Pythia: A Just-in-Time Instrumentation Framework for Distributed Systems

Lily Sturmann✩△, Emre Ates†, Peter Portante△, Orran Krieger☆+△, Ayse Coskun†, Raja Sambasivan☆+
Massachusetts Open Cloud☆ / Boston University† / Red Hat△

Unique challenges in debugging distributed systems

Where is the problem? could be in ...
- One of many components
- One of several stack levels, ex:
  - VM vs. hypervisor
  - Application vs. kernel
- Inter-component interactions

Pythia: a framework to **automate** the debugging cycle

- Key insight: Requests that have **similar workflows** in the same system should have **similar performance**. **High performance variation can indicate a problem.**

- Proposed Pythia framework finds where to enable additional instrumentation and **what** instrumentation to enable using ML and statistics techniques.
- Goal: an always-on framework to diagnose problems or send alerts about potential problems in real time.

Proposed Approach

1. Gather current request skeletons
2. Extract critical path from skeletons
3. Group critical path skeletons based on similarity
4. ID groups with high perf. variation
5. Localize perf. var
6. Determine what instrumentation to enable

Pythia Next Steps: Case studies with real problems

- Analyzing traces from OpenStack working “normally” with Pythia
  - Observing baseline values for performance variation
- Recreating known problems in OpenStack: ex. resource contention (please send us your OpenStack problems related to performance degradation!)
- Analyzing traces from OpenStack with problem