Advanced x86:
BIOS and System Management Mode Internals
SMM Conclusion

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”

SMM Lockdown Summary

- Although these may generally only be implemented by the vendor, you can verify most of these

- Use TSEG
- Ensure entire SMRAM range is contained in the protected space
- SMI handler code should not execute code outside the protected memory range
- Don't use interrupts in SMM, unless you explicitly WBINVD the cache before generating an interrupt
- Ensure D_LCK is set to lock down both memory-mapping registers as well as SMRAM
- Verify SMRR are supported
- Verify SMRR are used
- Verify SMRR range overlaps/matches TSEG
- Verify SMM_LOCK bit is asserted to prevent an attacker from suppressing SMI#
- Verify the SMM_BWP bit is set in the BIOS_CNTL register to permit writes to flash only when processor is in SMM

SMM Conclusion

- Holds a lot of responsibility in protecting the system
  - Protects the BIOS flash
  - Protects itself, because it is instantiated by the BIOS from binary on the BIOS flash
- So it is very fragile in case of a writeable BIOS
  - It is not difficult to locate and “carve” out the SMI code module and replace it with a malicious one
  - Once written to BIOS the attacker can lock down the once-vulnerable system
  - Which highlights a general problem with tools like Copernicus. We’ll touch on this at the end of the Trusted Computing section
- Bottom line:
- If the attacker can write to the BIOS, they can modify SMM (and a lot of other stuff, unlocking protections, etc.)
- Therefore, the most important thing to lock down is the SPI Flash, first and foremost.
  - The protection of which relies first and foremost on SMM