Citadel: A Trusted Reference Monitor for Linux using Intel SGX Enclaves

A.H. Bell-Thomas

Computer Laboratory, University of Cambridge

26th June, 2020

Background

Background

1. **Reference Monitor**

Background

1. **Reference Monitor**
   $\rightsquigarrow$ *Information Flow Control*

Background

1. **Reference Monitor**
   $\rightsquigarrow$ *Information Flow Control*

2. **Intel SGX**

Information Flow Control

- Access Control specifics *who* can access resources. IFC also mediates *how* they can be used once opened.
- Construct an abstract system of *entities*;
  $\rightsquigarrow$ processes, files, sockets, etc.
- Each *entity* carries a *security context*, defining its granular ownership or restriction information.
- Aim: achieve *non-interference* between all *security contexts*.

Information Flow Control

Very briefly;

- **Tagging**
  Entities must be uniquely and reliably identifiable to support decisions.

- **Tracking**
  Contexts are mutable to accommodate an evolving situation.

- **Policy Decisions**
  Is an operation acceptable given its consequences?
  e.g. $A \to B \iff A_s \preceq B_s \ \wedge \ A_i \succeq B_i$
  c.f. Biba, Bell-LaPadula

Decentralised Information Flow Control

- Centrally administered systems are highly restrictive.
- Idea: let entities specify their own protection policy for assets they own. Enforcement becomes *discretionary*, allowing more flexibility and support for operations such as *declassification*.

Enforcement is implemented using a *reference monitor*.

Linux Security Modules

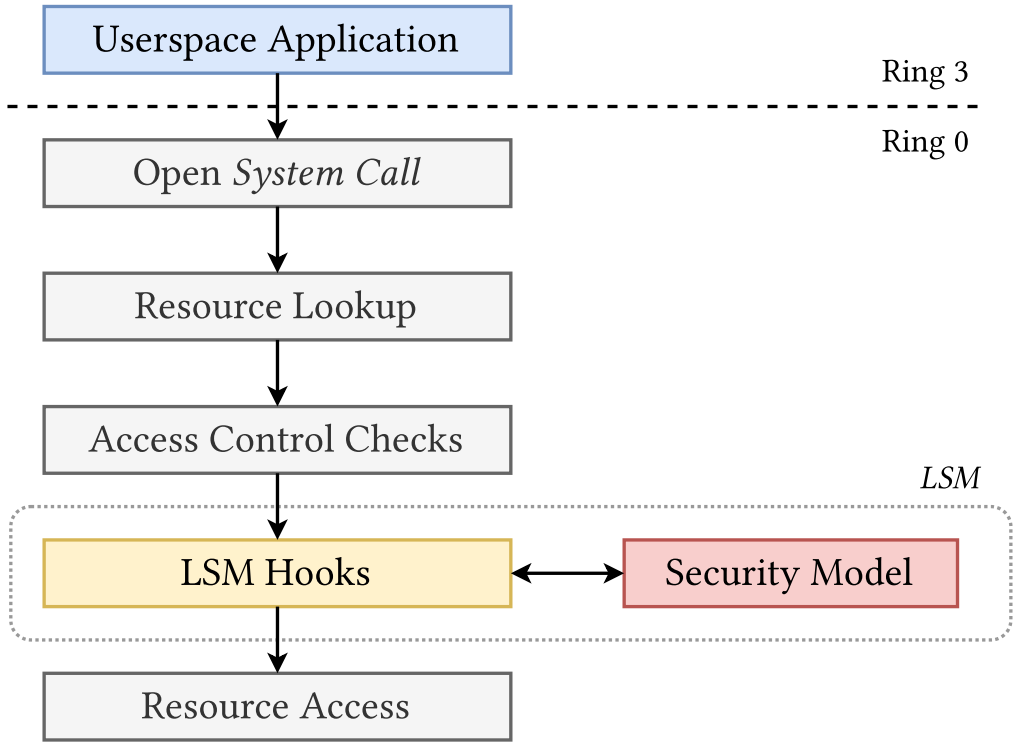


Figure: High level overview of the CITADEL architecture.

Intel SGX

A general-purpose *trusted execution environment* provided via x86 at the architectural level in modern processors.

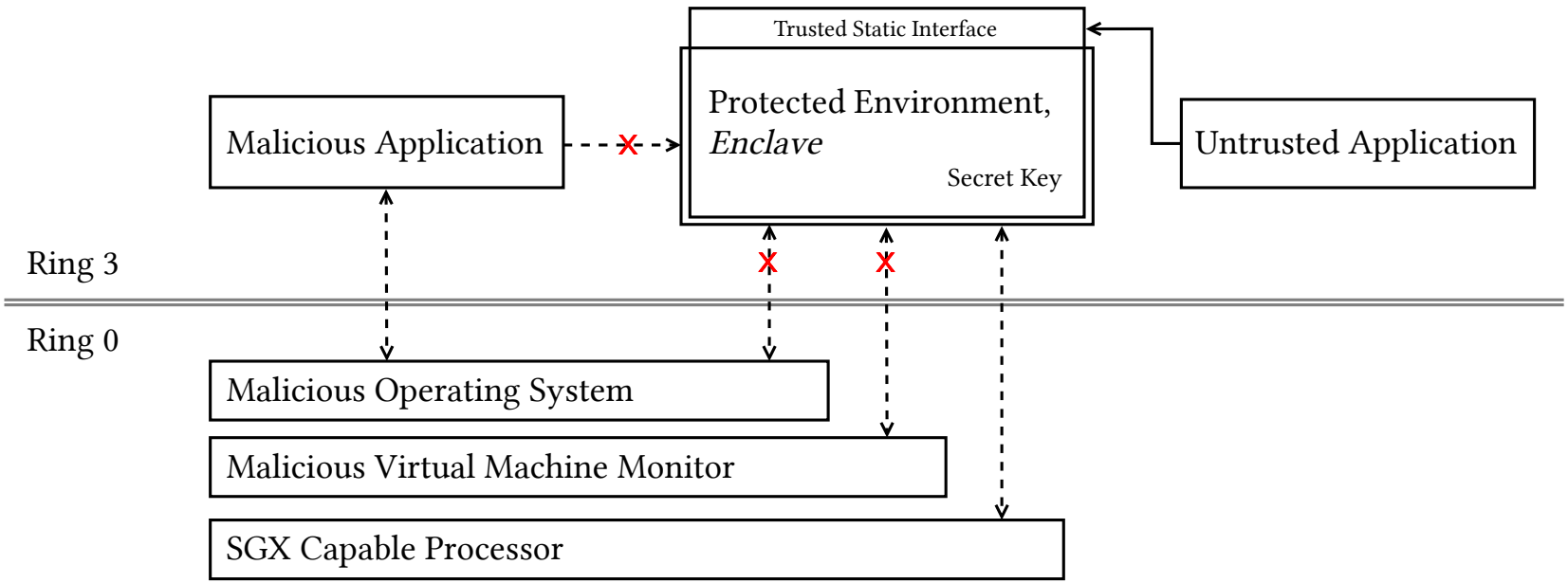


Figure: High level overview of the CITADEL architecture.

Intel SGX

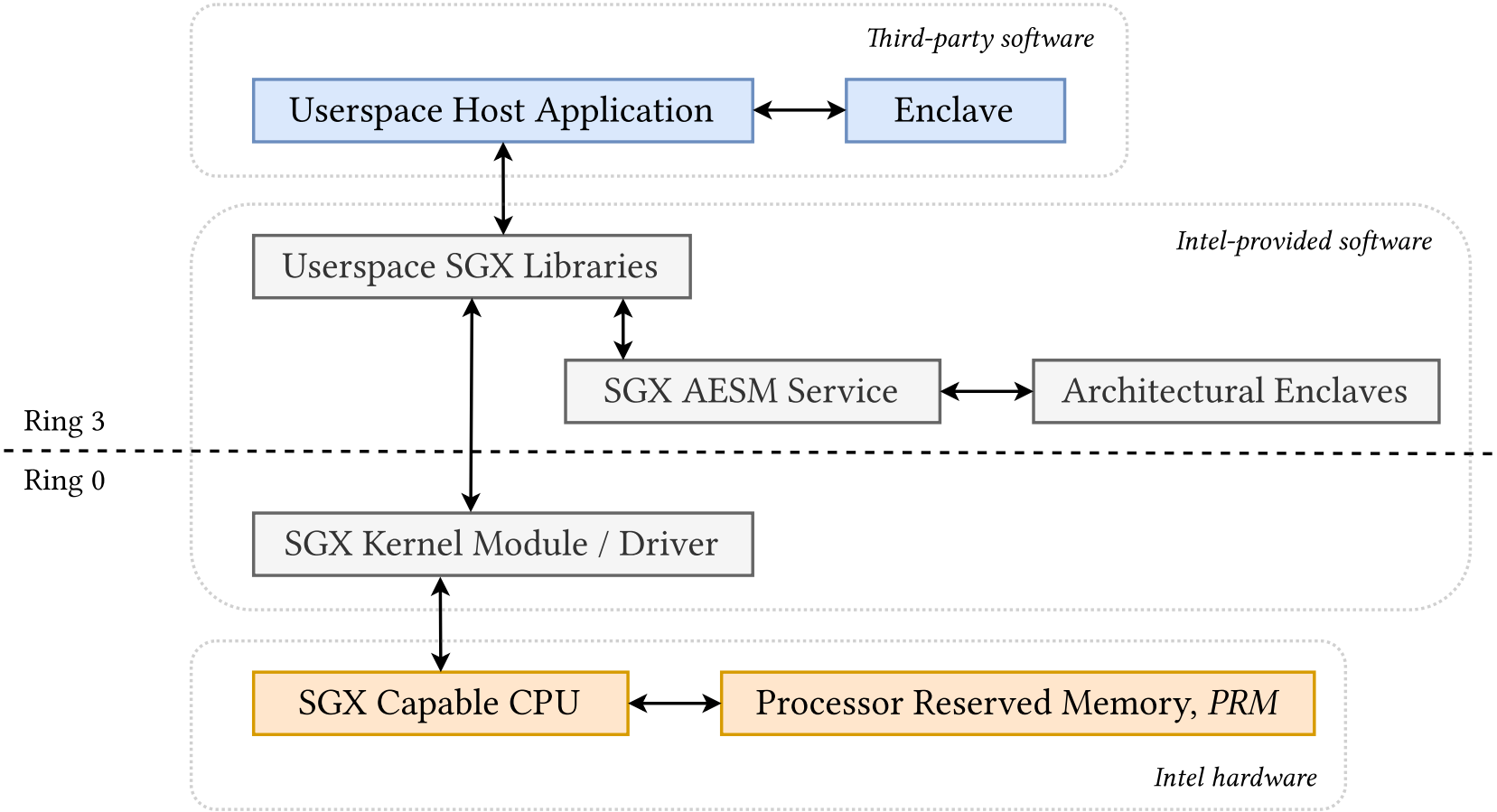


Figure: High level overview of the CITADEL architecture.

Citadel

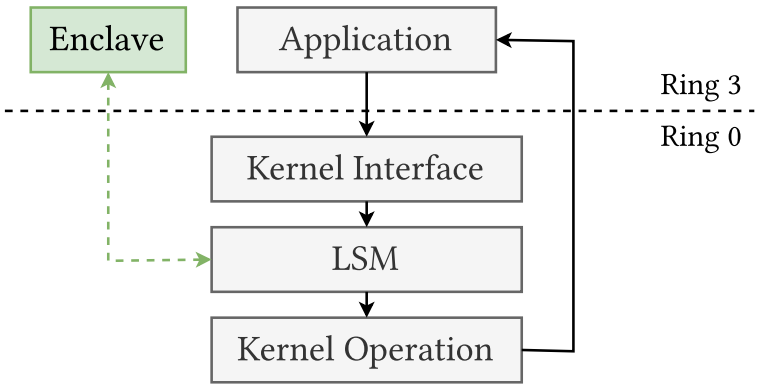
A prototype implementation of an SGX-protected reference monitor for Linux.

Reference monitors must be;

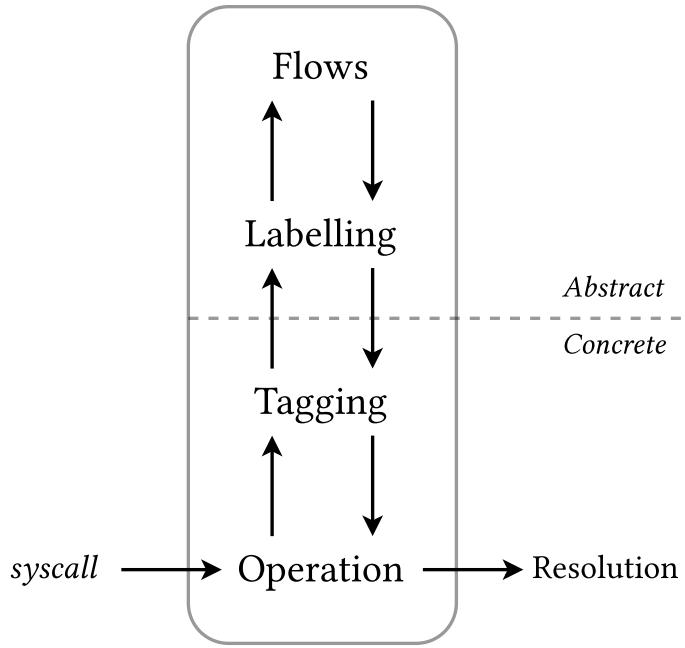
- Always invoked.
- Evaluable.
- Tamper proof.

— *in theory, a perfect use case for SGX.*

Architecture

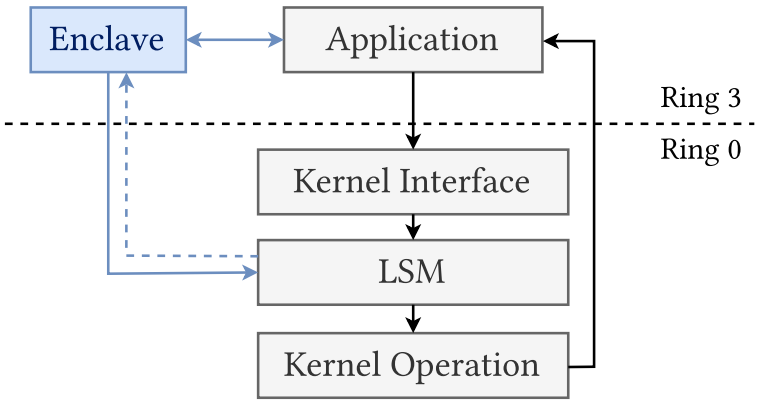


(a) A subfigure

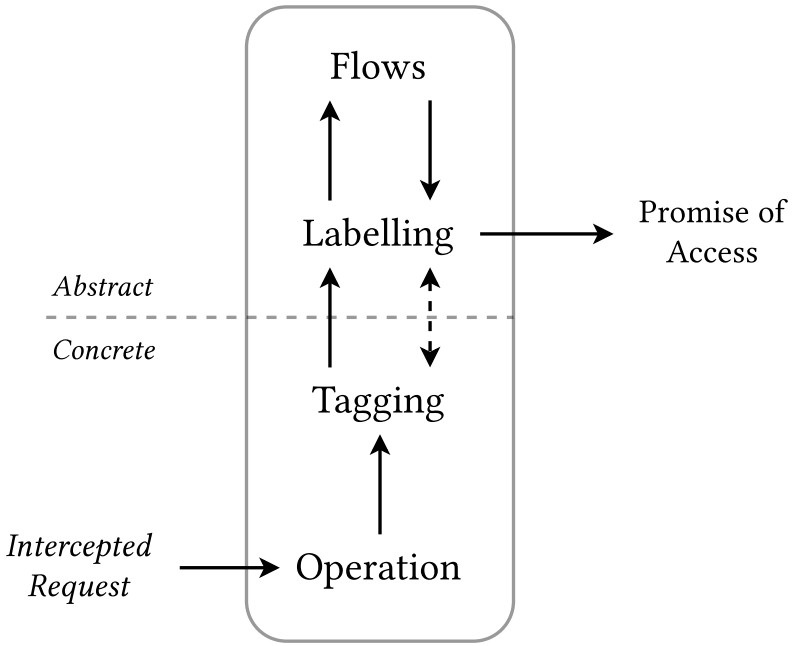


(b) A subfigure

Architecture

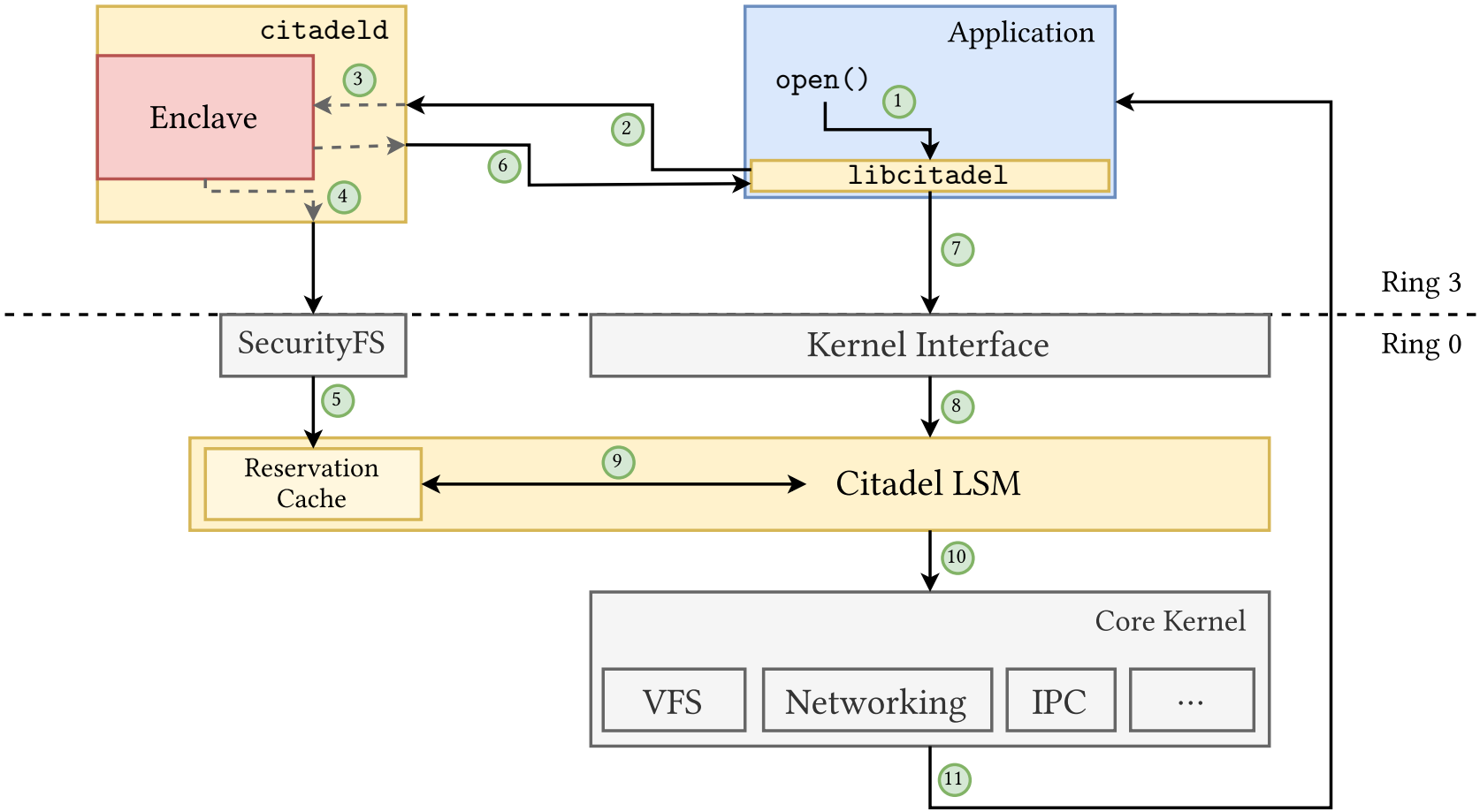


(a) A subfigure



(b) A subfigure

Architecture



Results

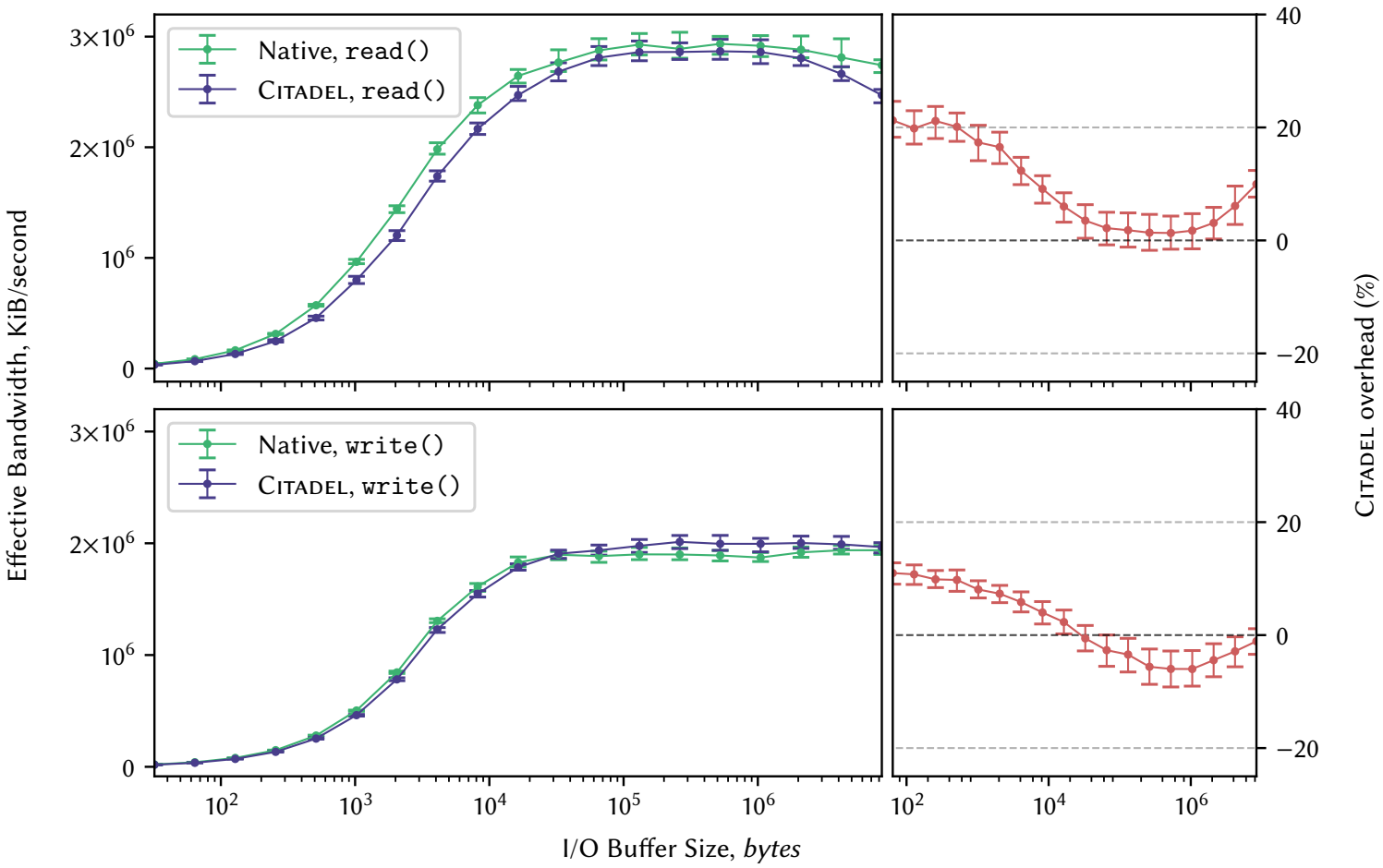


Figure: High level overview of the CITADEL architecture.

Results

- Median *syscall* overhead of $43\mu s$ $(1 - 2\mu s$ amoritsed).
- $20 - 25\%$ effective throughput decrease for IPC.
- Real-world benchmarks using NGINX;
  - Low latency trials: 24% median overhead.
  - High bandwidth file transfers: $\sim 0\%$ median overhead.
- Security characteristics — *promising*.

Conclusion

- Citadel — a modular, enclave-backed reference monitor to securely and verifiably implement IFC methods in the Linux kernel.
- Implemented using enclaves, an LSM, and an auxiliary library for unobstrusive application integration.
- Real-world performance overhead of $20 - 25\%$ observed using Nginx and microbenchmarks.
- Demonstrated the viability of a symbiotic enclave-kernel relationship.

References