ORACLE

Poison & Remedy of VMAs

Instead of Guard VMAs

Lorenzo Stoakes

Liam R. Howlett

Linux Kernel Developers
20th September, 2024

Reason & Status

1. Reason
   - Wasted memory
   - Wasted time
2. Status
   - Prototype implemented and tested
   - Series with full test suite **complete!** (last week), will upstream RFC soon
   - PTE poison marker set/removed by madvise(MADV_GUARD_POISON / MADV_GUARD_REMEDY)
   - Can use vector process_madvise()
3. Testing of Prototype
   - 5x faster than mmap()-ing guards
   - 13% fewer VMAs on idle Android system (optimisations and load likely to be far better)

Open Questions

1. Accounting VMAs
   - Guards are no longer counted in mmap->map_count
   - But no extra resources are used, however if no anon_vma, we must prepare one for fork to copy page tables
2. Userspace cannot see the guards
   - Is this really an issue?
   - Is a change in **fault** behaviour, not VMA behaviour. Poison PTEs are **non-present**
   - When remedied, behaviour of poisoned ranges returns to normal
3. SIGSEGV or SIGBUS?
4. Restrictions
   - Anon only
   - No hugetlb, no ‘special’ VMAs
   - No mlock()’d pages

ORACLE