Extending Trusted Execution Environments in Architectural Simulators

Will Buziak
Dept. of Computer Science
Colorado School of Mines

Iris Bahar
Dept. of Computer Science
Colorado School of Mines

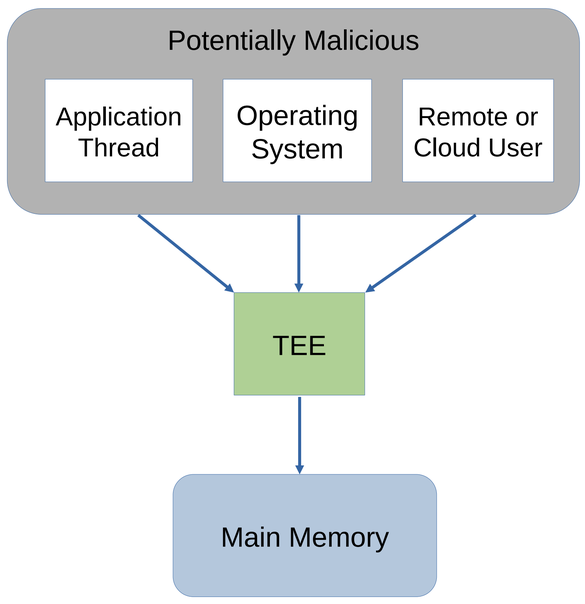
Tamara Silbergleit Lehman
Dept. of Electrical, Computer, & Energy Engineering
University of Colorado, Boulder

Zach Moolman
Dept. of Electrical, Computer, & Energy Engineering
University of Colorado, Boulder

Sam Thomas
Dept. of Computer Science
Brown University

Background

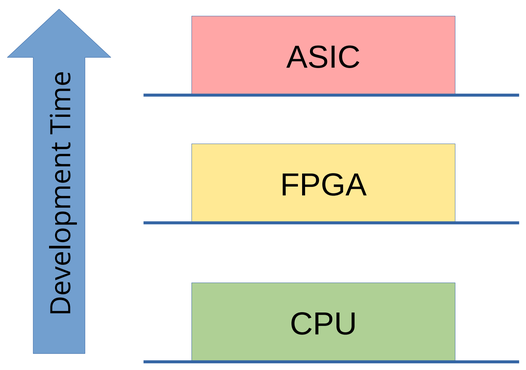
Trusted Execution Environments (TEEs) provide hardware guarantees that seek to protect the security and isolation of off-chip data.



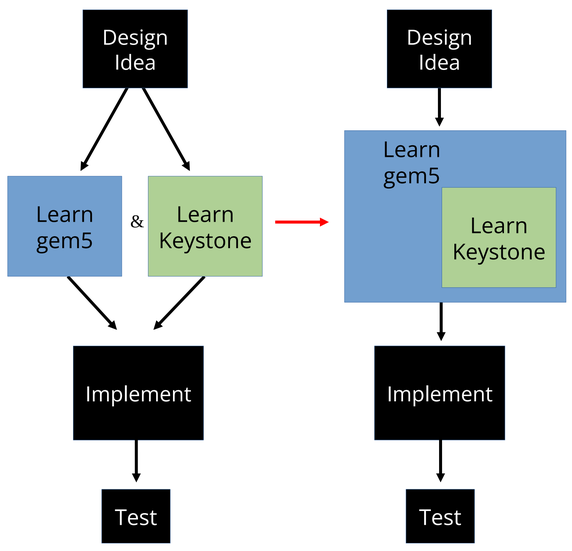
This work outlines methods for implementing and evaluating contributions to open-source TEEs within architectural simulation.

Motivation

Simulation allows a shorter pipeline from design idea to implementation testing.

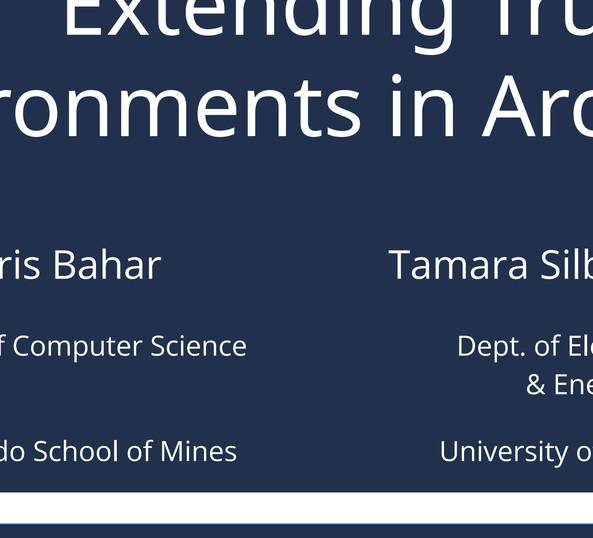


Achieving a baseline model is non-trivial, often allocating much of the development time to de-coupled, self-guided learning.

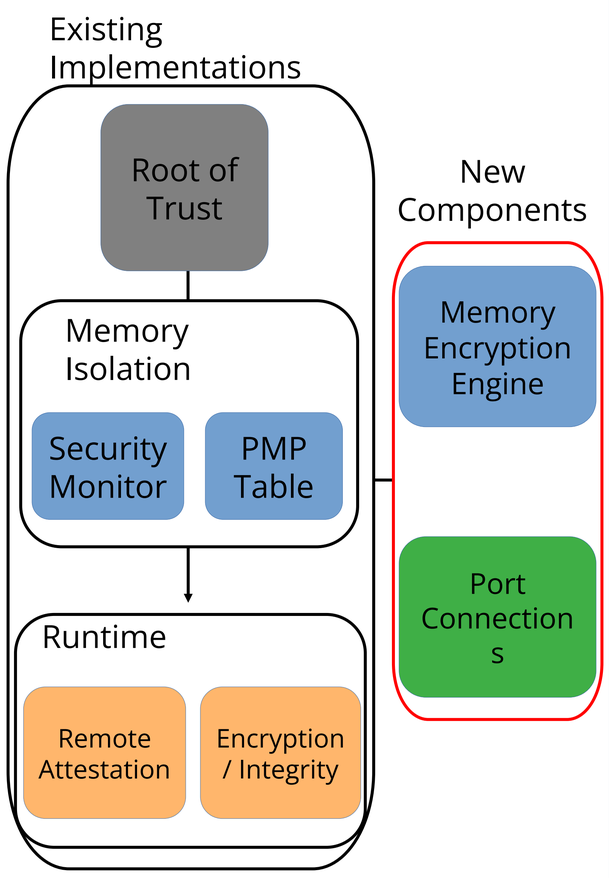


Keystone

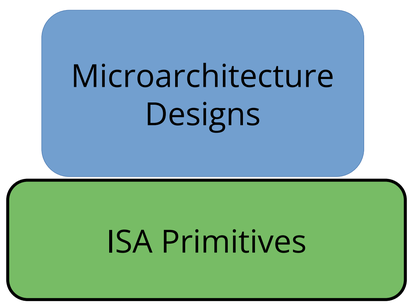
Many TEE implementations exist, but Keystone[1] is a popular, open-source version with many pre-existing simulator components.



Keystone provides security through memory isolation, utilizing customized RISC-V hardware primitives.

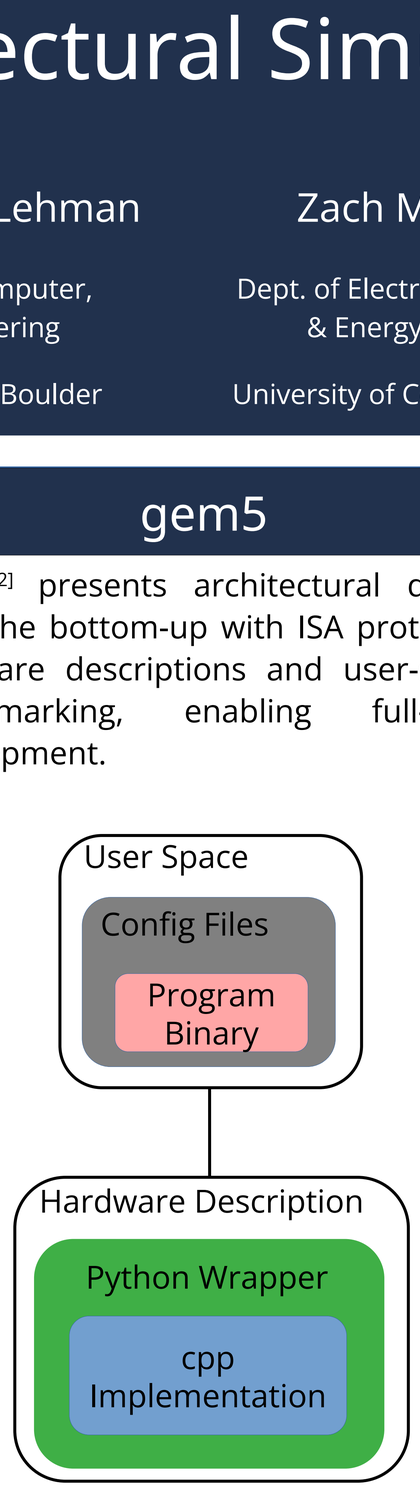


For researchers wishing to continue developing Keystone components, contributions must also implement hardware designs, built on corrresponding ISA extensions.

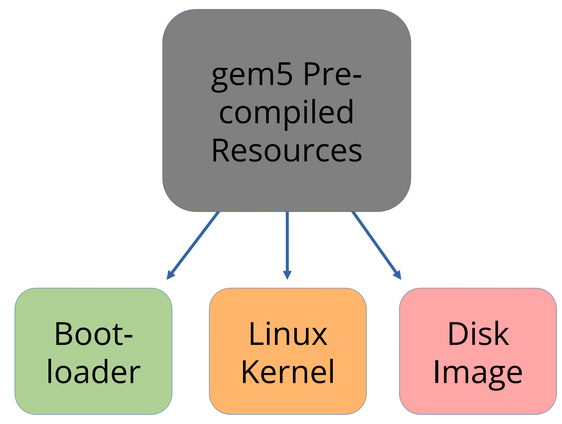


gem5

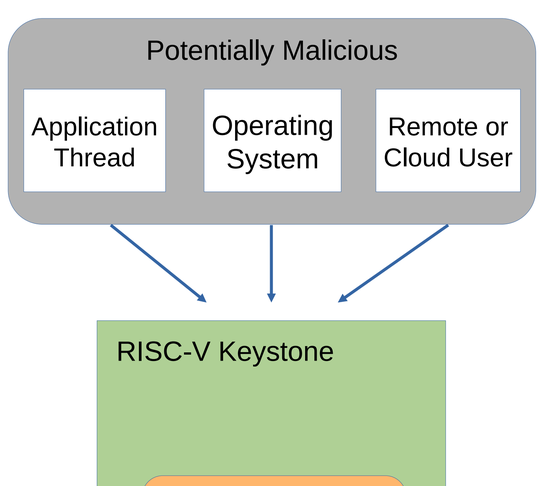
Gem5[2] presents architectural design from the bottom-up with ISA protocols, hardware descriptions and user-space benchmarking, enabling full-stack development.



In order to build Keystone in gem5, the developer must also make use of full-system resources.

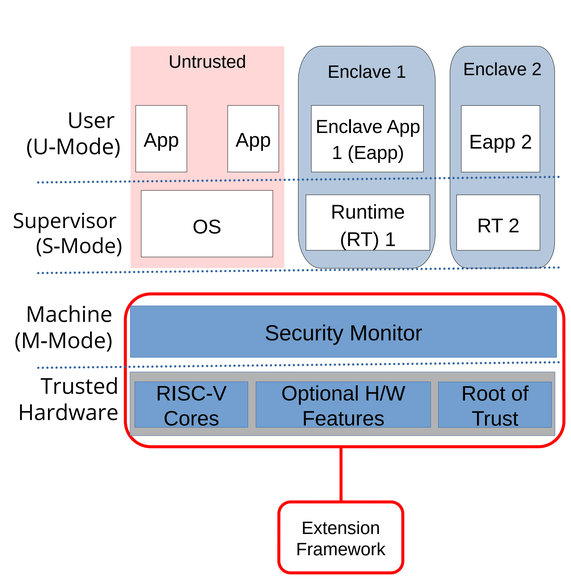


To further protect state-of-the-art TEEs, we extend[3] Keystone to include secure memory protocols in the gem5 simulation environment.



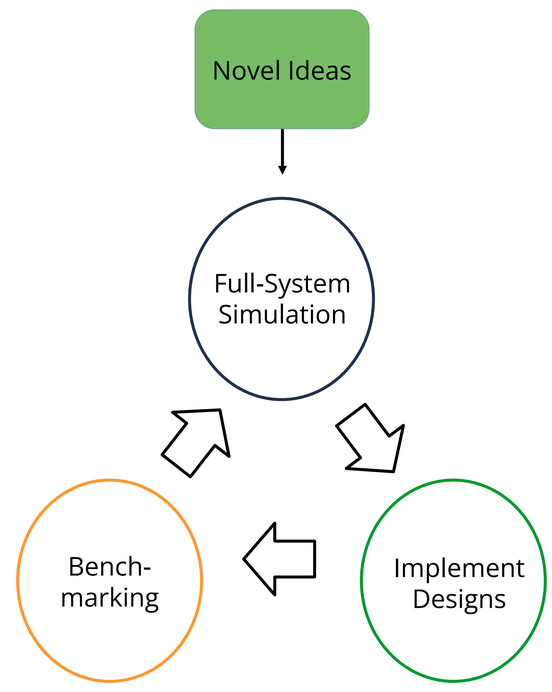
Future Work

This work proposes a framework that aims to assist future developers to implement their contributions.



Our goal is to provide researchers with tools that expedite the development cycle when working with TEEs, primarily targeting:

- New researcher learning curve
- Novel contribution development time
- Testing methods



Bibliography

[1] Dayeol Lee, David Kohlbrenner, Shweta Shinde, Krste Asanovic, and Dawn Song, Keystone: An Open Framework for Architecting Trusted Execution Environments, In *Fifteenth European Conference on Computer Systems (EuroSys '20)* 2020.

[2] Jason Lowe-Power, Abdul Mutaal Ahmad, Ayaz Akram, Mohammad Alian, and et. Al, The gem5 Simulator: Version 20.0+, (arXiv) 2007.

[3] Zach Moolman and Tamara Silbergleit Lehman, Extending RISC-V Keystone to Include Efficient Secure Memory, In: *Eighth Workshop on Computer Architecture Research with RISC-V (CARRV 2024)*