Contribution ID: **220** Type: **not specified**

PCSC: Caching PCI Config Space Accesses for faster Live Updates

Saturday 13 December 2025 17:45 (15 minutes)

At scale, virtualization uncovers hidden bottlenecks, including the cost of PCI configuration space accesses. In SR-IOV deployments with thousands of VFs, each configuration read triggers a hardware transaction. As VFs increase, these accesses scale linearly, leading to longer VM boot times, heavier bus contention, and noticeable startup delays.

The PCI subsystem today treats every configuration space read as a live hardware operation, even when accessing registers that are effectively static for the lifetime of the device. Vendor and device IDs, many capability registers, and control fields are read again and again during enumeration and driver initialization, despite rarely changing. This inefficiency becomes pronounced in high-density virtualization workloads.

This presentation discusses the **PCI Configuration Space Cache** (**PCSC**), a new caching layer that transparently accelerates configuration space operations without requiring driver changes. PCSC attaches a per-device cache to struct pci_dev, intercepts reads and writes through custom PCI ops, and enforces coherency with a write-invalidate policy. This approach allows even registers involved in operations such as BAR sizing to be cached safely, maximizing coverage while preserving correctness. Cacheability is inferred dynamically by walking each device's capability chains in a specification-aware way, avoiding static lists and hard-coding.

The implementation goes beyond caching alone by enabling persistence across kexec transitions using the Kernel HandOver (KHO) framework. Cache contents are preserved using KHO, allowing the new kernel to reuse configuration data instead of re-probing hardware.

Testing on ARM platforms with typical PCI Devices shows cache hit rates near 70% in virtualization scenarios, with substantial improvements in VM bring-up time. The discussion will explore design trade-offs, path to merging upstream and integration with the Live Update Orchestrator.

Primary author: PETRONGONAS, Evangelos (Amazon Web Services)

Co-author: GRAF, Alexander

Presenter: PETRONGONAS, Evangelos (Amazon Web Services)

Session Classification: Live Update MC

Track Classification: Live Update MC