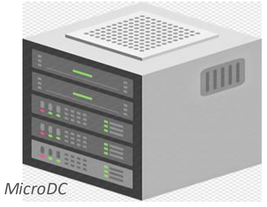
# TCR: A Unified Monitoring Approach to Enhancing the Security and Resiliency of Hazard Workflows on Heterogeneous Infrastructures

PIs: Sudarsun Kannan[1], Ram Durairajan[2], Ulrich Kremer[1], Shiqing Ma[3] Students: Long Tran[1], Veda Vadakalli[1], River Bartz[2]
[1]Rutgers University [2]University of Oregon [3]University of Massachusetts

*Project URL: https://github.com/RutgersCSSystems/hazardmon-ds-public | Transition partners: SAGE Continuum, NDP (National Data Platform)| NSF Award # 2319944*

- **Setting.** Edge cyberinfrastructure (e.g., SAGE/Waggle) deploys sensor-equipped edge nodes alongside on-site MicroDCs in remote locations for near real-time wildfire, flood, and hazard detection.
- **ML at the edge.** Edge MicroDCs increasingly run ML workflows for real-time inference and training similar to traditional datacenters, but on tight power, memory, and trust budgets.
- **Multi-tenancy is the norm.** A single MicroDC hosts diverse workloads: different sensor modalities (wildfire cameras, weather monitoring) and different functional roles (sensor ingestion, data logging, ML inference).
- **Our project.** We are designing PAMS, a Priority-Aware Memory & Security framework. We are transitioning it into SAGE and NDP to deliver multi-tenancy that is performant, sustainable, and secure, without requiring OS kernel changes.

*Edge Node → Gateway → MicroDC*

## Current Deployments

**Lab Testbed**
*Power-instrumented edge boards*

**MicroDC Site**
*Solar-powered, rooftop-deployed compute hub*

**SAGE Waggle Node**
*Sensor-equipped edge*

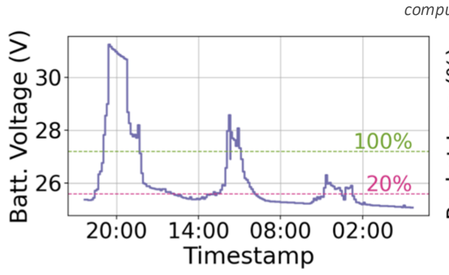
## Cybersecurity Innovation Being Transitioned

### Challenge 1: Managing Resources on Edge Systems

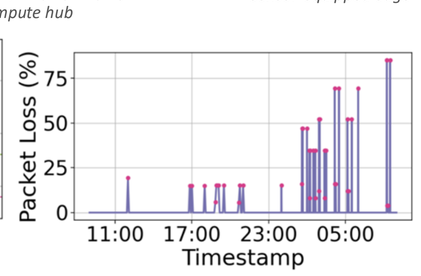
- Edge and MicroDCs run concurrent workflows under tight compute, memory, network, and power budgets.
- Contention arises across CPU, memory, accelerators, storage, and network links.
- Memory contention is one example: heap, page cache, GPU buffers, and lookup tables compete across containers.
- Containers isolate workloads but do not intelligently balance resources across them.
- Priority-oblivious control treats wildfire detection as routine logging.
- Edge runtimes lack integrated, power-aware scheduling.

### Challenge 2: Security on Multi-Tenant Edge Systems

- Malicious or compromised co-tenants may run alongside trusted hazard workloads.
- Memory contention can leak tenant activity through page-cache and heap behavior.
- GPU/CPU side channels could reveal when detection runs and which model is active.
- DoS attacks can inflate memory pressure and evict critical victim data.
- Emergency offloads need attestation to reach trusted tenants.
- App-level priority triggers abused through fake "critical" signals.




**Battery voltage reduction**

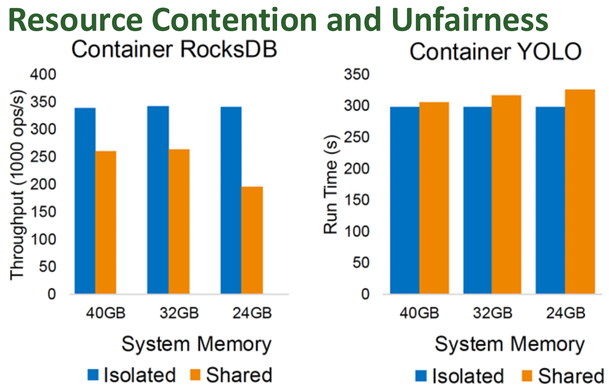



**Network packet loss**

### Setup and Analysis

- Lab testbed: Jetson edge boards + power meters; single-socket Xeon + GPU MicroDC (16-core, 48 GB DRAM, P5000 GPU).
- Workloads: YOLO wildfire inference (heap-heavy), RocksDB sensor logging (page-cache-heavy), adversarial co-tenant.
- Real deployment: SAGE Waggle nodes + MicroDC sites; NDP data fabric for telemetry export.
- Performance metrics: Throughput, tail latency, energy.
- Security metrics: Side-channel info-leakage, DoS resilience, anomaly-detection precision/recall.

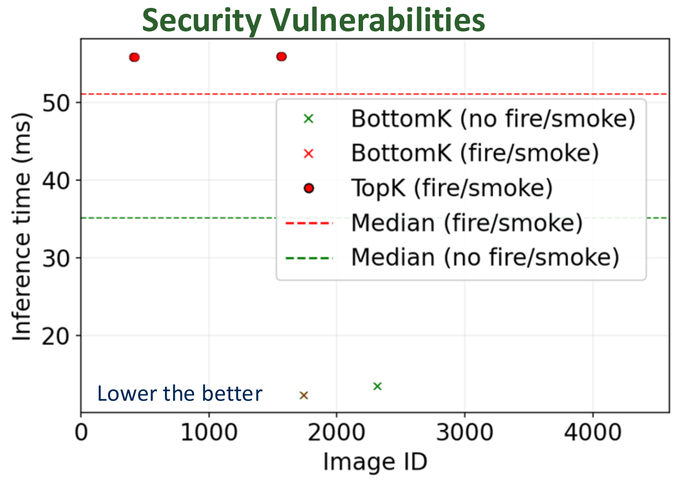
### Resource Contention and Unfairness




***Empirical Evidence***

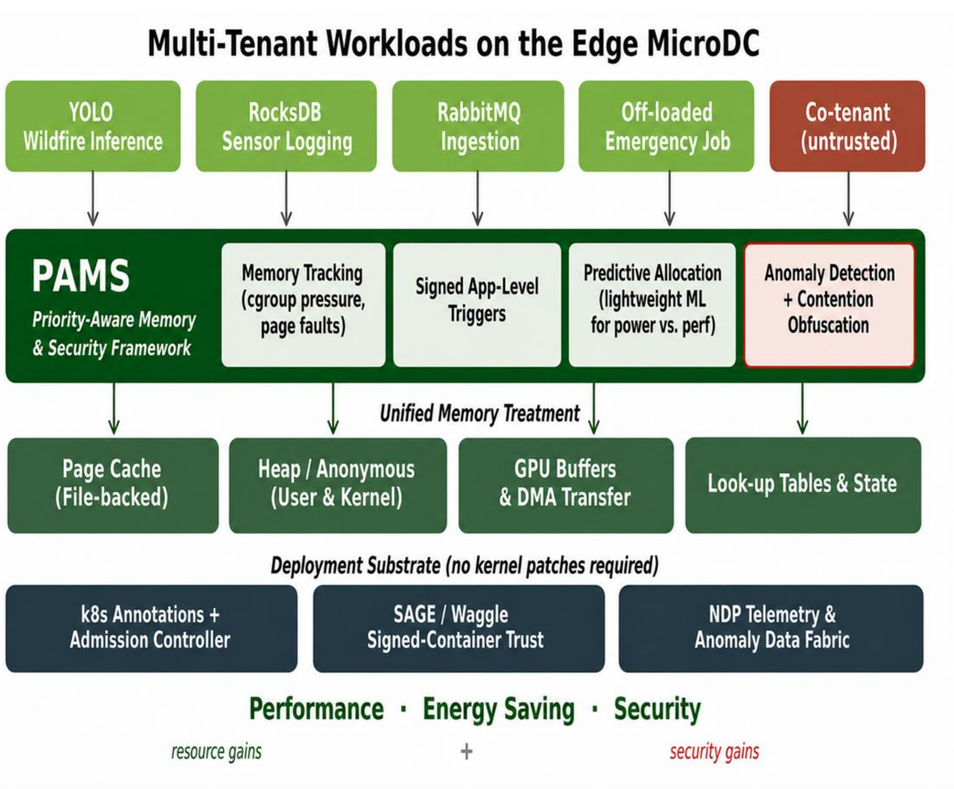
*Memory contention causes asymmetric degradation: RocksDB throughput drops sharply, while YOLO runtime increases under shared execution.*

### Security Vulnerabilities




*Wildfire-detection inference is consistently ~4× slower than no-fire frames, turning timing into a side channel: a co-tenant observing latency can infer when a fire is being detected.*

## Approach: PAMS — Priority-Aware Memory & Security Framework




### Resource and Security Innovations

- **Unified treatment of memory types.** Page cache, heap, GPU buffers, look-up tables as first-class citizens
- **Application-level triggers.** Apps signal pressure via cgroup pressure + page-fault tracking — no kernel patches required
- **Predictive resource allocation.** ML balances performance vs. power, anticipates emergency off-load spikes
- **Signed application triggers**
- **Contention obfuscation** prevent visibility to other tenants during critical fire events such as side channels going dark
- **Anomaly detection from existing signals.**

### Risks vs. Potential for Advances

- **Risks:** ML predictors may fail on unseen workloads; security obfuscation may affect performance.
- **Mitigations:** Userspace-first design with static-policy fallback and open SAGE-style benchmarks.
- **Potential:** Generalizes beyond hazard monitoring and establishes a unified resource-security primitive for edge CI.

### Resulting Benefits and Transition Plan

- Secure multi-tenancy without performance collapse
- Lower static power and embodied carbon through right-sized memory use
- Stronger isolation for shared SAGE and NDP edge nodes
- Security observability from existing resource signals
- Transition to SAGE Waggle nodes, MicroDC sites, and NDP telemetry
- Evaluate with YOLO, RocksDB, RabbitMQ, and adversarial co-tenants
- Measure throughput, latency, energy, carbon, side-channel leakage, and DoS resilience

### Related Publications

- **Are Edge MicroDCs Equipped to Tackle Memory Contention?,** Long Tran, River Bartz, Ramakrishnan Durairajan, Ulrich Kremer, Sudarsun Kannan, ACM HotStorage 2025.
- **Stateful Triage for Reliable and Secure Wildfire Monitoring at the Edge,** Long Tran, River Bartz, Rajesh Sankaran, Ulrich Kremer, Ramakrishnan Durairajan, Sudarsun Kannan, IEEE MILCOM 2025.
- **Leveraging Prefix Structure to Detect Volumetric DDoS Attack Signatures with Programmable Switches,** Chris Misa, Ramakrishnan Durairajan, Arpit Gupta, Reza Rejaie and Walter Willinger, IEEE S&P 2024.
- **Redesigning Edge-Centric Micro-Datacenters for Efficient Multitenancy,** Sudarsun Kannan, River Bartz, Ramakrishnan Durairajan, Uli Kremer, ACM HotInfra 2024.