BITARMOR

Braving the Cold: New Methods for Preventing Cold Boot Attacks on Encryption Keys

Patrick McGregor, Ph.D.
Tim Hollebeek
Alex Volynkin, Ph.D.
Matthew White

BitArmor Systems, Inc.

BITARMOR

Outline

- Who cares about Full Disk Encryption, anyway?
- The anatomy of a Cold Boot Attack
- New software-based methods for defense
  - Tidy up at power down time
  - Built-in temperature monitoring
  - Taking advantage of default BIOS behavior
  - Efficient virtual compartmentalization
- Thoughts for the future

BITARMOR

What is Full Disk Encryption (FDE)?

- **Encrypts every bit of data on a disk or disk volume**
  - Mostly used to encrypt laptop drives
  - Uses standard algorithms, e.g., AES, Triple DES
- **User authentication used to decrypt disk keys**
  - Some use different keys for different partitions
- **In real time, sectors of a disk are read/written without impacting the user**
  - The keys are stored persistently in memory to ensure high performance

BITARMOR

FDE: Why Do People Buy This Stuff?

- **To mitigate risk**
  - Lowers chance losing data, being sued, being fined
  - Data breaches can cost between $90 and $305 per record exposed
  - Average cost: $4.8 million per company per incident
- **Compliance**
  - Industry government regulations say certain data has to be encrypted
    - PCI DSS, OMB M-06-16, others
- **Avoiding breach notification requirements**
  - Laws in at least 39 states force disclosure of incidents
  - Encryption is a “get out of jail free” card

User Work Patterns Can Increase Risks

- **BitArmor survey of 250 business users**
  - More than 40% of users leave laptops in sleep or hibernation mode when traveling
  - No difference between techie users and business users!
- **Desktops matter too!**
  - Cold boot attacks apply to any PC type
  - Desktops in screen lock mode are vulnerable

The FDE Market

- **How many companies have invested in it?**
  - 20% of companies reported encrypting laptops in 2007*
    - Most common application of encryption
  - Based on analyst estimates, over $200 Million sold in 2007**
  - $1 Billion total market potential

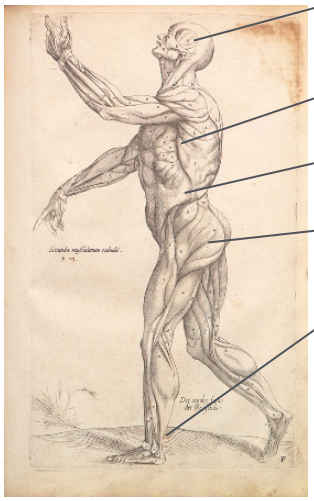
**Ponemon Institute: 2008 Annual Study: U.S. Enterprise Encryption Trends*

***The 451 Group, deal analysis, Nov. 22, 2006, October 9, 2007*

BITARMOR

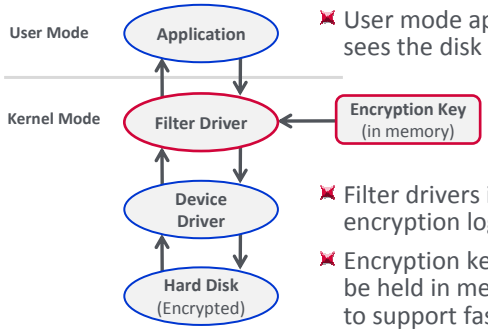
Anatomy of a Cold Boot Attack

- Physical and computing principles
- Attack methods
- Key recovery and reconstruction



BITARMOR

Disk Encryption Internals



- User mode application sees the disk as unencrypted
- Filter drivers implement disk encryption logic
- Encryption key must be held in memory to support fast disk I/O

BITARMOR

Disk Encryption Assumptions

- **Encryption keys**
  - Disk encryption key is unlocked during pre-boot authentication, and held in memory
  - With standard algorithms, the disk encryption key and disk decryption key are the same
  - Security of system depends on secrecy of decryption key
- **PC Memory (DRAM)**
  - Standard PC memory is based on small capacitors, which slowly leak over time
  - It does not hold information indefinitely, but needs to be periodically "refreshed"
  - **It is assumed that a disk decryption key cannot be recovered from DRAM after power is removed**

BITARMOR

Disk Encryption Reality

DECRYPTION KEY BITS
PERSIST IN DRAM
EVEN AFTER POWER IS LOST

BITARMOR

Memory Remanence

- **DRAM drainage rates can be slow**
  - Information stored in DRAM becomes irrecoverable on a fairly short timescale (seconds)
  - This time can be substantially longer on older hardware (c. 1999-2003)
- **Memory is not cleared during reboots**
  - Some machines zero out memory via a Power On Self Test (POST), but it is usually disabled
  - ECC memory may also clear memory during initialization, but may systems do not use ECC
  - OSes do not assume memory contains zeroes; they are responsible for initializing it, and provide the "illusion" that data doesn't survive reboots.

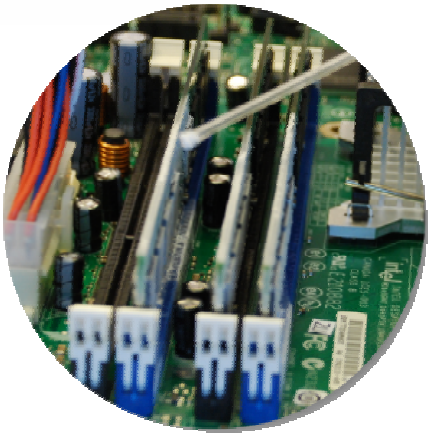
BITARMOR

A Cold Boot Attacker's Bag of Tricks

- **Booting an alternative operating system**
  - A custom, alternative OS may boot and record memory values instead of overwriting them
  - Alternative OSes may be quite small, overwriting very little memory
  - Alternative OSes may be delivered via a wide variety of methods: USB, floppy, network …
- **Physically transferring DRAM chips**
  - Memory chips may be transferred to an alternative computer with better characteristics (POST disabled, no BIOS password, spare hard drive for storage, no ECB support, etc.)
  - The allowable time period for a transfer can be significantly extended by chilling the memory to -50° C or colder

BITARMOR

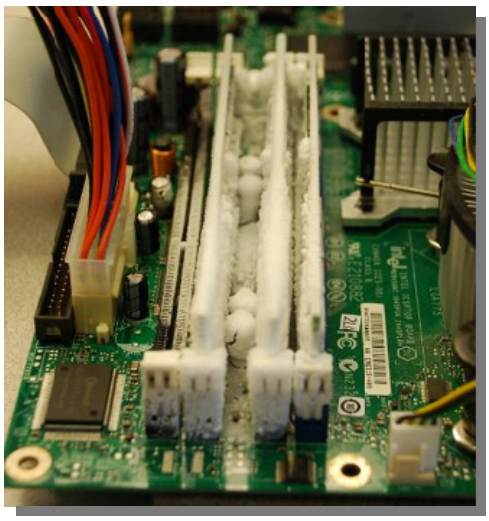
Cooling Down RAM



- **Several cooling techniques are possible**
  - Natural environmental cooling
  - Aerosol cans
  - Liquid nitrogen

BITARMOR

Cold RAM!



BITARMOR

Key Recovery and Reconstruction

- **Keys are readily identifiable in DRAM**
  - Key material, and expanded key schedules in particular, have very distinctive patterns in memory
  - May not be necessary to understand memory layout or reverse engineer the encryption software in order to recover the key
- **Princeton reconstruction algorithms are efficient**
  - Redundancies allow keys to be recovered even in the presence of a moderate number of bit errors in the key schedule
  - Reconstruction possible unless bit value ambiguity makes brute forcing all possibilities infeasible
  - 25% error rates are tolerable in certain cases

BITARMOR

New Software Defenses against Cold Boot Attacks

- **Implement several defenses against the most feasible Cold Boot Attack scenarios**
  - Use software, not any new hardware
- **Address scenarios where computer physically stolen:**
  - Shortly after being turned off
  - While hibernating
  - While sleeping
  - While screen locked

BITARMOR

Defense # 1: No Power, No Keys*

- Address scenarios where computer physically stolen:
  - **Shortly after being turned off**
  - **While hibernating**
  - While sleeping
  - While screen locked
- **Idea: Discard keys in memory immediately before power down**
  - Princeton paper citation not sufficient for FDE keys
  - We propose a simple OS-driven approach

* Patent pending.

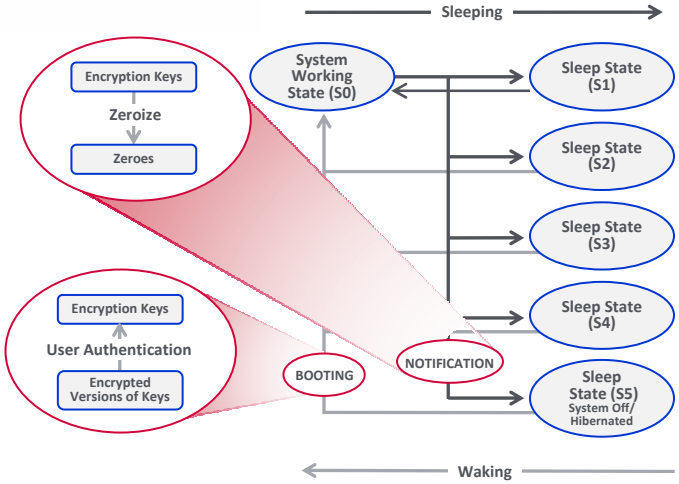
BITARMOR

Key Scrubbing

- Prevents key material from being available after shutdown or hibernation
- Machine must be “cleanly” shutdown or hibernated
- Feasible through Windows OS mechanisms

BITARMOR

Protect Keys by Watching State Transitions



BITARMOR

Implementation

- **Notification**
  - Handle power state IRP in filter driver
    - Recognize when machine entering state S5
  - Overwrite cryptographic keys with zeroes
    - In memory (DRAM)
- **Booting**
  - Obtain keys using an authentication procedure
- **Possible for Windows 2000, XP, Server 2003, Vista, Server 2008**

BITARMOR

Additional Defenses

- **Address scenarios where computer physically stolen:**
  - Shortly after being turned off
  - While hibernating
  - **While sleeping**
  - **While screen locked**

- **Consider three attack vectors:**
  1. Booting alternate OS (remote or local), no RAM transfer
  2. Cooling RAM before power loss, RAM transfer
  3. Cooling RAM immediately after power loss, RAM transfer

BITARMOR

Defense # 2: BIOS Is Our Friend*

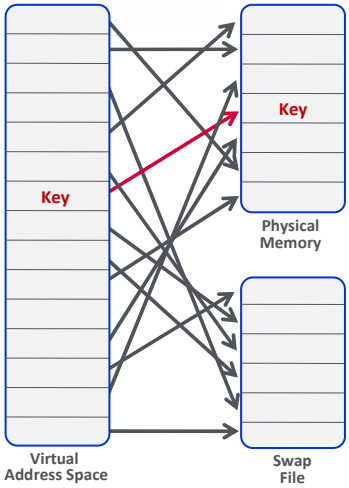
- **Consider three attack vectors:**
  1. **Booting alternate OS (remote or local), no RAM transfer**
  2. Cooling RAM before power loss, RAM transfer
  3. Cooling RAM immediately after power loss, RAM transfer

- **Idea: Take advantage of certain specific default behavior of BIOS that would apply to all PC architectures**
  - The defense will work no matter what OS is used by an attacker

* Patent pending.

BITARMOR

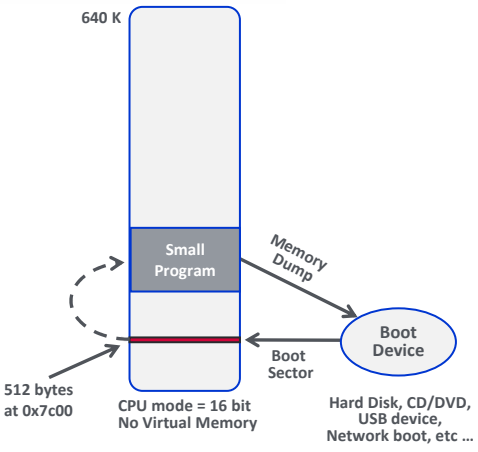
Virtual Memory



- Encryption keys should be in non-paged memory, but are not usually stored at a fixed physical address, and could be anywhere.
- This could be considered an advantage, but crypto material is generally easy to find, especially if all physical memory can be dumped for later analysis.
- Unless you're a hardware guy, you rarely have to think about physical memory or physical addresses. But sometimes physical addresses are **very important**.

BITARMOR

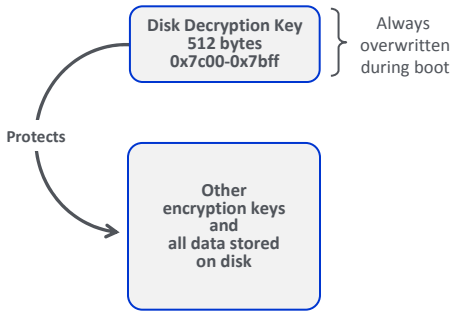
BIOS Boot



- Immediately after powering up, machine is in a 1980s era configuration
- Initial 512 bytes of a program are loaded at a **fixed address** and executed
- Other contents of memory are *undisturbed*
- Code necessary to scan memory for encryption keys, or to simply copy all physical memory for later analysis can be loaded into a very small region of memory.
- Remnants of encryption keys can be recovered with very high probability. **Unless …**

BITARMOR

0x7C00 Defense



- **... unless** your master disk keys are stored in the physical address range 0x7c00 – 0x7bff.
- A disk encryption filter driver loads very early in the boot process and can allocate this memory range for its exclusive use
- 512 bytes is enough room for multiple disk encryption keys, or for an AES-256 key schedule
- Any attempt to boot to **ANY** alternative operating system or device will overwrite the keys stored in this address range.
- When the disk encryption key is destroyed during the boot process, the disk information is no longer recoverable.

BITARMOR

Defense # 3: Watch for Fleeing Joules*

- **Consider three attack vectors:**
  1. Booting alternate OS (remote or local), no RAM transfer
  2. **Cooling RAM before power loss, RAM transfer**
  3. Cooling RAM immediately after power loss, RAM transfer

- **Idea: Detect and respond to RAM cooling**
  - Princeton paper only discussed specialized hardware for detecting temperature variations
  - We can instead use common, built-in sensors!

* Patent pending.

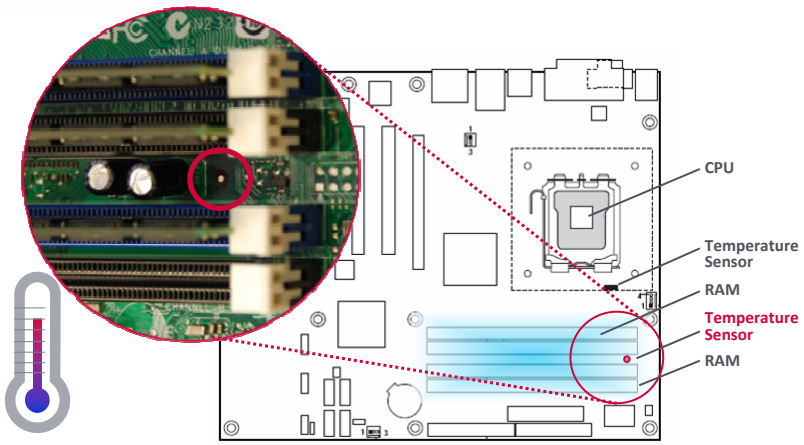
BITARMOR

Temperature Detection

- Modern motherboards have temperature sensors embedded for heat control in various zones including RAM
- Sensor data is readily available from the OS and BIOS
- Sensitive decryption keys can be erased by software when extreme drop in temperature for RAM zone is detected

BITARMOR

On-board Temperature Sensors in RAM Zone



BITARMOR

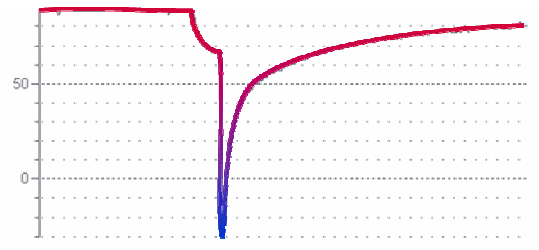
Sensor Reliability Facts

- Sensors can report low temperatures to -65C ° with accuracy ±3C °
- Sensors are infused into the board. This mitigates risk of sensor tampering.
- Sensor response time is on the order of milliseconds. Actions can be taken immediately.

BITARMOR

Cooling Down RAM

- Cooling down RAM causes measurable temperature drop in Memory Zone



RAM blasted with Difluoroethane (aerosol propellant).
Sensor registers -30F°

BITARMOR

Erasing Decryption Keys

- **Sensitive decryption keys can be immediately erased by software when extreme drop in temperature for RAM zone is detected**
- **Poll temperature sensors using either:**
  - Direct access to sensor controller
  - OS API: WMI:MSAcpi_ThermalZoneTemperature
- **Temperature detection**
  - Analyze rate of drop
  - Analyze absolute temperature against threshold

BITARMOR

Defense # 4: A Virtual Secure Enclave for Storing and Using Keys*

- **Consider three attack vectors:**
  1. Booting alternate OS (remote or local), no RAM transfer
  2. Cooling RAM before power loss, RAM transfer
  3. **Cooling RAM immediately after power loss, RAM transfer**
- **Idea: Using OS, processor, and cryptographic techniques, efficiently create a secure enclave for exercising disk keys**
  - Technique would also defend against case where temperature sensing is thwarted
  - Princeton paper options cannot meet performance and key availability requirements of FDE systems

* Patent pending.

BITARMOR

FDE Key Management: Three Problems

- **Any data stored in memory may be available to attacker with relatively high fidelity**
  - With cooling, bit error rate might be extremely low (tens of errors over MBs of data)
- **Since encryption/decryption is needed for every disk I/O operation, keys must be perpetually available**
  - If keys require significant time to compute, performance may be adversely affected
- **If encryption/decryption is in progress, exposure of intermediate values may compromise key**
  - Full AES keys can be recovered from portions of AES round keys

BITARMOR

The Solution

- **Long-term disk data** is protected using a key that can be quickly derived from a huge number of DRAM bits
- **Short-term key data** is stored in plaintext only in processor registers
- **Spilling** of sensitive data is avoided by running at high interrupt priority

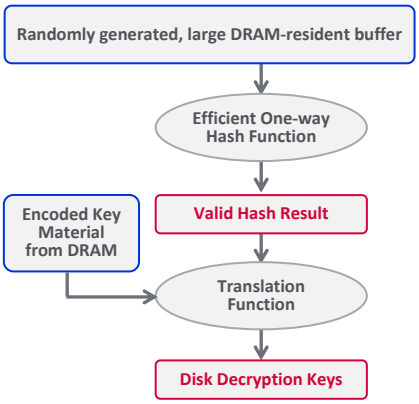
BITARMOR

The Solution

- **Long-term disk data is protected using a key that can be quickly derived from a huge number of DRAM bits**
- **Short-term key data** is stored in plaintext only in processor registers
- **Spilling** of sensitive data is avoided by running at high interrupt priority

BITARMOR

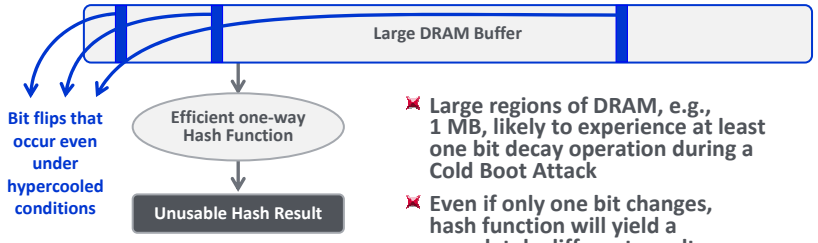
Encode Key Material Using Large Regions of DRAM



- **Indirectly distribute key material across large regions of DRAM (such as 1 MB)**
- **Reconstruct disk keys as needed in real time**
  - Use encoded key material loaded from DRAM and valid hash result
  - Perform one reconstruction per sector/page disk I/O
- **Hash Function and Translation Function MUST BE highly efficient**
  - SHA-256 method described in Cold Boot paper not fast enough for FDE applications
  - Use alternative that requires few instructions (e.g., under 25,000) per invocation

BITARMOR

Cold Boot Attacks Render Buffer Useless



- Large regions of DRAM, e.g., 1 MB, likely to experience at least one bit decay operation during a Cold Boot Attack
- Even if only one bit changes, hash function will yield a completely different result
- Mathematics of hash function will prevent an attacker from generating correct result even with global computing resources
- Size of buffer makes brute-forcing "all possible bit flips" infeasible even with very few flips

BITARMOR

Empirical Results on Memory Remanence

- **Detailed understanding of decay profile is critical**
  - For a defense like ours, time-to-first flip is more important than time-to-average flip
  - There has been no published systematic exploration of dependence of either of these values on temperature or other variables (bus speed, density, how long the value was held, etc.)
- **But, bit decay happens**
  - Order of decay is fairly deterministic, with some bits decaying quickly and others slowly
  - Majority of decay happens over a relatively short time period
  - Modern DRAM decays much faster than DRAM of 10 years ago (or even 3 years ago!)

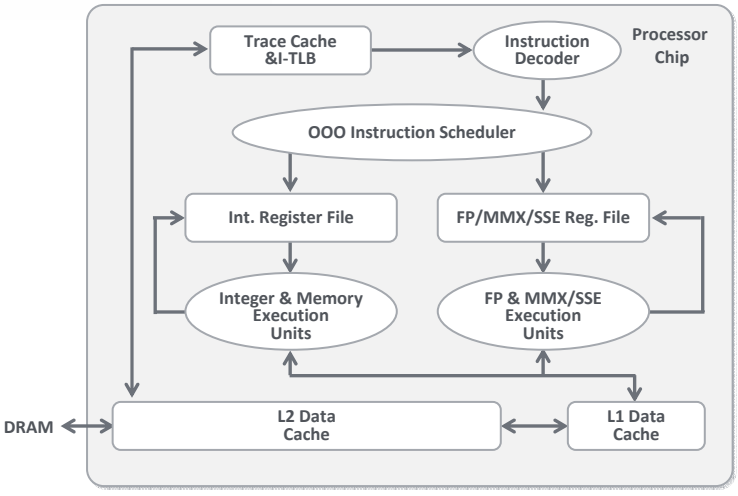
BITARMOR

The Solution

- **Long-term disk data** is protected using a key that can be quickly derived from a huge number of DRAM bits
- **Short-term key data is stored in plaintext only in processor registers**
- **Spilling** of sensitive data is avoided by running at high interrupt priority

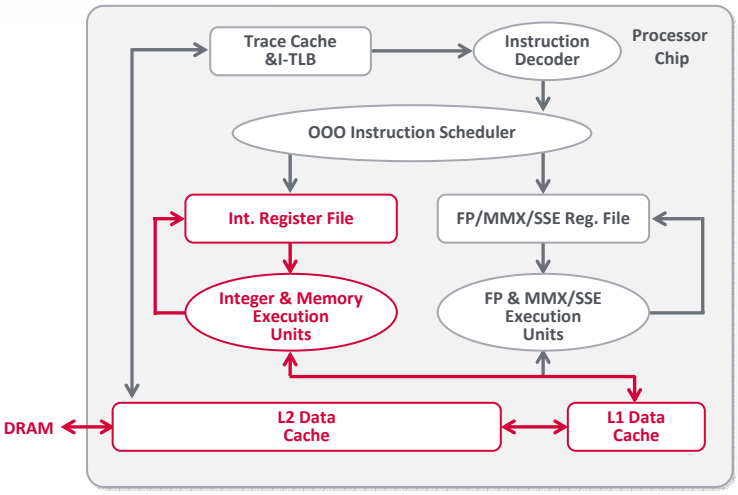
BITARMOR

Simplified Pentium Processor Internals



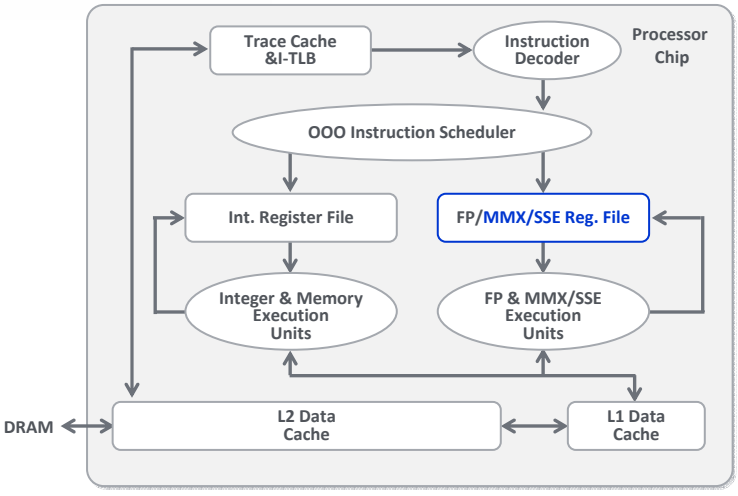
BITARMOR

Typical Flow Of Key Data



BITARMOR

We Can Use MMX and SSE for On-chip Storage



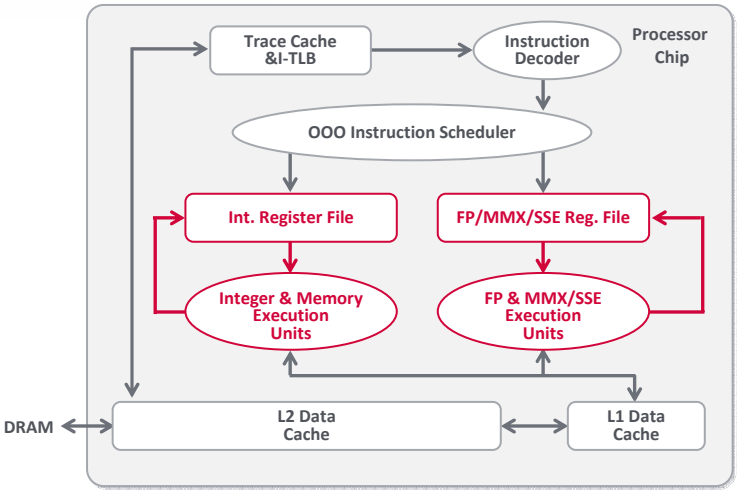
BITARMOR

Want to Avoid Having Key Material in Memory at *Any* Time

- **Use MMX and SSE registers to store key material**
  - Primarily intended for multimedia SIMD applications
  - Powerful but underutilized
- **Benefits**
  - Relatively volatile
  - Permits much faster encryption implementation
  - Can store entire expanded AES key schedule
- **Annoyance**
  - Instruction set is not exactly "general purpose"

BITARMOR

Protected Key Data Flow



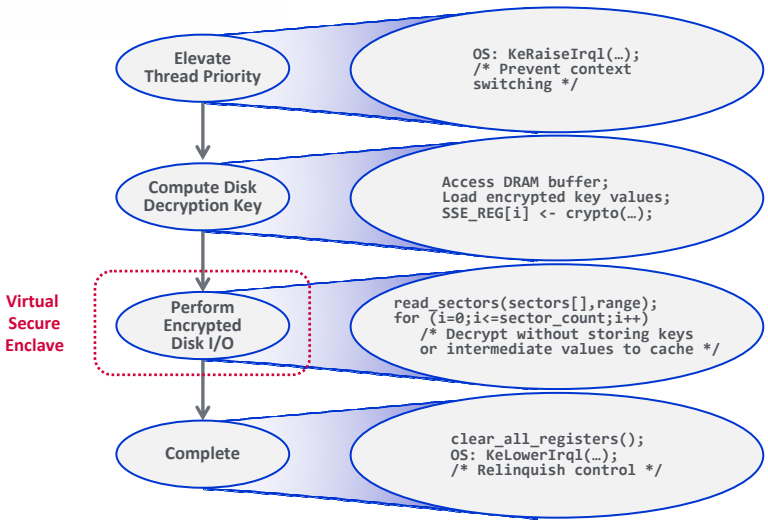
BITARMOR

The Solution

- **Long-term disk data** is protected using a key that can be quickly derived from a huge number of DRAM bits
- **Short-term key data** is stored in plaintext only in processor registers
- **Spilling of sensitive data is avoided by running at high interrupt priority**

BITARMOR

Protected Decryption



BITARMOR

Thoughts for the Future

- Cold Boot Attacks on encryption keys can be prevented with software solutions
- Both attacks and defenses can (and will) continue to evolve
- Need to start thinking about what sorts of architectural changes can be made to support secure computing in the future
- Availability of secure, long-term storage on CPUs would be a big win

BITARMOR

Thank you!

For more information, check out:

www.bitarmor.com/coldboot

Also, special technical thanks to

Jesse Twardus, Tim Shirley, and Ed Felten

BITARMOR

Why We're Here

- Founders and engineers from BitArmor, a software company that leverages encryption in unique ways
- Extensive research and development backgrounds in security and cryptography
- McGregor's association with Cold Boot: Collaborated with Prof. Ed Felten's research group while completing Ph.D. at Princeton; his research cited in Cold Boot paper