

Zachary Yedidia

zyedidia@stanford.edu

zyedidia.github.io, github.com/zyedidia

Areas of interest: operating systems, security, compilers, computer architecture.

EDUCATION

Stanford University

Stanford, CA

Ph.D. candidate in Computer Science

Sep. 2021 – 2027 (expected)

Advisors: Prof. David Mazières and Prof. Dawson Engler

Harvard University

Cambridge, MA

S.M. in Computer Science

Sep. 2020 – May 2021

Harvard College

Cambridge, MA

A.B. Magna cum laude with highest honors in Computer Science

Aug. 2017 – May 2021

PUBLICATIONS

Deterministic Client: Enforcing Determinism on Untrusted Machine Code

Zachary Yedidia, Geoffrey Ramseyer, David Mazières

To appear at OSDI 2025

Segue & ColorGuard: Optimizing SFI Performance and Scalability on Modern Architectures

Shravan Narayan, Tal Garfinkel, Evan Johnson, Zachary Yedidia, Yingchen Wang, Andrew Brown, Anjo Vahldiek-Oberwagner, Michael LeMay, Wenyong Huang, Xin Wang, Mingqiu Sun, Dean Tullsen, Deian Stefan
ASPLOS 2025

DeCl: Deterministic and Metered Native Sandboxes

Zachary Yedidia, Geoffrey Ramseyer, David Mazières

SIB 2024 (workshop)

Lightweight Fault Isolation: Practical, Efficient, and Secure Software Sandboxing

Zachary Yedidia

ASPLOS 2024

Fast Incremental PEG Parsing

Zachary Yedidia and Stephen Chong

SLE 2021

Precision Batching: Bitserial Decomposition for Efficient Neural Network Inference on GPUs

Maximilian Lam, Zachary Yedidia, Colby Banbury, Vijay Janapa Reddi

PACT 2021

OPEN SOURCE PROJECTS

Lightweight Fault Isolation (LFI)

Paper, GitHub Project

A state-of-the-art software-based sandboxing system that enforces memory isolation on machine code within a single address space. LFI can be used for in-process library sandboxing, low-latency serverless functions, application sandboxing, smart contracts, and more. This is my primary current project.

Micro Text Editor

Website, GitHub Project

I created and launched a text editor called Micro in April 2016. Since then, Micro has gained over 25,000 stars on GitHub, more than 750,000 downloads, and 100+ contributors. It aims to be a successor to Nano as a simple-to-use terminal-based text editor. Micro is available in many package managers such as Homebrew, Apt, Snap, AUR, Chocolatey and more.

GPeg	Paper, Thesis, Slides, GitHub Project
Library for PEG parsing, as part of my senior thesis research with Professor Stephen Chong. Additionally includes a library for syntax highlighting: github.com/zyedidia/flare .	
Literate Programming Tool	Website, GitHub Project
A tool for compiling Literate programs written in any programming language.	
Eget	GitHub Project
A tool for automatically installing pre-built binaries distributed in GitHub releases.	
Go Generic Data Structures	Github Project
A library of generic data structures for Go.	
Knit	GitHub Project
A flexible build tool that combines Lua with Make's declarative rules language.	
<hr/>	
WRITING	
Incremental PEG Parsing	
Senior thesis (2021), advised by Prof. Stephen Chong.	
SystemVerilog Guide	
Course materials for CS 141 (Spring 2020) at Harvard	
<hr/>	
AWARDS	
NSF Graduate Research Fellowship (2022-2024).	
<hr/>	
SERVICE	
External reviewer for ACM TOPLAS.	
<hr/>	
EXPERIENCE	
SiFive	San Mateo, CA
<i>Chisel Team Intern</i>	Summer 2022
Zero ASIC	Virtual
<i>Intern</i>	Summer 2021
Harvard University	Cambridge, MA
<i>HCRP Research Fellow (Advisor: Prof. Stratos Idreos)</i>	Summer 2020
Raytheon Company	Tucson, AZ
<i>Internal Research and Development Intern</i>	July – August Summer 2019
Princeton University	Princeton, NJ
<i>Research Assistant (Advisor: Prof. Naveen Verma)</i>	May – June Summer 2019
Harvard University	Cambridge, MA
<i>PRISE Research Fellow (Advisor: Prof. Eddie Kohler)</i>	Summer 2018
<hr/>	
TEACHING	
Distributed Systems (CS244b)	Stanford
<i>Course Assistant for Prof. David Mazières</i>	Spring 2024
Advanced Topics in Operating Systems (CS240)	Stanford
<i>Course Assistant for Profs David Mazières and Dawson Engler</i>	Spring 2023
Using Bits to Control Atoms (CS49n)	Stanford
<i>Course Assistant for Prof. Dawson Engler</i>	Autumn 2021

Systems Programming and Machine Organization (CS61)	Harvard
<i>Teaching Fellow for Profs Eddie Kohler and Minlan Yu</i>	<i>Fall 2020</i>
Computing Hardware (CS141)	Harvard
<i>Teaching Fellow for Profs David Brooks and Vijay Reddi</i>	<i>Spring 2019, Spring 2020</i>
Compilers (CS153)	Harvard
<i>Teaching Fellow for Prof. Stephen Chong</i>	<i>Fall 2019</i>

TALKS

Deterministic Client: Enforcing Determinism on Untrusted Machine Code	
<i>MIT DCI</i>	<i>May 2025</i>
Securing Native Libraries on Android with LFI	
<i>Qualcomm Product Security Summit</i>	<i>May 2025</i>
Deterministic Client: Enforcing Determinism on Untrusted Machine Code	
<i>Stanford Security Workshop</i>	<i>April 2025</i>
Lightweight Fault Isolation	
<i>Software Compartmentalization Community Tech Talk</i>	<i>April 2025</i>
Deterministic Client: Enforcing Determinism on Untrusted Machine Code	
<i>Stanford Security Lunch</i>	<i>March 2025</i>
Sandboxing Native Libraries on Android with Lightweight Fault Isolation (LFI)	
<i>Google</i>	<i>February 2025</i>
DeCl: Deterministic and Metered Native Sandboxes	
<i>SIB</i>	<i>September 2024</i>
DeCl: Deterministic and Metered Native Sandboxes	
<i>Stanford Software Lunch</i>	<i>July 2024</i>
Lightweight Fault Isolation: Practical, Efficient, and Secure Software Sandboxing	
<i>ASPLOS</i>	<i>April 2024</i>
Lightweight Fault Isolation: Practical, Efficient, and Secure Software Sandboxing	
<i>Stanford Security Workshop</i>	<i>April 2024</i>
Lightweight Fault Isolation: Practical, Efficient, and Secure Software Sandboxing	
<i>Stanford Security Lunch</i>	<i>November 2023</i>
Classic Software Fault Isolation and WebAssembly	
<i>WebAssembly Research Day</i>	<i>October 2023</i>
Multiplix: using D for kernel development	
<i>DConf</i>	<i>September 2023</i>
Fast Incremental PEG Parsing	
<i>SLE</i>	<i>November 2021</i>
Fast Incremental PEG Parsing	
<i>Stanford Software Lunch</i>	<i>November 2021</i>