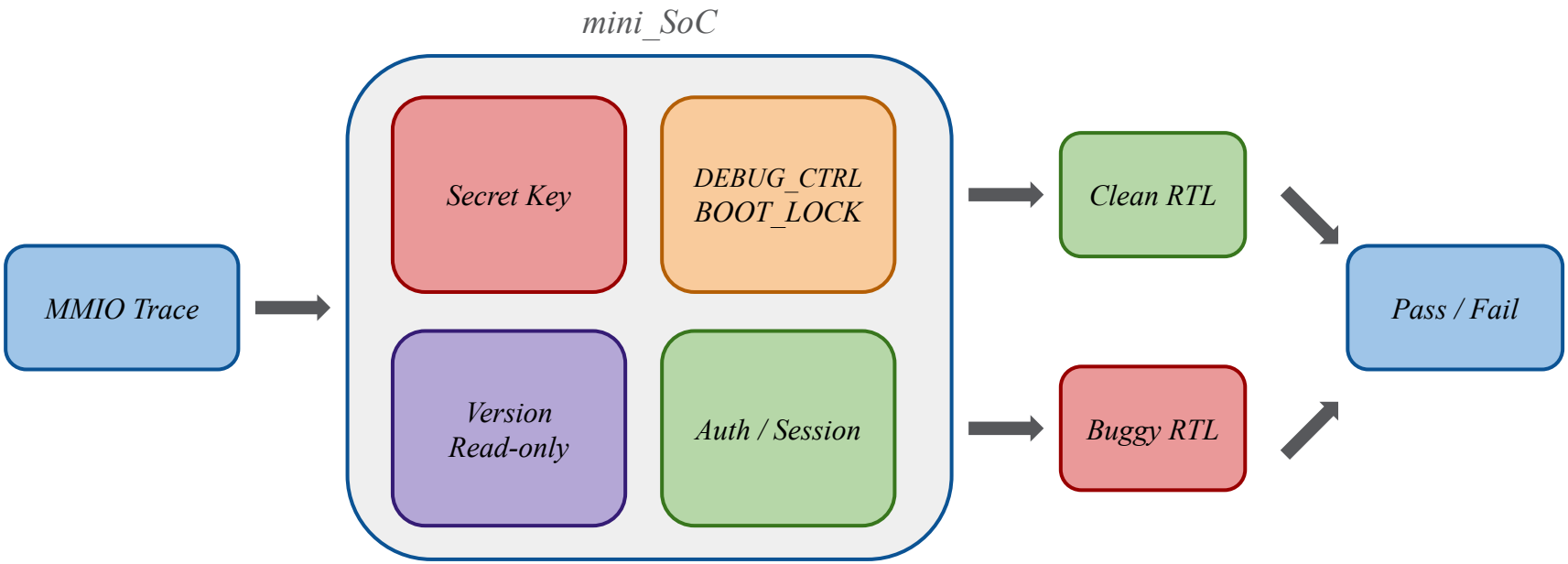
Github link: https://github.com/advaitParanjpe/agentic-rtl-security

Agentic RTL Security Discovery

Can an LLM act like a RTL security validation engineer?

Advait Paranjpe

Controlled Mini-SoC Security Benchmark



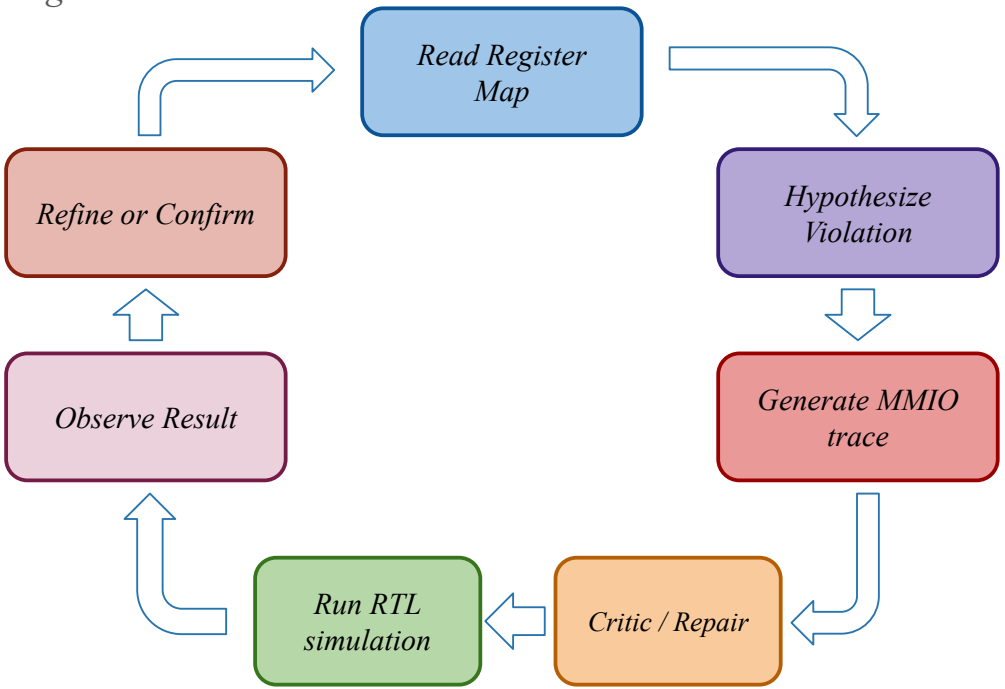
What the traces can check: Privilege check, debug lock, read-only integrity, reserved addresses, session revocation

Discovery Strategies

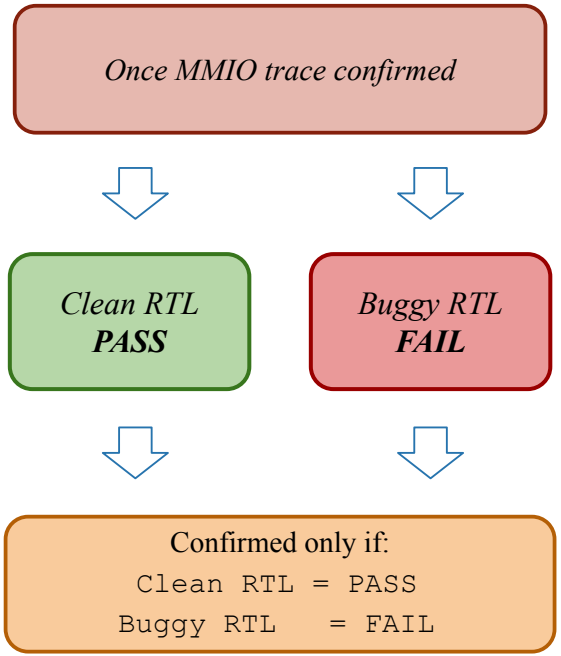
Method	Trace Source	Strength	Limitation
Directed Regression	Hand-written policy traces	Strong baseline	Manual
Random Fuzzing	Random MMIO sequences	Good shallow coverage	Weak on stateful rules
LLM agent	Policy-guided trace proposals	Explainable, state aware	Stochastic, can clean-fail

Agentic Loop: *Propose, Simulate, Refine*

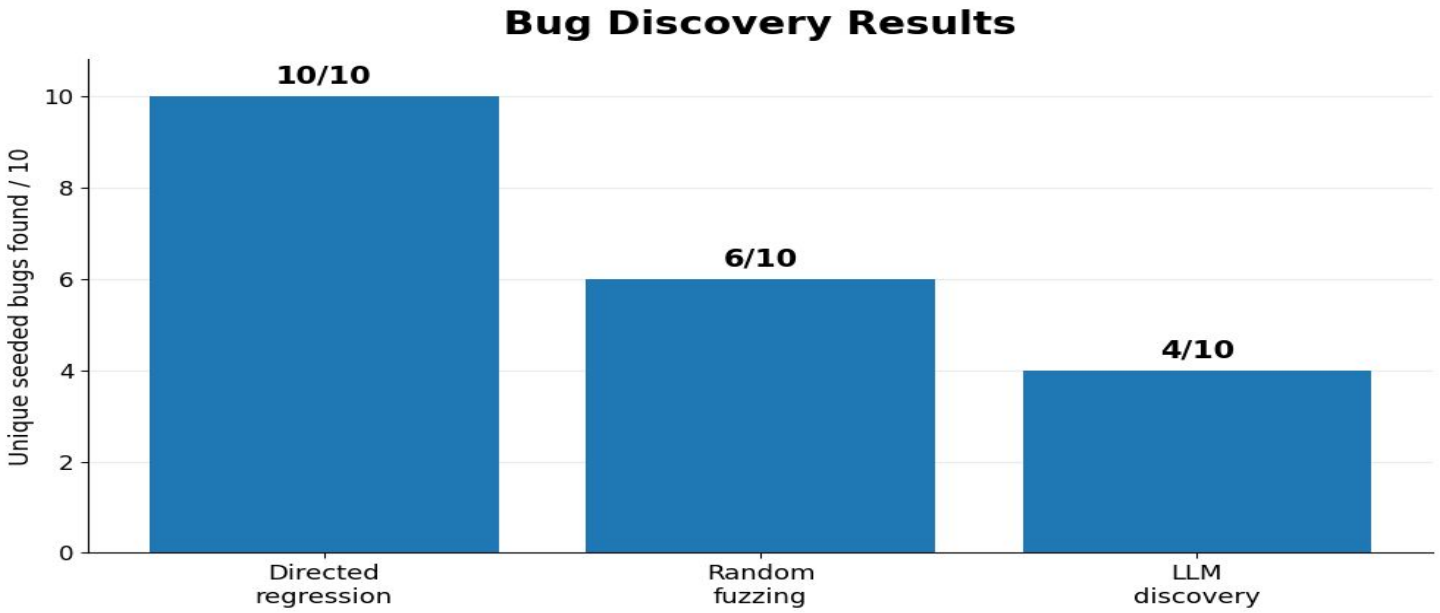
Agentic Part



Oracle-gated validation



Results



Key nuance: LLM found more complex bugs than the random fuzzing did

Takeaways / Future Work

What worked	What needs improvement
Policy-guided trace generation	Better clean-behaviour modelling
Simulator caught bad claims	Stronger coverage planning
Minimised reproducers	Better stateful protocol reasoning