TECHNISCHE UNIVERSITÄT DRESDEN

THE INTEL MANAGEMENT ENGINE

Hauptseminar Technischer Datenschutz, TU Dresden

Dominik Pataky

Dresden, 2017-06-13

1 Introduction

What is the Intel ME?

- The Intel ME is a **hardware chip** inside modern Intel chipsets.

1 Introduction

What is the Intel ME?

- The Intel ME is a **hardware chip** inside modern Intel chipsets.
- Part of the vPro umbrella security and management product name: ME, AMT, VT-x, TXT, ...

1 Introduction

What is the Intel ME?

- The Intel ME is a **hardware chip** inside modern Intel chipsets.
- Part of the vPro umbrella security and management product name: ME, AMT, VT-x, TXT, ...
- The chip features multiple hardware components that enable a wide range of functionalities which are used by the software running on the ME

1 Introduction

What is the Intel ME?

- The Intel ME is a **hardware chip** inside modern Intel chipsets.
- Part of the vPro umbrella security and management product name: ME, AMT, VT-x, TXT, ...
- The chip features multiple hardware components that enable a wide range of functionalities which are used by the software running on the ME
- Works in all power states, as long as power is connected

What is the Intel ME?

- The Intel ME is a **hardware chip** inside modern Intel chipsets.
- Part of the vPro umbrella security and management product name: ME, AMT, VT-x, TXT, ...
- The chip features multiple hardware components that enable a wide range of functionalities which are used by the software running on the ME
- Works in all power states, as long as power is connected
- Software is full stack, from Real Time Operating System (RTOS) kernel up to Java Virtual Machine (JVM)

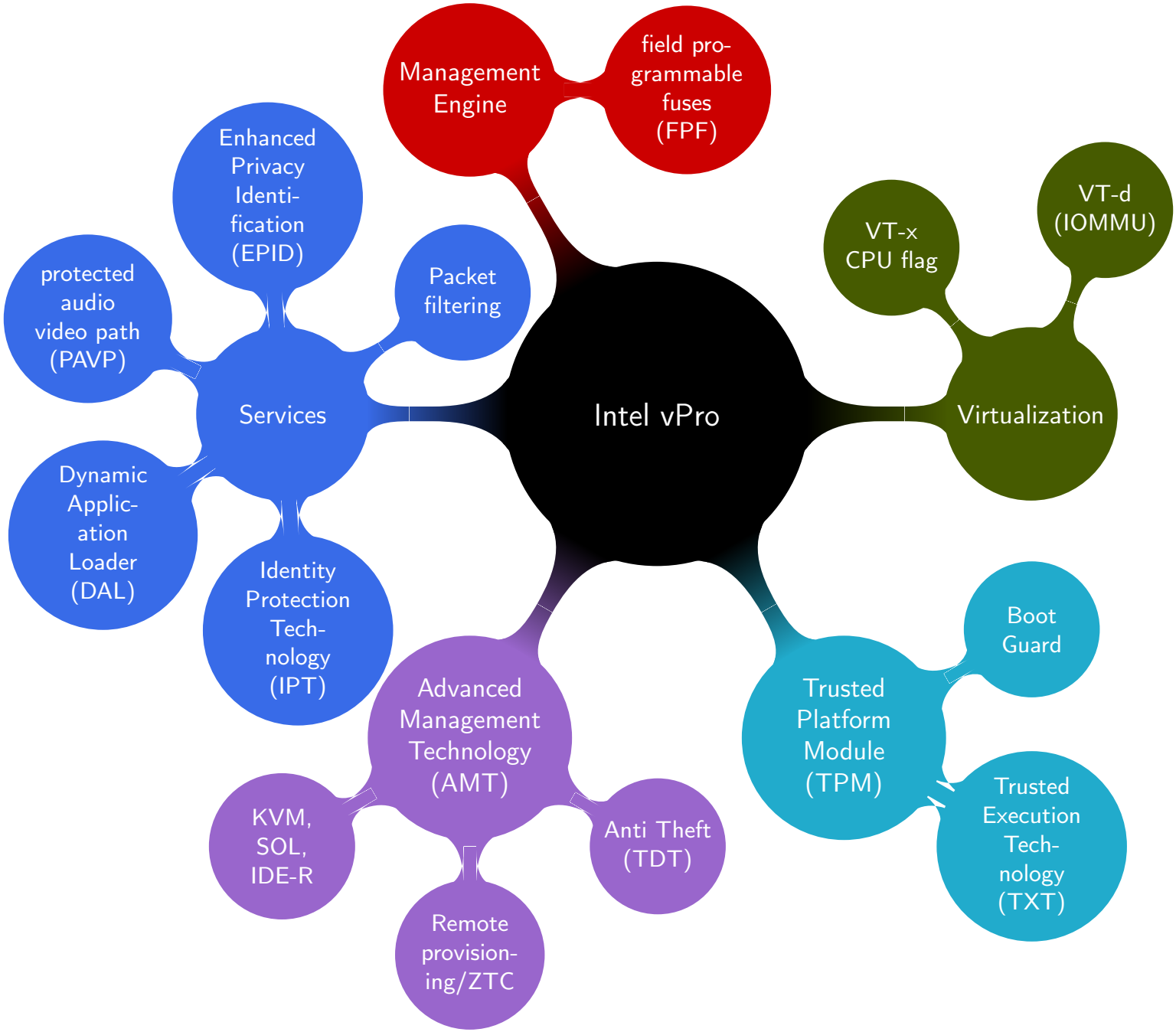

Figure 1: Intel vPro security and management products overview

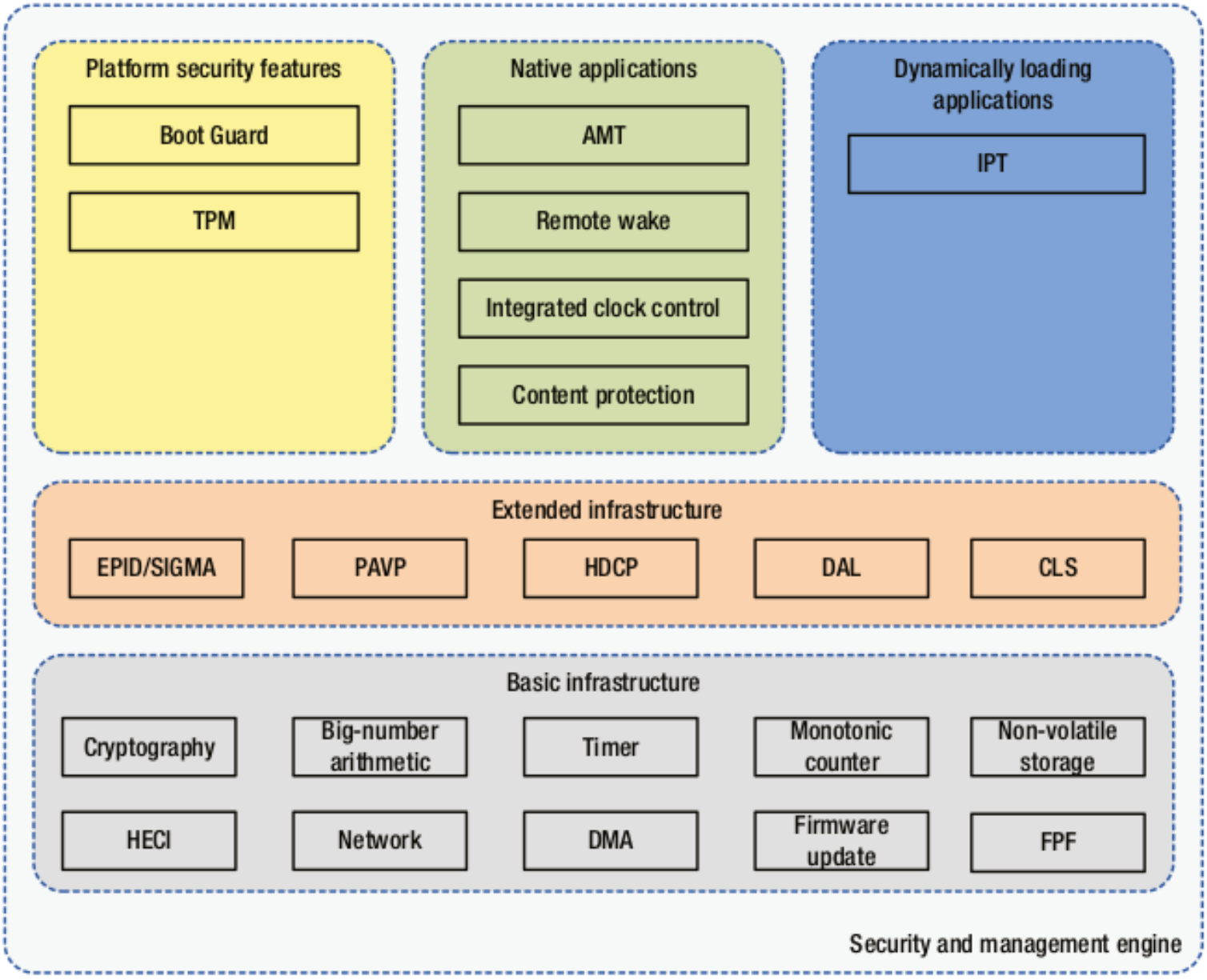

Figure 2: Intel ME components. Source: [Rua14]

Comparison with similar technologies

- **ARM** developed the „TrustZone“, which divides CPU execution contexts in secure and unsecure „worlds“

Comparison with similar technologies

- **ARM** developed the „TrustZone“, which divides CPU execution contexts in secure and unsecure „worlds“
- **AMD** uses the „Secure Processor“ for implementing the ARM TrustZone execution environment. Also mentions anti theft, which needs remote capabilities

Comparison with similar technologies

- **ARM** developed the „TrustZone“, which divides CPU execution contexts in secure and unsecure „worlds“
- **AMD** uses the „Secure Processor“ for implementing the ARM TrustZone execution environment. Also mentions anti theft, which needs remote capabilities
- **IPMI** (Intelligent Platform Management Interface) is an open standard with multiple implementations by different hardware manufacturers

2 Hardware

The chipset design after 2009

- Before 2009 the ME was integrated into the northbridge

2 Hardware

The chipset design after 2009

- Before 2009 the ME was integrated into the northbridge
- After 2009 all functionality of the ME was placed into the southbridge.

2 Hardware

The chipset design after 2009

- Before 2009 the ME was integrated into the northbridge
- After 2009 all functionality of the ME was placed into the southbridge.
- Has access to network controller, SPI flash, DRAM via DMA and the CPU via DMI

The chipset design after 2009

- Before 2009 the ME was integrated into the northbridge
- After 2009 all functionality of the ME was placed into the southbridge.
- Has access to network controller, SPI flash, DRAM via DMA and the CPU via DMI
- Note: the northbridge is the CPU called „(graphics and) memory controller hub“ (MCH/GMCH), the southbridge is named the „platform controller hub“ (PCH)

2 Hardware

Short history of Intel chipsets since the redesign in 2009

Series	Year	CPU code name	Core	ME (AMT)
5	2009	Nehalem	1^{st} gen	6.0-6.2
6	2011	Sandy Bridge	2^{nd} gen	7.0-7.1
7	2012	Ivy Bridge	3^{rd} gen	8.0-8.1
8	2013	Haswell	4^{th} gen	9.0-9.5
9	2014	Broadwell	5^{th} gen	10.0
100	2015	Skylake	6^{th} gen	11.0
200	2017	Kaby Lake	7^{th} gen	11.5-11.6

Chipset Series are represented as the first digit in the chipset model name, e.g. Z87 is Series 8, Q250 Series 200.

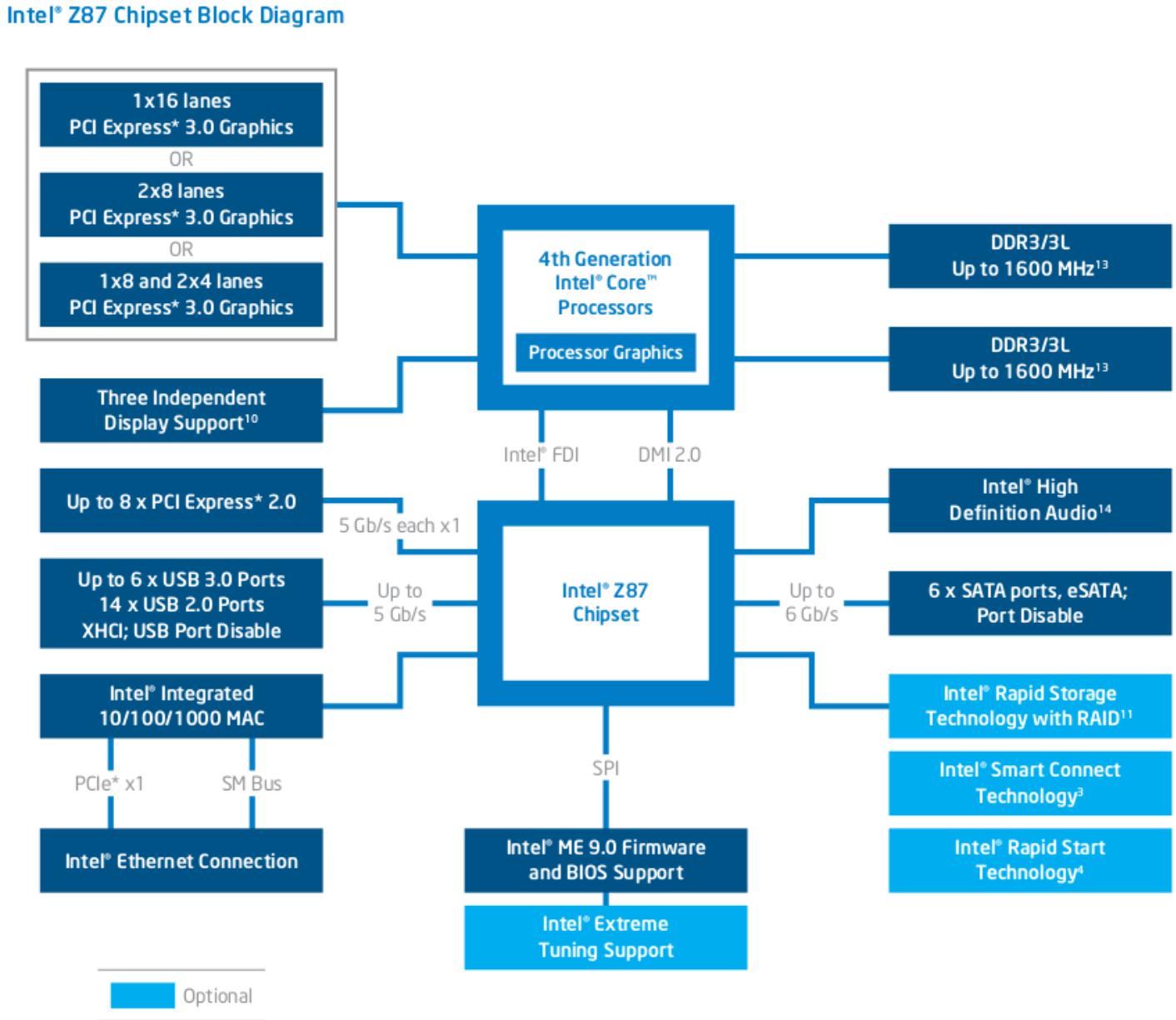

Figure 3: Intel Z87 (Series 8) chipset hardware. Source: `http://www.intel.com/content/www/us/en/chipsets/z87-chipset-brief.html`

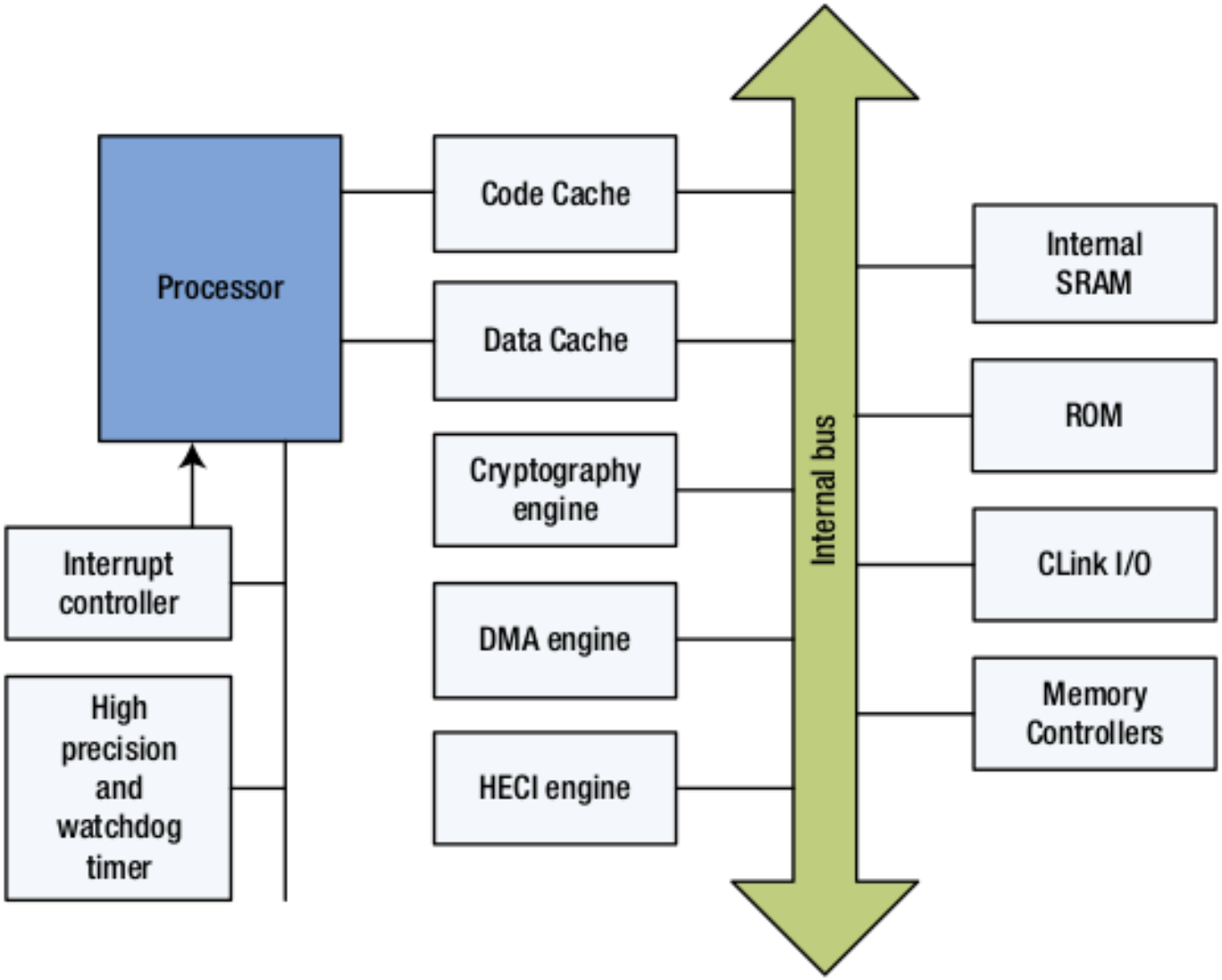

Figure 4: Hardware implemented components of the embedded ME chip inside the southbridge chipset. Source: [Rua14].

Firmware, modules and tasks

- The firmware is split into multiple modules. Stored in SPI flash together with BIOS

3 Software

Firmware, modules and tasks

- The firmware is split into multiple modules. Stored in SPI flash together with BIOS
- Modules are loaded as priviledged or nonpriviledged

3 Software

Firmware, modules and tasks

- The firmware is split into multiple modules. Stored in SPI flash together with BIOS
- Modules are loaded as priviledged or nonpriviledged
- Memory is organized in regions, enforced by memory manager. Access violations result in exception

3 Software

Firmware, modules and tasks

- The firmware is split into multiple modules. Stored in SPI flash together with BIOS
- Modules are loaded as priviledged or nonpriviledged
- Memory is organized in regions, enforced by memory manager. Access violations result in exception
- Running modules are called „tasks“ which are isolated from another. Access to resources („assets“) secured by kernel.

3 Software

Firmware, modules and tasks

- The firmware is split into multiple modules. Stored in SPI flash together with BIOS
- Modules are loaded as priviledged or nonpriviledged
- Memory is organized in regions, enforced by memory manager. Access violations result in exception
- Running modules are called „tasks“ which are isolated from another. Access to resources („assets“) secured by kernel.
- Exchange of data via kernel interface

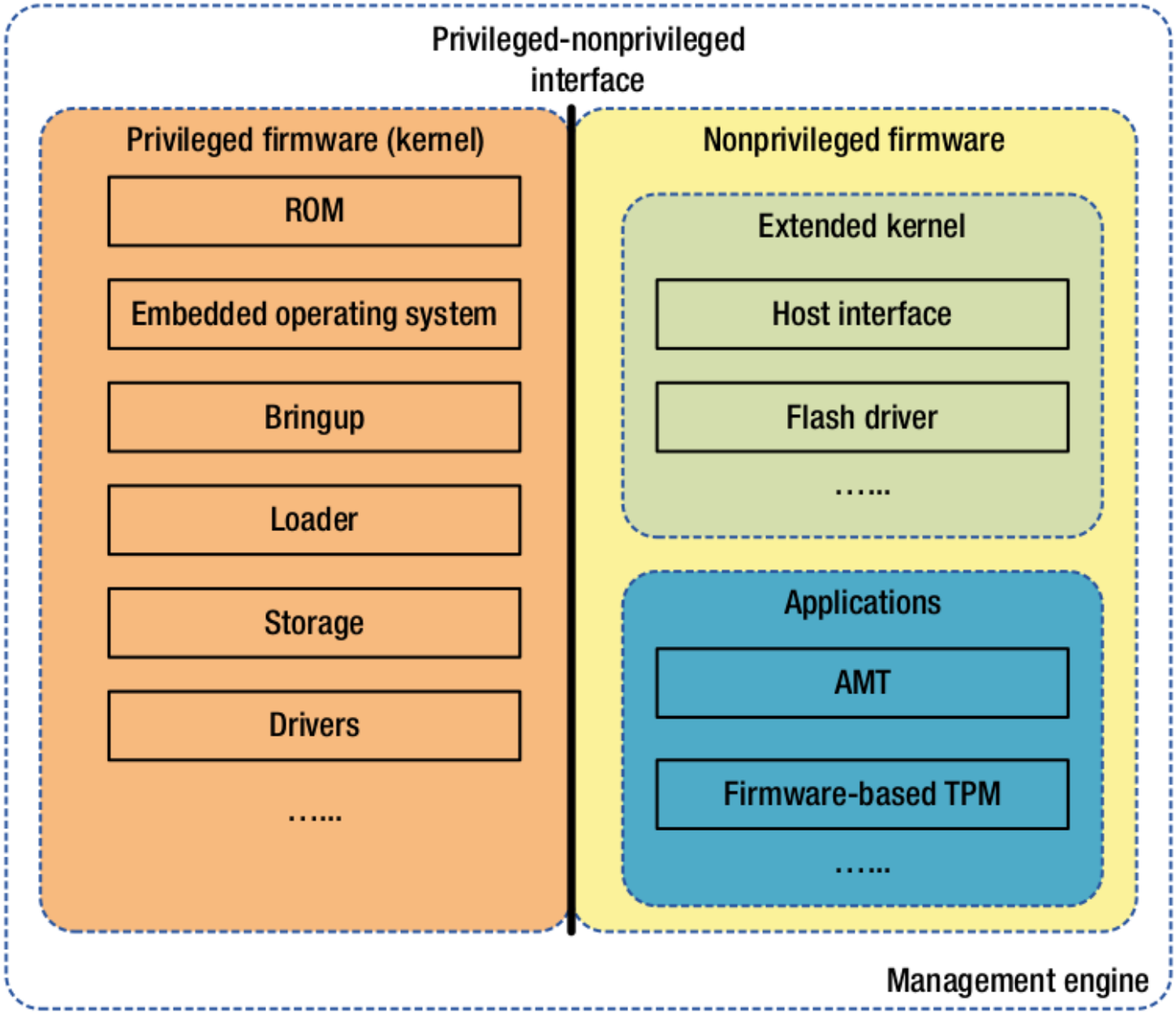

Figure 5: The Intel ME firmware components. Source: [Rua14, p. 35]

3 Software

Dynamic Application Loader (DAL)

- For applets not included inside the firmware

3 Software

Dynamic Application Loader (DAL)

- For applets not included inside the firmware
- DAL is a JVM with included library for ME functionalities (crypto, storage)

3 Software

Dynamic Application Loader (DAL)

- For applets not included inside the firmware
- DAL is a JVM with included library for ME functionalities (crypto, storage)
- Code is transfered from the OS to the DAL via the Host Embedded Controller Interface (HECI)

3 Software

Dynamic Application Loader (DAL)

- For applets not included inside the firmware
- DAL is a JVM with included library for ME functionalities (crypto, storage)
- Code is transfered from the OS to the DAL via the Host Embedded Controller Interface (HECI)
- Applets are also isolated from another

3 Software

Everything is signed

- The Intel signature public key is burnt into the ROM

3 Software

Everything is signed

- The Intel signature public key is burnt into the ROM
- All modules inside the firmware are individually signed, including applets in DAL

3 Software

Everything is signed

- The Intel signature public key is burnt into the ROM
- All modules inside the firmware are individually signed, including applets in DAL
- The ME uses TPM and similar routines to ensure a safe boot and firmware integrity

Known successful attacks against the ME

- „Ring -3“ rootkits by Invisible Things Lab. Targets memory remapping in Series 3 (2009)

Known successful attacks against the ME

- „Ring -3“ rootkits by Invisible Things Lab. Targets memory remapping in Series 3 (2009)
- „Zero touch“ enforced configuration by Vassilios Ververis. Targets unsecured deployment via DHCP in Series 4 (2010)

Known successful attacks against the ME

- „Ring -3“ rootkits by Invisible Things Lab. Targets memory remapping in Series 3 (2009)
- „Zero touch“ enforced configuration by Vassilios Ververis. Targets unsecured deployment via DHCP in Series 4 (2010)
- „Silent Bob is Silent“ by Maksim Malyutin from Embedi. Exploits broken HTTP digest authentication in AMT (May 2017)

Known successful attacks against the ME

- „Ring -3“ rootkits by Invisible Things Lab. Targets memory remapping in Series 3 (2009)
- „Zero touch“ enforced configuration by Vassilios Ververis. Targets unsecured deployment via DHCP in Series 4 (2010)
- „Silent Bob is Silent“ by Maksim Malyutin from Embedi. Exploits broken HTTP digest authentication in AMT (May 2017)
- Misuse of Serial over LAN (SOL) by the PLATINUM group. Redirects malware traffic on compromised host through ME, bypassing OS firewall (June 2017)

Further problems for security and privacy

- Patching the firmware is difficult for most end users, needs awareness of problem and know-how

Further problems for security and privacy

- Patching the firmware is difficult for most end users, needs awareness of problem and know-how
- Buying used hardware is a risk for privacy, AMT could still be enabled

Further problems for security and privacy

- Patching the firmware is difficult for most end users, needs awareness of problem and know-how
- Buying used hardware is a risk for privacy, AMT could still be enabled
- ME with AMT and other features could be misused by Intel, governments and spy agencies

Further problems for security and privacy

- Patching the firmware is difficult for most end users, needs awareness of problem and know-how
- Buying used hardware is a risk for privacy, AMT could still be enabled
- ME with AMT and other features could be misused by Intel, governments and spy agencies
- Key for signing could be stolen or broken, leading to spread of malware applications (DAL!)

Protection of systems with the ME

- Disabling the Management Engine is difficult. Software implementation depends on OEM with custom BIOS (MEBX), flaws could silently reactivate
- Patching BIOS and ME firmware regularly is important

Protection of systems with the ME

- Disabling the Management Engine is difficult. Software implementation depends on OEM with custom BIOS (MEBX), flaws could silently reactivate
- Patching BIOS and ME firmware regularly is important
- Hardware access even more difficult, needs SPI flash manipulation – but possible!

Protection of systems with the ME

- Disabling the Management Engine is difficult. Software implementation depends on OEM with custom BIOS (MEBX), flaws could silently reactivate
- Patching BIOS and ME firmware regularly is important
- Hardware access even more difficult, needs SPI flash manipulation – but possible!
- Filter ME related traffic before reaching the chip

Protection of systems with the ME

- Disabling the Management Engine is difficult. Software implementation depends on OEM with custom BIOS (MEBX), flaws could silently reactivate
- Patching BIOS and ME firmware regularly is important
- Hardware access even more difficult, needs SPI flash manipulation – but possible!
- Filter ME related traffic before reaching the chip
- Pressure Intel to change product design with laws, regulations and licences

Protection of systems with the ME

- Disabling the Management Engine is difficult. Software implementation depends on OEM with custom BIOS (MEBX), flaws could silently reactivate
- Patching BIOS and ME firmware regularly is important
- Hardware access even more difficult, needs SPI flash manipulation – but possible!
- Filter ME related traffic before reaching the chip
- Pressure Intel to change product design with laws, regulations and licences
- „Vote with your wallet“, use alternative CPUs: AMD, ARM, OpenCores

6 References and meta

Sources, references and additional information

[Rua14] Xiaoyu Ruan. *Platform Embedded Security Technology Revealed*. Apress, 2014. ISBN: 978-1-4302-6571-9.

While researching it's important to keep an eye on the investigated version, e.g. reverse engineering results by Skochinsky from 2012 are based on earlier versions. Ruan mentions in 2014, newer implementations were designed as a „reaction to attacks". Kaby Lake might introduce x86 microprocessor with Minix kernel and a whole new architecture.

These slides and the associated report with further references will be published on my website `https://dpataky.eu`