

Questions about "Scale and Performance in the Denali Isolation Kernel"

Introduction

- What is meant by scale?
- What is an isolation kernel? (how is it different/same than a VMM?)

Case for IKs

- What is the key motivation for IKs?
Looking back, in what ways were the authors right/wrong?
- Why not just use an OS? (w/ processes)
- Why are "Zipf workloads" relevant?

Denali Design

- How is the ISA different/similar to x86?
Why is this important?
- New instructions: What are they, why needed?
- Memory architecture: Describe
How is this similar to/different than typical VMM?
- I/O devices: what are virtual interrupts?
(how different from physical interrupts?)

Denali Implementation

- Two policies for CPU: gatekeeper and scheduler; what are these, why are they needed?
- Memory management: Static swap region per VM: why needed?
- When are page tables allocated/used?
- 16MB per VM in this paper: what is up with that?
- I/O devices: VMs can only process packets during their quantum; why is this important?

Ilwaco Library OS

- What is a "library OS"
- What are pros/cons of this approach?
- Is there a file system?

Evaluation

- What is good/bad about the experimental setup?
- How many machines are used to drive load?
- Could you repeat these experiments (assuming access to source code/etc.)?
- Figure 3: "Denali's app-level performance closely tracks that of BSD": true? Explain.
- Figure 4: What to learn here?
Why does performance drop notably towards the right side of the graph?
- Figure 5: What is the point of this graph?
Do the numbers match what is seen in Figure 4? (should they?)
- Figure 6: What is this comparing? Is it a fair comparison?
- Table 2: Why is the memory cost/VM important?
- What is the "mbuf entropy" problem? How addressed?
- Figure 8: When does drop off start? Why at that point?
- Figure 9: Why is there a difference in 4k/16 line towards left of graph?
- Is there anything to learn from Figures 10 and 11?
- Does Quake experiment teach us anything?