Diyu Zhou

Ph.D.
University of California, Los Angeles
404 Westwood Plaza
Los Angeles, California, 90024, USA

Email: zhoudiyu@cs.ucla.edu
Web: http://web.cs.ucla.edu/~zhoudiyu
Phone: +1-424-666-5977

Research Interests

Operating Systems, Dependable Computing, Virtualization, Cloud Computing, Computer Architecture, Software Debugging and Testing

Dissertation Research

My thesis is focused on building **practical, low-overhead** dependability mechanisms (i.e., fault-tolerance mechanisms and debugging tools) for hypervisors, containers, and server applications. The main approach involves identifying sweet spots in the design space of dependability mechanisms that balance soundness and overhead, leveraging hardware/operating system/hypervisor support.

Education

University of California, Los Angeles
Ph.D., Computer Science — December 2020
Advisor: Yuval Tamir
Thesis: Practical Dependable Systems with OS/Hypervisor Support

University of California, Los Angeles
M.S., Computer Science — June 2018

Peking University
B.S., Computer Science — June 2013

Peer-Reviewed Publications

[C.1] **Fast Hypervisor Recovery Without Reboot**
pdf Diyu Zhou and Yuval Tamir
In proceedings of *48th IEEE/IFIP International Conference on Dependable Systems and Networks (DSN 2018)*, Luxembourg City, Luxembourg, pages 115-126, June 2018. Acceptance Rate: 51/202 = 25.2%

[C.2] **PUSh: Data Race Detection Based on Hardware-Supported Prevention of Unintended Sharing**
pdf Diyu Zhou and Yuval Tamir
In proceedings of *ACM/IEEE 52nd Annual Symposium on Microarchitecture (Micro 2019)*, Columbus, OH, USA, pages 886–898, October 2019. Acceptance Rate: 79/345 = 22.8%

[C.3] **Fault-Tolerant Containers Using NiLiCon**
pdf Diyu Zhou and Yuval Tamir
In proceedings of *34th IEEE International Parallel and Distributed Processing Symposium (IPDPS 2020)*, New Orleans, LA, USA, pages 1082-1091, May 2020. Acceptance Rate: 110/446 = 24.7%

[C.4] **HyCoR: Fault-Tolerant Replicated Containers Based on Checkpoint and Replay**
Diyu Zhou and Yuval Tamir
Under preparation

[J.1] **Data Race Detection by Prevention of Unintended Sharing Using PUSh**
Diyu Zhou and Yuval Tamir
Under preparation

Dissertations

[T.1] **Practical Dependable Systems with OS/Hypervisor Support**
Diyu Zhou
Ph.D. Dissertation, University of California, Los Angeles, December, 2020

Past Projects

***NiLiHype*: Fast Hypervisor Recovery** 2017
For a hypervisor, the latency to reboot a new instance to recover it from transient faults (e.g., transient bit-flips in memory or registers) is unacceptably high. *NiLiHype* [C.1] is a hypervisor resilience technique that recovers the hypervisor from transient faults by resetting it to a quiescent state that is highly likely to be valid. Compared to the prior state-of-the-art work that is based on reboot, *NiLiHype* reduced the service interruption time during recovery from 713ms to 22ms, a factor of over 30x, while achieving nearly the same recovery success rate.

***PUSh*: Low-Overhead Hardware-Based Data Race Detectors** 2018
PUSh [C.2] is an extremely low-overhead dynamic data race detector based on requiring the programmer to specify intended sharing and the use of existing memory protection hardware to detect unintended sharing. A key novelty in *PUSh* is its use of memory protection keys, a hardware feature available in some recent versions of the x86 ISA, to eliminate the overhead of multiple page tables and more critically, the overhead of updating page tables every time a lock is acquired or released. *PUSh* contributes an efficient algorithm to detect incorrect annotations that can hide data races, which can be applied to other annotation-based data race detectors. Several other key performance/memory optimization is achieved by enhancing the memory management subsystem in the kernel [J.1]. For a set of 10 real-world benchmarks, *PUSh*'s memory overhead is less than 5.8% and performance overhead is less than 54% while detecting all the data races reported by ThreadSanitizer.

***NiLiCon*: Fault-Tolerant Containers** 2019
NiLiCon [C.3], to the best of our knowledge, is the first container fault-tolerance mechanism that is application- and client-transparent and supports stateful applications. *NiLiCon* applies *Remus*, a widely used VM replication technique, to containers. A key implementation challenge is that, compared to VMs, there is much tighter coupling between the container state and the state of the underlying platform. *NiLiCon* meets this challenge with various kernel enhancements and achieves performance that is competitive with *Remus*.

***HyCoR*: Efficient Replicated Containers with Deterministic Replay** 2020
HyCoR [C.4] enhances *NiLiCon* with deterministic replay to decouple the checkpointing interval from output delay. It thus addresses a fundamental drawback in all *Remus*-based replication mechanisms: unacceptably long delays of outputs to clients. With *HyCoR*, the non-deterministic events in the primary are recorded and sent to the backup for future replaying upon failover. The outputs thus only need to be delayed by the short amount of time of sending the non-deterministic events to the backup. For a set of eight benchmarks, with *HyCoR*, the incurred extra delay of outputs is less than 600μs versus 38ms-63ms with *NiLiCon*. Since output delays are not determined by the checkpointing interval, for data-race-free applications, *HyCoR* can achieve better performance by using a longer checkpointing interval. Specifically, with a one-second checkpointing interval, *HyCoR* achieves a performance overhead of 2%-58% versus 18%-139% with *NiLiCon*.

Teaching Experience

University of California, Los Angeles
Teaching Assistant, CS111: Operating Systems
Summer 2020; Fall 2019; Summer 2019; Summer 2017; Winter 2017; Fall 2016; Fall 2015; Winter 2015

University of California, Los Angeles
Teaching Assistant, CS151B: Computer Systems Architecture
Spring 2019; Fall 2018; Spring 2018; Spring 2017; Summer 2016; Spring 2016; Winter 2016; Summer 2015; Spring 2015

University of California, Los Angeles
Teaching Assistant, CS35L: Software Construction Laboratory
Winter 2019; Winter 2018; Fall 2017

Honers and Scholarships

UCLA Computer Science Department Fellowship	2020
UCLA Computer Science Department Travel Grant	2019
Micro 2019 Travel Grant	2019
UCLA Doctoral Student Travel Grant	2019
DSN 2018 Travel Grant	2018
UCLA Graduate Division Fellowship	2013
Tencent Technology Excellence Scholarship	2012
Yihai Kerry Scholarship	2011
Second Prize, The 10th Peking University Netease Youdao Cup Programming Contest	2011
5th in the world and 1st in Region 10, IEEE Xtreme 4.0 Programming Contest	2010
Silver Medal, The 2010 ACM-ICPC Asia Tianjin Regional Contest	2010
Silver Medal, The 2010 ACM-ICPC Asia Harbin Regional Contest	2010
Second Prize, The 9th Peking University Netease Youdao Cup Programming Contest	2010
Gold Medal, National Olympiad in Informatics in Provinces	2008

Languages

English: Full professional proficiency
Chinese: Native

Professional Memberships

ACM Member, ACM SIGMICRO
IEEE Member