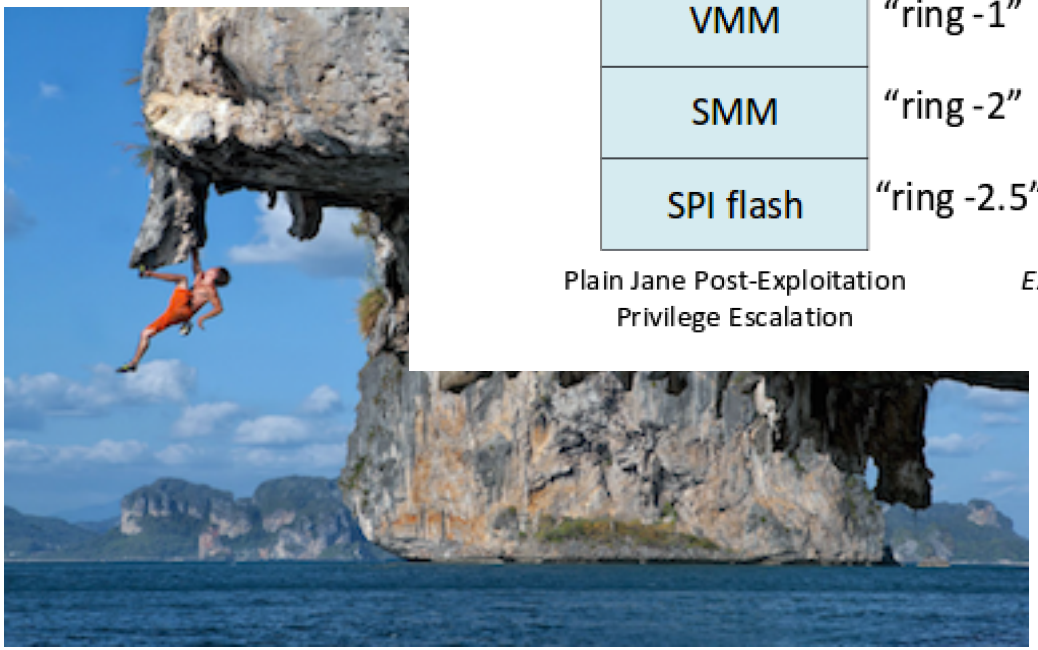
We start where others end!

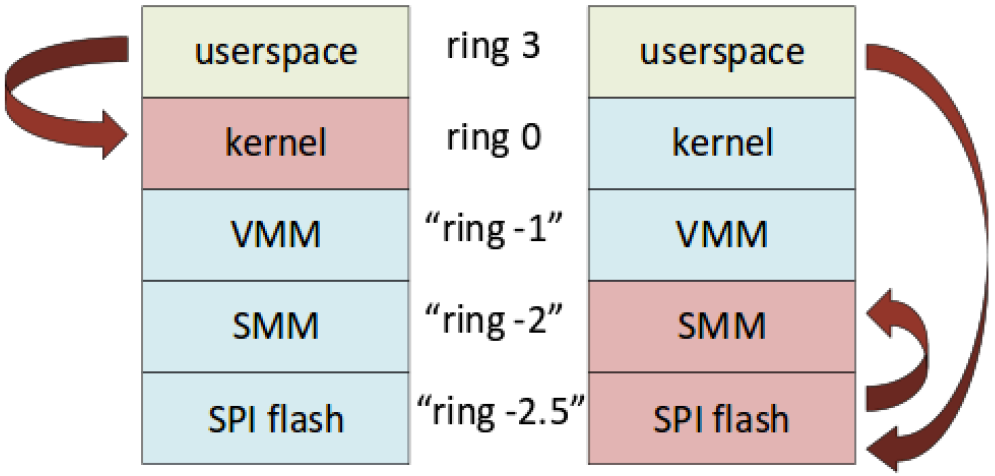
- Ring 3 – Userspace-Based
- Ring 0 – Kernel-Based
- "Ring -1" – Virtualization-Based
  - Intel VT-x(Virtualization Technology for x86), AMD-V (AMD Virtualization), Hypervisor subverted
- "Ring -1.5?" - Post-BIOS, Pre OS/VMM
  - e.g. Master Boot Record (MBR) "bootkit"
  - Peripherals with DMA(Direct Memory Access) (this can be ring 0, -1, or -1.5 depending on whether VT-d is being used)
  - Not a generally acknowledged "ring", but the place I think it fits best
- "Ring -2" – System Management Mode (SMM)
- "Ring -2.25 – SMM/SMI Transfer Monitor (STM)
  - A hypervisor dedicated to virtualizing SMM
  - Another one of my made up "rings", I just added this ring for this presentation :)
- "Ring -2.5" - BIOS (Basic Input Output System), EFI (Extensible Firmware Interface)
  - because they are the first code to execute *on the CPU* and they control what gets loaded into SMM
  - Not a generally acknowledged "ring", but the place I think it fits best
- "Ring -3" – Chipset Based - *not valid anymore on modern architectures*
  - Intel AMT(Active Management Technology)/ME(Management Engine) – Now just ring ?
  - Could maybe be argued that any off-CPU, DMA-capable peripherals live at this level?

But BIOS could use VT-d to prevent DMA, and it initializes peripherals, so…?
Yeah, things get squishy (non-precise) at the bottom with non-real-rings.

Which is why what we can do is *Extreme!*







System Management Mode (SMM) is the true "God Mode" on x86 systems

- BIOS loads SMM code
- SMM can read/write everyone else's memory
- No one can read/write SMM's memory once it has been locked by the BIOS
  - Unless they have an exploit ;)
- Only the PC makers should be able to change the code in the BIOS (digitally signed)
  - Unless they have an exploit ;)
  - Or unless they have physical access ;)

Threats

- In Sept. 2011 the first crimeware (Mebromi) was found using BIOS infection [13]
- In Dec. 2013 NSA defensive director said other states are developing BIOS attack capabilities [14]
- In Dec. 2013 Snowden leaks said NSA's offensive side had a catalog of 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]
- In Jun. 2015 the HackingTeam leaks[18] showed that they had developed a UEFI-based persistence mechanism to install their typical Windows RAT

“DarkSeoul”

- Attributed to the North Koreans
- Targeted at South Korea banks
- Included MBR wiper malware that induced direct economic loss
- “Tens of thousands” of machines rendered unbootable
  - Aided by compromising software update servers

Sony Hack

- Attributed to the North Koreans
- Primarily theft of information and extortion
- Included MBR wiper malware that induced direct economic loss
- I was a bit skeptical at first, until it was said that the reason the FBI was so certain was because the NSA folks said so :)
  - http://www.nytimes.com/2015/01/19/world/asia/nsa-tapped-into-north-korean-networks-before-sony-attack-officials-say.html

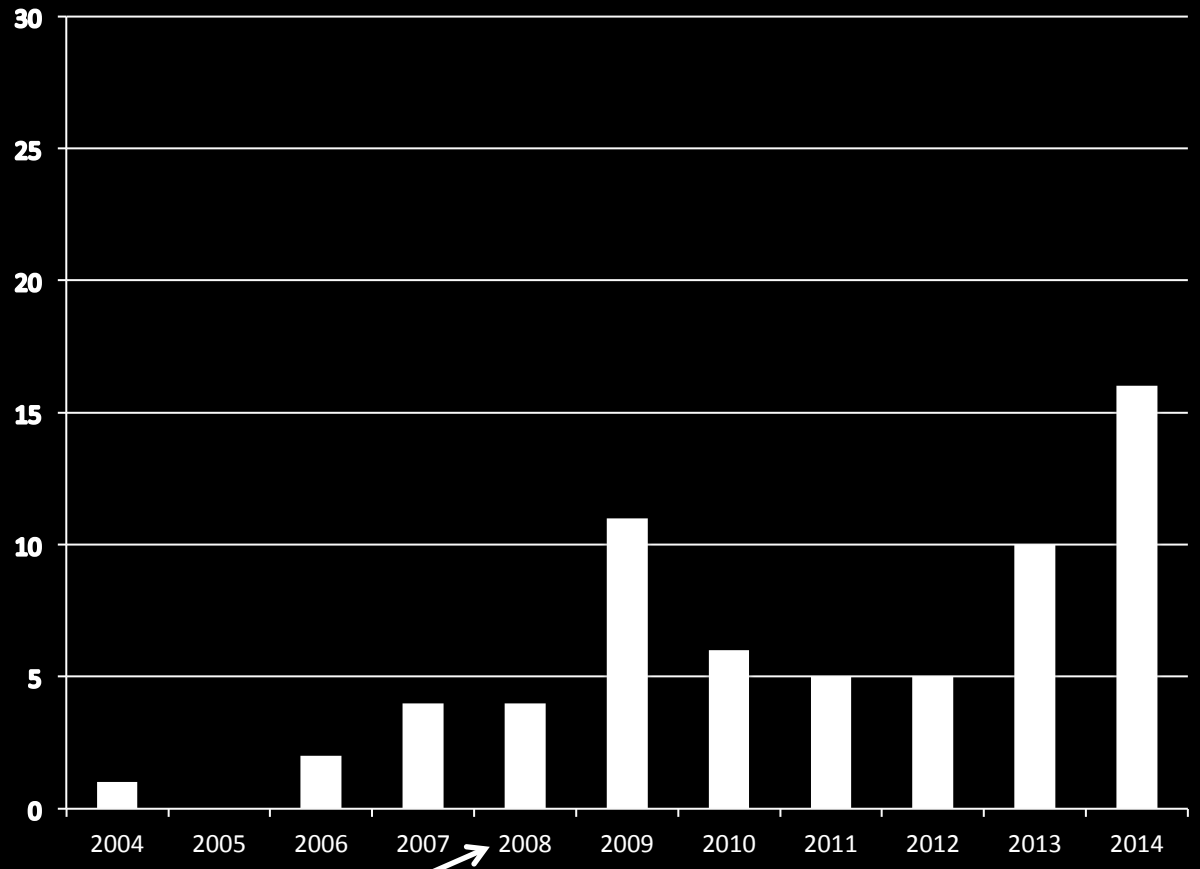
“Shamoon”

- Attack against Saudi Aramco
- Attributed to Iranians against their regional rival Saudi Arabia
- Included HD wiper malware introduced direct economic loss
- Took down 30,000+ systems
  - What if it had been BIOS malware? :)

From ShmooCon 2015 talk "Betting BIOS Bugs Won't Bite Y'er Butt?"
http://www.legbacore.com/Research_files/2015_ShmooCon_BIOSBugs.pdf

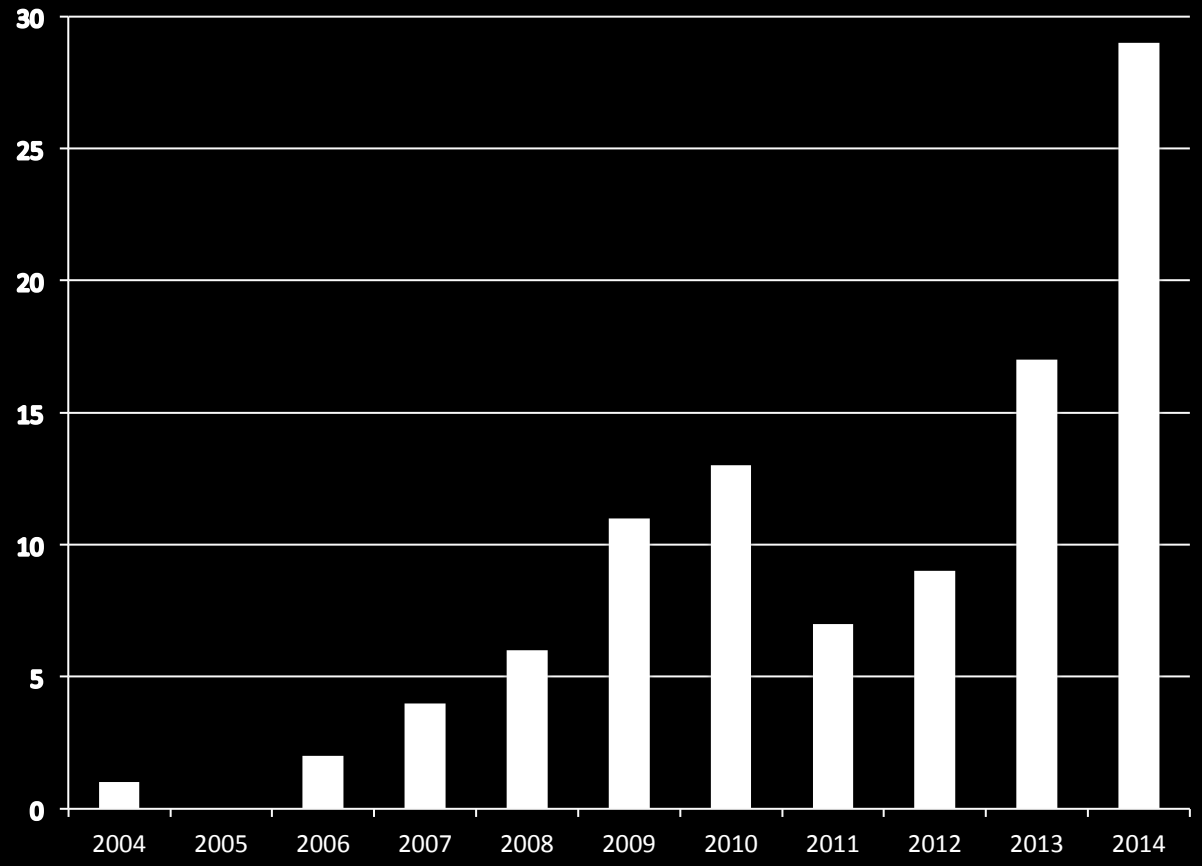
BIOS/SMM/OROM/DMA/ACPI/ME/TXT/Firmware Attack Talks

(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

So the questions are:

- Can you tell a limited and targeted BIOS corruption from a hardware failure?
- Are you going to give BIOS attackers a free pass to live on your machines & networks forever?
- Would you know what to do to detect them even if you wanted to?
- What if a HD-wiping adversary steps up their game and becomes a BIOS-wiping one? Are you prepared to recover from that?
- Do you want to learn about REAL ULTIMATE POWER?!


BRING IT ON!

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 – Trammel 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=vILAlhwUgIU (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 ;)

[18] https://twitter.com/NikolajSchlej/status/618076694117789696