

Eric S. Missimer

CONTACT INFORMATION *E-mail:* missimer@cs.bu.edu
Webpage: <http://cs-people.bu.edu/missimer>

WORK EXPERIENCE Hypervisor Engineer **March 2016 - June 2017**
Barkly Protects Inc.

Responsible for hypervisor and Windows kernel filter driver development. I worked on a small team that developed a thin hypervisor for interposing on Windows system calls to do behavioral malware detection. Used Intel VT-x and EPT technology in the hypervisor implementation. Developed technology to trace program execution undetectable by typical malware anti-analysis techniques. Implemented runtime malware behavior detection in both the hypervisor and Windows kernel filter drivers. Developed the Barkly kernel filter driver framework for rapid development of new malware detection. Also developed the testing suite used to ensure the stability of the hypervisor and kernel drivers.

Software Intern Graduate Level **June 2015 - August 2015**
Intel Corporation

For my summer internship at Intel I was responsible for porting the Quest operating system that I have worked on at Boston University to the Intel Edison board. This was a proof of concept to show how the Edison running Quest can be used for applications with hard real-time requirements that cannot be met with Linux. It involved modifying U-Boot to boot Quest instead of Linux. I also had to rewrite the portions of Quest that relied on ACPI to use the simple firmware interface (SFI) instead. I removed some dependencies we had on PC-compatible machines that were not present on the Edison (such as the PIT and PIC). Finally, I had to port the GPIO driver from Linux to Quest. The source code for both the modified U-Boot and Quest can be found on github.

Intern Member of Technical Staff in the Office of the CTO **May 2014 - November 2014**
VMware, Inc

Worked on ESXi, VMware's flagship product, specifically the on the VMKernel and VProbes. VProbes allows a VM guest or ESXi to be instrumented during run-time to observe and collect data about ESXi or a guest. It is similar to Linux Kprobes.

Windows System Administrator **Nov 2011 - July 2013**
Boston University, Computer Science Department

RESEARCH INTERESTS operating systems, real-time embedded systems, virtualization, robotics

EDUCATION **Boston University**, Boston, Massachusetts USA

B.A., Computer Science

- Magna Cum Laude
- Admitted: Fall 2007
- Graduated: September 25, 2010

Ph.D., Computer Science

- Advisor: Richard West
- Thesis Defended
- Admitted: Fall 2010
- Graduating: September, 2017

ACADEMIC
EXPERIENCE

Boston University, Boston, Massachusetts USA

Graduate Student

September, 2010 - September 2017

Includes current Ph.D. research, Ph.D. coursework.

Thesis (defended): *Distributed Real-Time Fault Tolerance in a Virtualized Separation Kernel*

Ph.D. research includes:

- **Quest Operating System Development**

Quest (www.questos.org) is a real-time high-confidence operating system designed for embedded and safety-critical applications. It has been developed from scratch at Boston University and runs on the x86 architecture. Some of the features it supports are SMP, Unix-like syscalls, hardware virtualization, a multi-kernel mode and separation kernel mode. It is currently the active research project of myself and others working under my advisor Prof. Richard West.

- **Real-Time I/O**

Developed a real-time USB EHCI host controller driver for Quest allowing real-time I/O and communication. This allows both real-time sensor input and actuator control along with real-time point-to-point communication between hosts. This is advantageous as embedded systems are becoming more distributed and more demanding in their sensory data input.

- **Virtualization**

Quest has a virtualization variant that uses hardware virtualization technology to enable isolation between sandboxes in a low-cost and predictable manner. As opposed to traditional virtual machine monitors the Quest monitor does not perform scheduling of virtual machines. Instead, it is responsible for isolation of sandboxes, inter-sandbox communication and restarting sandboxes. This allows Quest to operate as a separation kernel.

- **Fault Detection and Recovery**

Developed a novel fault recovery technique, that can take advantage of the Quest separation kernel design, that allows a faulty process to be *rolled-forward* to a correct future state. Timing guarantees can be made on the recovery cost so faults can be taken into account when profiling the system.

Undergraduate Research Assistant

November, 2007 - September, 2010

Worked on multiple projects including:

- A 3D graphics based interactive language learning software
- ACE - statistical analysis software developed for epidemiology
- Camera Mouse - assistive mouse replacement technology for the severely motion impaired
- Customizable Keyboard - onscreen keyboard developed to be used with Camera Mouse

PUBLICATIONS

Eric Missimer, Katherine Missimer, Richard West, "Mixed-Criticality Scheduling with I/O", in Proceedings of the 28th Euromicro Conference on Real-Time Systems (ECRTS), Toulouse, France, July 5-8, 2016.

Richard West, Ye Li, Eric Missimer and Matthew Danish, "A Virtualized Separation Kernel for Mixed Criticality Systems", in ACM Transactions on Computer Systems, Volume 34, Issue 3, Article 8, June 2016 (DOI: 10.1145/2935748)

Ye Li, Richard West, Zhuoqun Cheng and Eric Missimer, "Predictable Communication and Migration in the Quest-V Separation Kernel", in Proceedings of the 35th IEEE Real-Time Systems Symposium (RTSS), Rome, Italy, December 2-5 2014

Eric Missimer, Richard West and Ye Li, "Distributed Real-Time Fault Tolerance on a Virtualized Multi-core System", in Proceedings of the 10th Annual Workshop on Operating Systems Platforms for Embedded Real-Time applications (OSPERT), in conjunction with ECRTS, Madrid, Spain, July 8, 2014

Ye Li, Richard West, Eric Missimer, "A Virtualized Separation Kernel for Mixed Criticality Systems", in Proceedings of the 10th ACM SIGPLAN/SIGOPS International Conference on Virtual Execution Environments (VEE), Salt Lake City, Utah, March 1-2 2014.

Ye Li, Richard West, Eric Missimer, "The Quest-V Separation Kernel for Mixed Criticality Systems", in Proceedings of the 1st International Workshop on Mixed Criticality Systems (WMC) at the 34th IEEE Real-Time Systems Symposium (RTSS 2013), Vancouver, Canada, December 2013.

Eric Missimer, Ye Li and Richard West, "Real-Time USB Communication in the Quest Operating System", in Proceedings of the 19th IEEE Real-Time and Embedded Technology and Applications Symposium (RTAS 2013), Philadelphia, USA, April 2013.

Richard West, Ye Li and Eric Missimer, "Time Management in the Quest-V RTOS", in the 8th Annual Workshop on Operating Systems Platforms for Embedded Real-Time Applications (OSPERT), Pisa, Italy, July 10, 2012.

Eric Missimer, Samuel Epstein, John J. Magee, and Margrit Betke. "Customizable keyboard." In The 12th International ACM SIGACCESS Conference on Computers and Accessibility (ASSETS 2010), Orlando, Florida, USA, October 2010.

John J. Magee, Samuel Epstein, Eric Missimer, and Margrit Betke. "Adaptive mappings for mouse-replacement interfaces." In The 12th International ACM SIGACCESS Conference on Computers and Accessibility (ASSETS 2010), Orlando, Florida, USA, October 2010.

Eric Missimer and Margrit Betke. "Blink and wink detection for mouse pointer control." The 3rd ACM International Conference on Pervasive Technologies Related to Assistive Environments (PETRA 2010), Pythagorion, Samos, Greece. June 2010. 8 pp.

TECHNICAL
REPORTS

Eric Missimer, Katherine Zhao, Richard West: "Mixed-Criticality Scheduling with I/O", Technical Report: arXiv:1512.07654, arXiv.org.

Richard West, Ye Li, Eric Missimer, "Quest-V: A Virtualized Multikernel for Safety-Critical Real-Time Systems", Technical Report: arXiv:1310.6349, arXiv.org.

Ye Li, Eric Missimer, Richard West, "Predictable Migration and Communication in the Quest-V Multikernel", Technical Report: arXiv:1310.6301, arXiv.org.

Ye Li, Richard West, Eric Missimer, "The Quest-V Separation Kernel for Mixed Criticality Systems", Technical Report: arXiv:1310.6298, arXiv.org.

TEACHING
EXPERIENCE

Computer Science Teaching Award
Boston University, Computer Science Department

Awarded Spring 2015

Teaching Fellow – CS552 Operating Systems
Boston University, Computer Science Department

Spring 2015

Senior Teaching Fellow (Lecturer) – CS410 Advanced Software Systems
Boston University, Computer Science Department

Fall 2014

Teaching Fellow – CS410 Advanced Software Systems
Boston University, Computer Science Department

Spring 2013

Teaching Fellow – CS410 Advanced Software Systems
Boston University, Computer Science Department

Spring 2012

Teaching Fellow – CS212 Physical Computing
Boston University, Computer Science Department

Spring 2011

ADDITIONAL
EXPERIENCE

Secondary Reviewer for IEEE Real-Time and Embedded Technology and Applications Symposium (RTAS)

Secondary Reviewer for IEEE Real-Time Systems Symposium (RTSS)