Daniel S. Upton

Present Address:
300 Montebello Circle #6C
Charlottesville, VA 22903
Cell: (434) 284-2302

Permanent Address:
HC 77 Box 296
Ballard, WV 24918
(304) 466-1640
Email: upton.dan@gmail.com

Objective

To obtain a summer internship in industry that leverages and enhances my background in compilers, virtual machines, architecture, and operating systems.

Education

August 2005 – Current
Computer Science Masters/PhD program
University of Virginia, Charlottesville, VA 22904

August 2001 – May 2005
Bachelor of Science, Computer Science
University of Richmond, Richmond, VA 23173

Graduate Courses

Architecture, Optimizing Compilers, Algorithms, Theory of Computation, Distributed Systems, Virtual Machines, Multicore Architecture

Relevant Undergraduate Courses

Operating Systems, Database Systems, Computer Graphics, Software Engineering, Programming Languages, Parallel Programming

Current Research

Studying cooperation between virtual execution environments and operating systems, allowing one layer to access and exploit information available in the other layer. Examples include the VEE using information about virtual memory usage and page faults monitored by the OS, or the OS using profiling information from the VEE to guide scheduling.

Publications

"Heterogeneous Chip Multiprocessor Design for Virtual Machines." Dan Upton and Kim Hazelwood. In *Second Workshop on Software Tools for Multicore Systems (STMCS'07)*.

Projects

Masters Project – presented Fall 2007

- "Design of a Custom VEE Core in a Chip Multiprocessor." Chip multiprocessors have become commonplace, but the best design still remains open for exploration. In this work, I profiled Pin, a virtual execution

environment (VEE) for dynamic instrumentation, at the microarchitectural level to guide design of a VEE-specific in a chip multiprocessor.

Multicore Architecture – Fall 2006, Spring 2007

- Implementation toward a multicore simulation framework. The class proposed a new heterogeneous multicore architecture consisting of one or more control processors for executing sequential portions of applications, alongside a large grid of small processors for parallel execution. I was responsible for developing a PinTool to allow us to use a physical processor as the control processor, including simulating new instructions to initialize and control the simulated grid. I also worked on designing and implementing cache policies for the proposed architecture.

Virtual Machines – Spring 2006

- "Detection and Subversion of Virtual Machines." Modern virtualization systems can be used for enforcing security policies, but increase the size of the trusted computing base. If a malicious program can detect the presence of a virtual environment, the malicious program may be able to exploit that or alter its behavior to go unnoticed. I explored methods of detecting and attacking process-level virtual machines such as DynamoRIO, HDTrans, and Pin.

Distributed Systems – Spring 2006

- Implementation toward Distributed CVS software. Repositories such as CVS provide central way to manage resources such as code and documentation, but also create a central point of failure. In this projuect, we designed and began implementation of a peer-to-peer-based distributed CVS model. Multiple peer servers can service any file request, and the peer-to-peer engine would automatically resolve conflicts whenever possible if one server's state changed independently of other servers.

Undergraduate Research Assistant – Summer 2004

- Machine learning applied to distributed computation security. Distributed volunteer computations provide greater computation power but in an untrusted environment. Thus, results received from participants must be verified. Common methods such as redundancy or "ringers" can drain useful resources or may be too computationally intensive. In this project, I explored applying Kohonen Self-Organizing maps, which project high-dimensionality data onto one or two dimensions, to automatically validate computational results.

Work Experience

University of Richmond Web Development - August 2002 – August 2005

- Building and maintaining web applications
- Static page design and maintenance

Freelance Consultant – Winter 1998 – Present

- Web design for assorted artisans and organizations
- Development of a web spider and data processor for collecting sports data

Computer Skills

Languages: Java, C/C++, some Scheme, Python, and x86 assembly

Software: Pin, JikesRVM, SimpleScalar, LaTeX, Visual Studio