Intel’s new virtualization features on Xeon platforms

Yu Zhang @ Intel Corporation

LINUXCON + CONTAINERCON + CLOUDOPEN

China, 2018

Legal Disclaimer

No license (express or implied, by estoppel or otherwise) to any intellectual property rights is granted by this document.

Intel disclaims all express and implied warranties, including without limitation, the implied warranties of merchantability, fitness for a particular purpose, and non-infringement, as well as any warranty arising from course of performance, course of dealing, or usage in trade.

This document contains information on products, services and/or processes in development. All information provided here is subject to change without notice. Contact your Intel representative to obtain the latest forecast, schedule, specifications and roadmaps.

The products and services described may contain defects or errors known as errata which may cause deviations from published specifications. Current characterized errata are available on request.

Copies of documents which have an order number and are referenced in this document may be obtained by calling 1-800-548-4725 or by visiting www.intel.com/design/literature.htm.

Intel and the Intel logo are trademarks of Intel Corporation or its subsidiaries in the U.S. and/or other countries.

*Other names and brands may be claimed as the property of others

Intel’ future Xeon platforms

Intel’s future Xeon platform is empowering cloud computing with

- 5-level paging.
- New instructions i.e. new Intel® AVX-512 instructions.
- Virtualization enhancements to Intel® Processor Trace.
- User Mode Instruction Prevention(UMIP).
- EPT-based subpage permissions(SPP).

We need to support them in KVM. ☺

5-level paging

- Current architecture and motivation
- 5-level paging overview
- KVM status & next to do

5-level paging

Current 4-level paging mode in IA32-e

- Linear address(LA) space: 48-bit, 256TB;
- Physical address(PA) space: 46-bit at most, 64TB.

Industry trend

- In-Memory Databases (IMDB);
- Emerging memory technology, NVDIMM using Intel® 3D XPoint™.

OS requirements – 2 more linear address(LA) bits than physical address(PA) bits

- To divide the linear address space in half : user/kernel spaces;
- To provide a direct mapping in kernel linear space for whole physical memory.

5-level paging

Conclusion:

With PA width greater than 46 bit foreseeable, LA width greater than 48 bit is required, hence 5-level paging.

Note: a wider linear address width can also benefit for (K)ASLR.

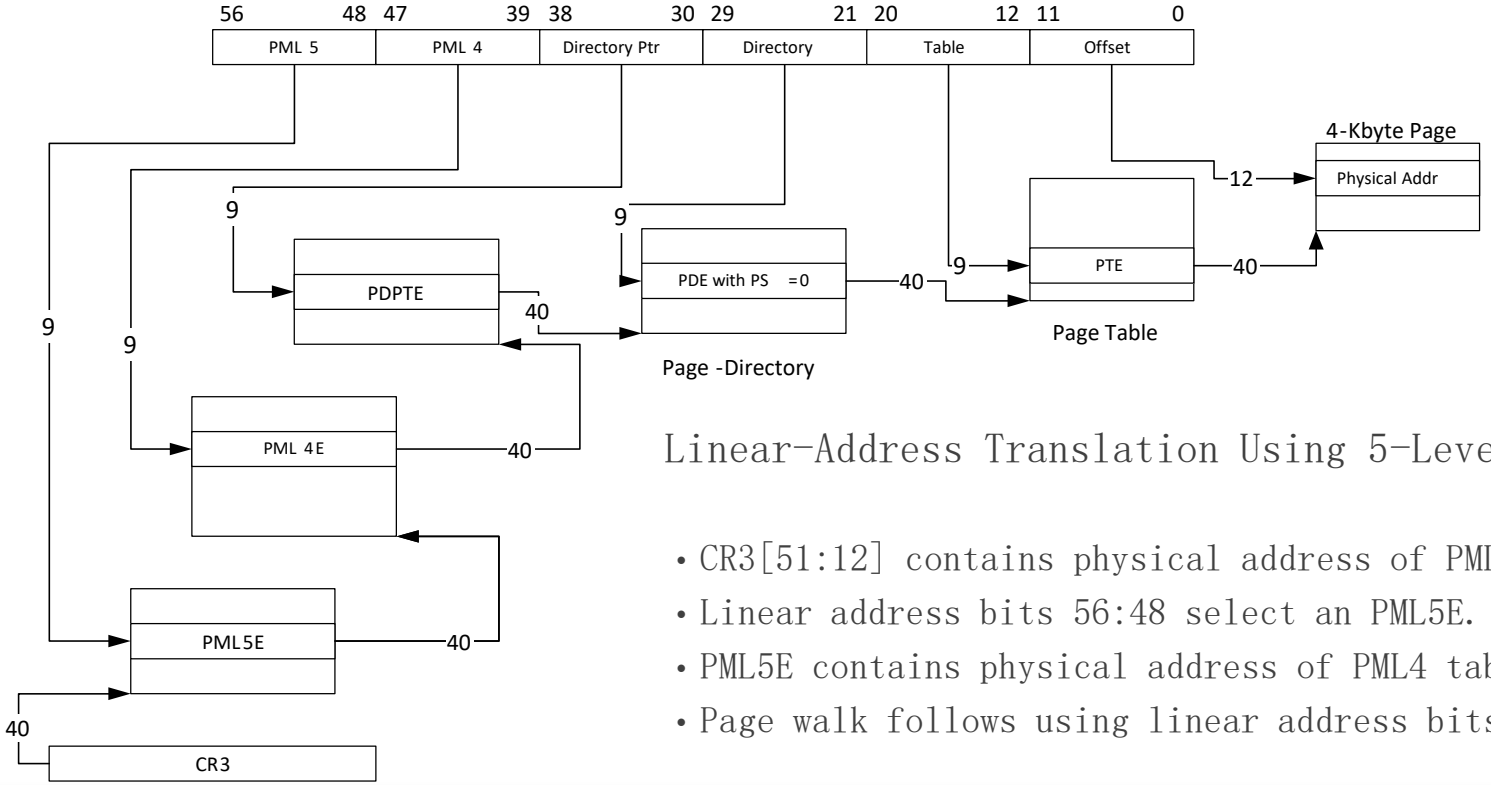
5-level paging

New paging mode in IA32-e: 5 level paging (AKA LA57).

Paging Mode	CR0.PG	CR4.PAE	IA32_EFER.LME	CR4.LA57	PA width(note)	LA width
IA32-e 4 level paging	1	1	1	0	Up to 46	48
IA32-e 5 level paging	**1**	**1**	**1**	**1**	**Up to 52**	**57**

- Note: PA(Physical Address) width is no greater than 46 on existing processors. And it can be extended to up to 52 on processors which have 5 level paging.

5-level paging



Linear-Address Translation Using 5-Level Paging

- CR3[51:12] contains physical address of PML5 table.
- Linear address bits 56:48 select an PML5E.
- PML5E contains physical address of PML4 table.
- Page walk follows using linear address bits 47:0.

5-level paging

5-level EPT:

- Provide 5-level EPT for VMs whose guest physical address width exceeds 48;

5-level IOMMU:

- For requests with PASID:
  - share 5-level CPU page table for first level translation.
- For requests without PASID or second level translation of requests with PASID:
  - provide 5-level IOMMU translation table.

5-level paging

5-level paging support in KVM:

- Expose 5-level paging feature to KVM guests.
- 5-level EPT support.
- 5-level shadow mode support.
- Other extensions, e.g. guest page table walk, linear & physical address validation.
- Patches merged in Linux 4.14.

Next to do

- Virtual IOMMU support for 5-level paging in Qemu.

New Intel® AVX-512 instructions

Intel® AVX-512: 512 bit Advanced Vector Extension SMID instructions

- Accelerate performance for workloads such as scientific simulations, deep learning, cryptography and encryption etc.
- Width of the SIMD register file extended to 512 bits.
- Feature detection with cpuid.

New Intel® AVX-512 instructions

New instructions to the Intel® AVX512 family on future platforms

- AVX512 VNNI: vector instructions for deep learning.
- AVX512 GFNI: GFNI algorithms used for cryptography/encryption areas.
- Others, e.g. VBMI2(vector byte manipulation) etc.

Intel® AVX512 support in KVM:

- Simple, just expose features with cpuid emulation for KVM guests;
- Merged in Linux 4.16.

Intel® Processor Trace

- Hardware feature that logs software execution information.

- Supports control flow tracing:
  - Decoder can process the captured trace data and reconstruct the exact program execution flow.
  - Therefore can also enhance control flow integrity.

- Can generate timing, and bookkeeping information that enables both functional and performance tuning of applications.

Intel® Processor Trace

Intel® Processor Trace packets:

- Control flow packets, e.g. branch taken/not taken, target/source IP addresses etc.
- Timing packets, e.g. TSC, MTC(minimal time counter) etc.
- Paging information packets.
- Others, e.g. VMCS packets, power management packets etc.

Enabling and configuration:

- IA32_RTIT_* MSRs.
- Can be filtered based on context(CR3), IP ranges, CPL(kernel/user mode).

Intel® Processor Trace

Trace Output:

- A single, contiguous region of physical address space in DRAM.
- A collection of variable-sized regions of physical memory referenced by ToPA(Table of Physical Addresses).
- A MMIO debug port, in order to reroute to a platform-specific trace endpoint.

Intel® Processor Trace VMX extension:

- New guest IA32_RTIT_CTL value field to the VMCS.
- Enabling use of EPT to redirect PT output.

Intel Processor Trace

Intel® Processor Trace VMX support in KVM:

- Expose Intel PT to KVM guests through cpuid emulation.
- Support system mode to track host and guest information in unified buffer.
- Support host/guest mode to track host and guest information separately.
- Patches under review: https://lkml.org/lkml/2018/5/21/1186

User-Mode Instruction Prevention (UMIP)

User-Mode Instruction Prevention (UMIP)

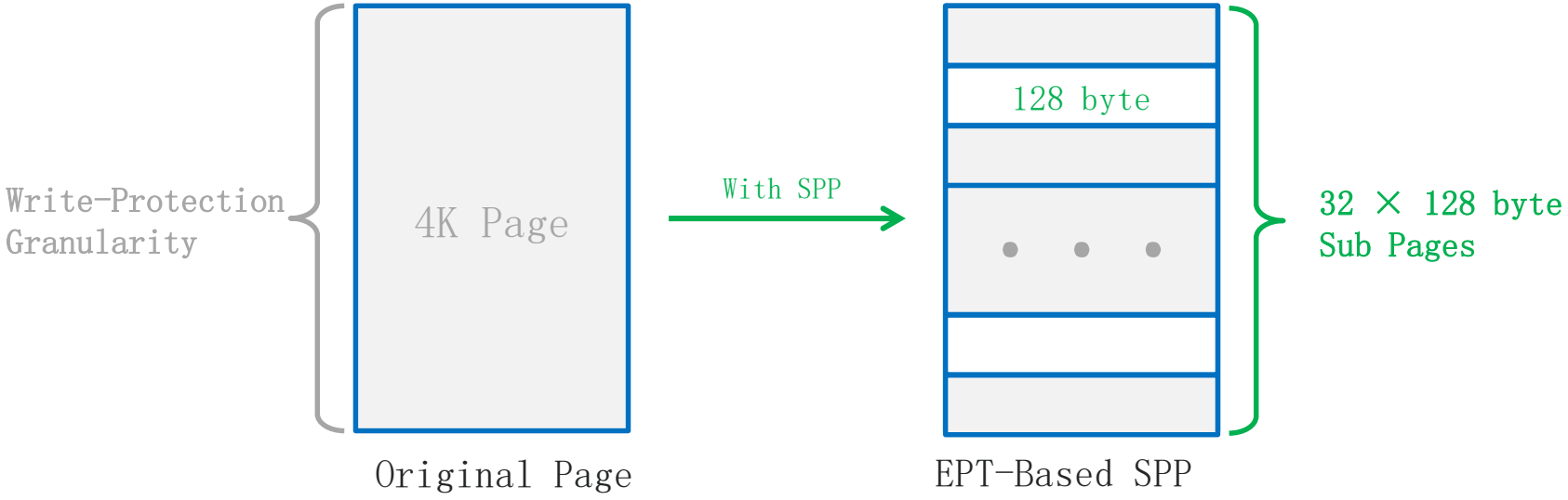
- Protect system-wide settings from exposure to user space:
  - SGDT/SIDT/SLDT/STR/SMSW prevented from execution with CPL > 0.
- Reduces the tools available to craft privilege escalation attacks e.g. CVE-2013-2094.

UMIP support in KVM

- Expose UMIP to KVM guests through cpuid emulation.
- Emulate UMIP on legacy platforms:
  - through descriptor table exiting;
  - with exception to SMSW.
- KVM patch merged in Linux 4.16(with a bugfix in 4.17-rc6).

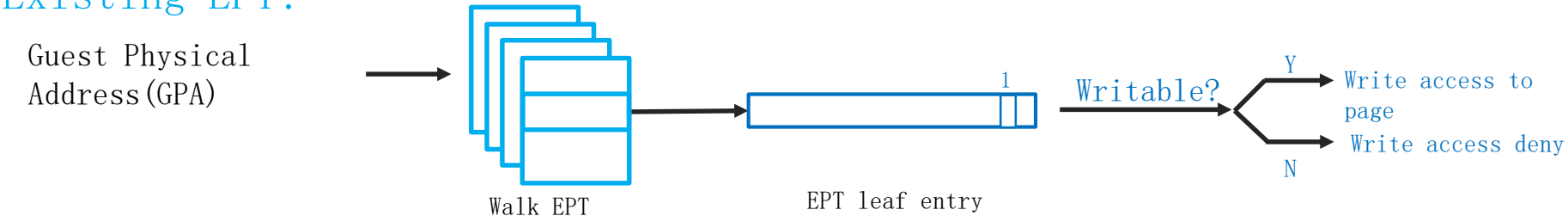
EPT-based Subpage permissions

- SPP(EPT-based subpage permissions)
  - Allows host to specify write-permission for guest physical memory at a sub-page (128byte) granularity.

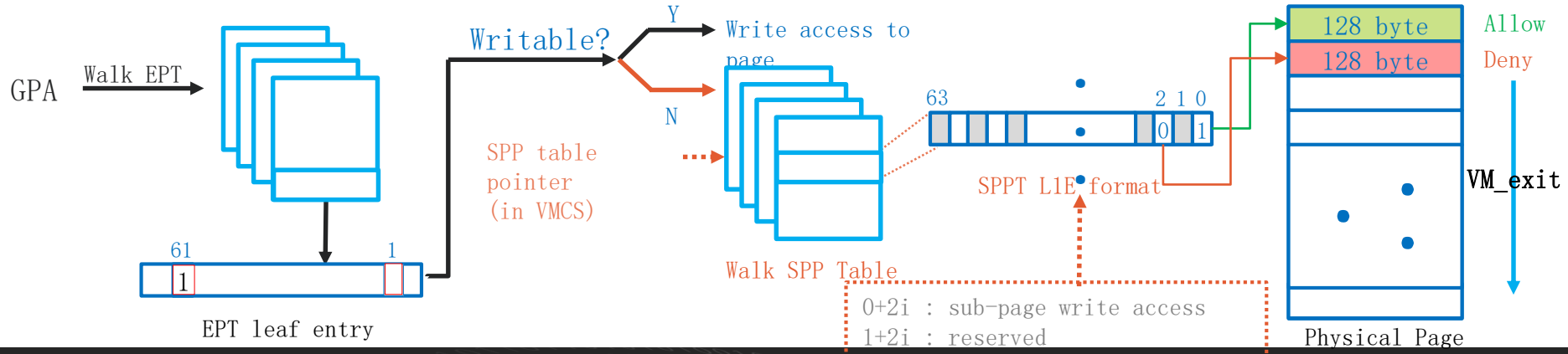


EPT-based Subpage permissions

Existing EPT:



EPT-Based SPP



EPT-based Subpage permissions

SPP support in KVM:

- New ioctl interfaces to set & get subpage permission settings.
- Creation of SPP structures and update to EPT leaf entries based on the subpage permission setting.
- Handling of SPP induced VM-Exits.
- RFC Patchset sent out: https://lkml.org/lkml/2017/10/13/475.

Summary

- Intel’s upcoming platform is empowering cloud computing with features to support huge memory, enhanced performance tuning and security guarding capabilities.

- KVM will have all these features enabled.

Q & A