Systems and Network Qualifier Spring 2011

Answers due by midnight Feb 4nd 2011 (due by midnight Friday)

1. The following questions relate to the papers: "Guest-Transparent Prevention of Kernel Rootkits with VMM-based Memory Shadowing" by Riley et al and the paper "The Turtles Project: Design and Implementation of Nested Virtualization" by Ben-Yehuda et al.

   a) Express, in your own words, the key idea of the rootkit prevention paper and provide a brief explanation of its underlying rationale! Do not merely paraphrase the abstract!

   b) Express, in your own words, the key idea of the Turtles project paper and provide a brief explanation of its rationale! Do not merely paraphrase the abstract!

   c) Riley et al state in Section 2.2: "*NICKLE is guided by the principle of kernel code guarding, but does so differently from the brute-force approach of tracking/checking kernel code address ranges – mainly for performance reasons. More specifically, the brute-force approach could store the address ranges of valid kernel code in a data structure (e.g., tree) with O(log N) search time.*" and later "*We point out that the interception of VM memory accesses can be provided by existing VMMs (e.g., QEMU+KQEMU, VirtualBox, and VMware).*"

      Evaluate these statements! Under what circumstances can provide VMMs interception of VM memory accesses in the way the authors describe? Could Riley's technique be used in the Turtle system?

   d) In your opinion, do the benefits that can be reaped from having nested virtualization (as provided by Turtles) justify the implementation complexity?

2. The following questions relate to the papers: "Efficient System-Enforced Deterministic Parallelism" by Aviram et al. and "Large-scale Incremental Processing Using Distributed Transactions and Notifications" by Peng & Dabek.

   a) Express, in your own words, the key idea of the Determinator paper and provide a brief explanation of its underlying rationale! Do not merely paraphrase the abstract!

   b) Express, in your own words, the key idea of the Percolator paper and provide a brief explanation of its underlying rationale! Do not merely paraphrase the abstract!

   c) Would Dekker's algorithm work in Determinator? Justify your opinion!

d) Would it be possible to implement Unix signals in Determinator? If so, describe how!

e) Percolator uses a multi-threaded approach. Based on the description provided, could its functionality be implemented using Determinator's model of parallelism?

f) The authors of Determinator state that "Since the multicore revolution is young and most application code is yet to be parallelized, we may still have a choice of what synchronization abstractions to use." --- yet, Determinator does not support widely used non-deterministic abstractions such as semaphores. Discuss whether Determinator's abstractions are, in your opinion, sufficient for all emerging multicore code!

3. The following questions relate to the papers: "Addressing Shared Resource Contention in Multicore Processors via Scheduling" & "Operating System Support for Overlapping-ISA Heterogeneous Multi-core Architectures"

    a. Express, in your own words, the key idea of *each* paper and provide a brief explanation of the underlying rationale for *each* paper.

    b. With respect to the paper on addressing shared resource contention in multicore processors via scheduling, the authors conclude that "the highest impact of contention-aware scheduling techniques is not in improving performance of a workload as a whole but in improving quality of service or *performance isolation for individual applications.*" Does this imply that the memory associated with individual cores in a multicore processor should become increasingly isolated? Why? Why not?

    c. In analyzing and synthesizing the information gleaned from the two aforementioned papers, how do you envision systems software to evolve with respect to future multicore (and manycore GPU) systems?

4. The following questions relate to the following papers: "Servet: A Benchmark Suite for Autotuning on Multicore Clusters" & "On the Limits of GPU Acceleration"

    a) Express, in your own words, the key idea of *each* paper and provide a brief explanation of the underlying rationale for *each* paper.

    b) The Servet paper claims that detecting the cache hierarchy, memory access bandwidth, and memory is sufficient to autotune codes to improve their performance in multicore systems.

a. Do you agree or disagree with the above? Explain your rationale.

b. How would you use the above information to tune the performance of a code on a GPU?

c) Discuss the effectiveness of the Servet benchmark suite on emerging multicore processors, e.g., AMD Magny Cours and Intel Westmere, keeping in mind that the benchmarked processors were the Intel Dunnington (Xeon E7450) hexacore processor and the Intel Itanium2 Montvale dual-core processor and keeping in mind the other papers that you read for the *Systems and Networking Ph.D. Qualifier Exam*.

d) Despite the high profile of the "On the Limits of GPU Acceleration" paper, discuss the technical weaknesses of that paper.

e) With these two papers as a backdrop, what would you propose to do as a "next step" research-wise? Be specific.

5. The following questions relate to the paper "The implementation of the Cilk-5 multithreaded language"

   a. Describe the core ideas behind the Cilk programming language.

   b. Describe in your own words how Cilk's work reduction strategy using *fast* and *slow* clones aims to reduce overhead and increase parallelism.

6. The following question relates to the following papers: "Distributed communications via global buffer" and "Ease of Use with Concurrent Collections (CnC)"

   a. In your own words, compare and contrast the programming models described in these papers. In particular, pay attention to the models for concurrent execution and data flow.