**M. Scott Doerrie**

|  |  |
| --- | --- |
| 8811 Colesville Road, Apt 1105  Silver Spring, MD 20910 | Email: doerrie@gmail.com  Phone: 443-413-3845 |

**Interests:**

Formal Methods and Verification, Type Systems, Protection Mechanisms, Security Policies, Capability-based Systems and Run-times, and Robust Operating Systems.

**Education:**

|  |  |
| --- | --- |
| **Johns Hopkins University** | August 2015 |
| Ph.D., Computer Science  Dissertation: *Confidence in Confinement: An Axiom-free, Mechanized Verification of Confinement in Capability-based Systems*  **Advisor: Dr. Jonathan S. Shapiro** | |

|  |  |
| --- | --- |
| **Gustavus Adolphus College**  B.A., Computer Science | May 2002 |

**Technical Expertise:**

|  |  |
| --- | --- |
| **Professional:** | Java, JDB, Perl, Python, SQL, C, C++, GDB, GNU Make, Bash, HTML, CSS, JavaScript, Linux kernel tracing, Linux VFS, Linux kernel modules, ACL2, Isabelle/HOL, TWELF, Coq, OCaml, SML, LaTeX, Git, Mercurial, SVN, CVS, QEMU, LotusScript, Lotus @Commands. |
| **Personal:** | Haskell, Agda, Jekyll, Liquid, Sass, Arduino, Raspberry Pi, OpenSCAD. |

**Research and Work Experience:**

|  |  |
| --- | --- |
| **Johns Hopkins University** | 2003-2015 |
| Graduate Researcher, *Systems Research Laboratory*  Core Technologies: Coq, ACL2, TWELF, Isabelle/HOL, OCaml, SML, Perl, GNU Make, Latex, Mercurial, GIT. | |

* Conclusively demonstrated that capability-based systems can enforce confinement by leveraging mechanical verification using the Coq proof assistant.
* Increased confidence in the model and widened its future applicability by providing permission propagation theorems independent of any specific policy that are subsequently applied to the confinement problem.
* Hardened the model by providing a general implementation that can be refined to describe any capability system and completes the model for an axiom-free proof.
* Critically assessed multiple first-order and higher-order proof systems including ACL2, TWELF, Isabelle/HOL, and Coq.
* Ensured the Coyotos and BitC projects would be fit for future verification by providing continual feedback through their development.

|  |  |
| --- | --- |
| **Johns Hopkins University** | 2003-2006 |
| Research Assistant, *Systems Research Laboratory*  Core Technologies: C, GNU Make, GDB, GIT, Mercurial, Linux kernel tracing, Linux VFS. | |

* Was a key participant in the coevolution of BitC and Coyotos.
* Reconciled the Intel paging mechanism with Coyotos’ general-purpose guarded page tables.
* Motivated guarded page tables by simulating process behavior taken from traces in Linux and demonstrating that guarded page tables would not add substantial overhead.
* Guided early BitC language development through issues regarding formalization and verification.

|  |  |
| --- | --- |
| **Oracular Systems and Software** | 2002-2003 |
| Software Consultant and System Administrator  Core Technologies: Java, HTML, JavaScript, CSS, LotusScript, Lotus @Commands, SQL, SVN. | |

* Improved corporate utility to clients by producing rapid prototypes of software services.
* Adapted to client software configurations when realizing prototypes into high-quality deliverables.
* Increased system uptime for clients and the company by diagnosing server and service issues.

|  |  |
| --- | --- |
| **Gustavus Adolphus College** | 2000-2002 |
| Electronic Reserves Developer, *Folke Bernadotte Memorial Library*  Core Technologies: Perl, SQL, GNU Make, HTML, JavaScript. | |

* Worked within a self-directed and self-managed team to design and deploy the Gustavus Library Electronic Reserves system.
* Took personal responsibility for the system backend and data processing.

**Teaching Experience:**

|  |  |
| --- | --- |
| **Johns Hopkins University** | 2009 |
| Teaching Assistant to Dr. Jonathan S. Shapiro, *Microkernels*  Core Technologies: C, C++, GNU Make, QEMU, GDB, GIT, Mercurial. | |

* Provided a constructive environment by assisting students with development and debugging in the Coyotos microkernel.
* Responded to student needs by offering lengthy mentoring sessions every night.

|  |  |
| --- | --- |
| **Johns Hopkins University** | 2006-2009 |
| Instructor, *Operating Systems*  Core Technologies: C, GNU Make, QEMU, GDB, GIT, Mercurial, Linux kernel modules, Linux VFS. | |

* Responsible for all aspects of course execution.
* Provided an extremely practical, hands-on, and critical examination of operating system design reflecting both academic and practical perspectives.
* Improved the scope of the course by designing and presenting novel materials on 64-bit memory architectures, Linux kernel module development and in-kernel debugging, and capability-based systems.
* Prepared students professionally with advanced course material including the slab allocator, file-system implementations and Linux VFS, microkernels, and virtualization techniques.

|  |  |
| --- | --- |
| **Johns Hopkins University** | 2007 |
| Lecturer, *TWELF Lecture series, Systems Research Laboratory*  Core Technologies: TWELF. | |

* Introduced the Systems Research Laboratory to verification techniques in TWELF by designing and presenting a half-semester lecture series.
* Communicated topics in advanced type theory and totality analysis and their relationship to mathematical proofs.

**Publications:**

M. S. Doerrie, *Confidence in Confinement: An Axiom-free, Mechanized Verification of Confinement in Capability-based Systems*, Ph. D. Dissertation, July 2015.

J. S. Shapiro, M. Scott Doerrie, E. Northup, S. Sridhar, and M. Miller, "Towards a Verified, General-Purpose Operating System Kernel", *Proc. 2004 NICTA Operating Systems Verification Workshop*, 2004.

J. S. Shapiro, S. Sridhar, S. Doerrie, M. Miller, E. Northup. *BitC Language Specification*, June 2006.

**Honors and Awards:**

1998 - 2001 Gustavus Dean’s List

1998 - 2001 Trustee Scholarship

1998 - 2001 Norelius Service Award

1998 BSA Eagle Scout