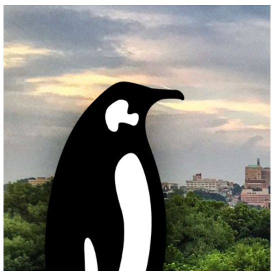

Contribution ID: **256** Type: **not specified**

IOMMU overhead optimizations and observability

Wednesday 15 November 2023 11:30 (45 minutes)

IOMMU overhead memory, which is primarily page table memory, is allocated directly from the buddy allocator, and is not charged or accounted for. Also, there is no easy way to debug IOMMU translations as there are no user interfaces that allow walking through IOMMU page tables. Below are the proposals to solve the problems.

Add an observability for IOMMU page table memory into /proc/meminfo:

```
PageTables:        XXX kB
SecPageTables:     XXX kB
IOMMUPageTables:   XXX kB
```

This would allow users to see how much IOMMU page table memory is
being used, which could help them identify and troubleshoot performance problems.

**Charge the IOMMU page table memory to the proper owner when DMA
mappings are established:**

This would allow users to control and limit the amount of IOMMU page table memory that is used by each process.

Allow walking through IOMMU page tables on live systems and in kdumps:

This would allow users to debug IOMMU translations and identify problems.

For live systems the interface should be similar to /proc/PID/pagemap, so users could walk through IOMMU page tables, and study which physical pages are currently mapped into page tables.

For kdumps, it should be a crash-utility extension to dump IOMMU page tables.

Limit the growth of page tables:

Currently, when pages are removed unmapped from the page table, the
free page table levels are not returned back to the system, see [1]
for example. This can cause substantial overheads in cases where VA
addresses are not recycled. On the other hand, recycling VA addresses in order to save memory can be a security risk, and in general a bad practice.

We propose to limit the maximum number of empty page table levels to a certain amount.

Add iova_stress[1] into kernel selftest:

This would allow us to verify that page table overhead does not
regress in the future.

[1] https://github.com/soleen/iova_stress

Primary authors: TATASHIN, Pasha; ZHAO, Yu (Google)

Presenters: TATASHIN, Pasha; ZHAO, Yu (Google)

Session Classification: VFIO/IOMMU/PCI MC

Track Classification: LPC Microconference: VFIO/IOMMU/PCI MC