Paravisor Integration with Confidential Services

Jon Lange, John Starks, Microsoft

Linux Plumbers Conference 2025

Paravisor vs SVSM

- Paravisor takes full responsibility for all confidential virtualization services
  - Virtualization intercepts (MSR access, CPUID, etc.) provided through architectural primitives (e.g. vTPM device accesses)
  - Confidential memory management (shared vs. private page control)
  - Support for legacy guests without awareness of confidential computing
- SVSM leaves confidential service management in the hands of the guest OS
  - Guest is fully enlightened to requirements of confidential computing
  - Guest interacts with hypervisor directly according to platform rules
  - SVSM is voluntarily invoked when services are required (vTPM etc.)

Confidential Services

- Separable, optional confidential services
  - Guest-driven attestation
  - TDISP device binding policy
- Desire for confidential services and legacy virtualization management are not mutually exclusive

Problem Statement

- How can a guest OS be made aware that it is operating in a confidential environment, with confidential services available, without requiring it to abandon paravisor interaction and take full responsibility for confidential computing responsibilities
- The nature of SEV-SNP and TDX Partitioning mean the SVSM and paravisor models may not behave equally on both platforms because the relationship between different VM privilege levels is fundamentally different, but the guest OS need for confidential serices is the same. How can confidential services be offered similarly on all platforms? (See also Stefano's talk)