Contribution ID: **65** Type: **not specified**

The Case for Memory Segregation

Wednesday 22 September 2021 08:00 (45 minutes)

In this talk, we would like to propose adding roles to memory pages. We contend that the current monochromatic memory model cannot address modern systems' security and performance needs.

We want to discuss two recent projects that perform memory segregation. DMA Aware Malloc for Networking (DAMN) that protects against DMA attacks (e.g., project thundeclap) while providing the same performance at +100Gb/s as with iommu=off.
And also, a Memory Allocator for I/O (MAIO), which facilitates overhead free zero-copy networking for user-space applications.

We implement memory segregation by adding extra metadata into Tail pages of huge (i.e., compound) pages. This additional metadata allows for
fast address translation and any additional operations per segment role.
The shared common DNA of both projects is a memory allocator that allots memory for specific operations. This memory can later be reclaimed by a simple put_page. While the memory pools are based on compound pages, different memory allocation techniques can be implemented, e.g., page_frag or single 4K page allocations. We take special care to ensure that the individual page ref_count is used by get/put_page used rather than the head pages ref_count.

This form of memory segmentation isolates general kernel memory from segmented memory that a device or a user also uses. This isolation of memory is vital to facilitate fast and secure I/O operations.

Both DAMN and MAIO are prime examples that demonstrate how to solve complex I/O problems by adding segmentation to the existing memory model.

I agree to abide by the anti-harassment policy

I agree

Primary author: MARKUZE, Alex (VMware)

Presenter: MARKUZE, Alex (VMware)

Session Classification: LPC Refereed Track

Track Classification: LPC Refereed Track (Closed)