KALEIDOSCOPE: LIVE FORENSICS FOR LARGE-SCALE DATA CENTER STORAGE SYSTEMS

Allocation: Exploratory/50 Knh
PI: Ravishankar Iyer[1]
Collaborators: Saurabh Jha[1], Shengkun Cui[1], Tianyin Xu[1], Jeremy Enos[2], Mike Showerman[2], Greg Bauer[2], Mark Dalton[3], Zbigniew Kalbarczyk[1], Bill Kramer[2]

[1]University of Illinois at Urbana–Champaign
[2]National Center for Supercomputing Applications
[3]Cray Inc.

EXECUTIVE SUMMARY

The research team has developed Kaleidoscope, an innovative system that supports live forensics for application performance problems caused by either individual component failures or resource contention issues in large-scale distributed storage systems. The design of Kaleidoscope was driven by the team's study of I/O failures observed in a petascale storage system.

Kaleidoscope is built on three key features: (1) using temporal and spatial differential observability for end-to-end performance monitoring of I/O requests, (2) modeling the health of storage components as a stochastic process using domain-guided functions that account for path redundancy and uncertainty in measurements, and (3) observing differences in reliability and performance metrics among similar types of healthy and unhealthy components to attribute the most likely root causes.

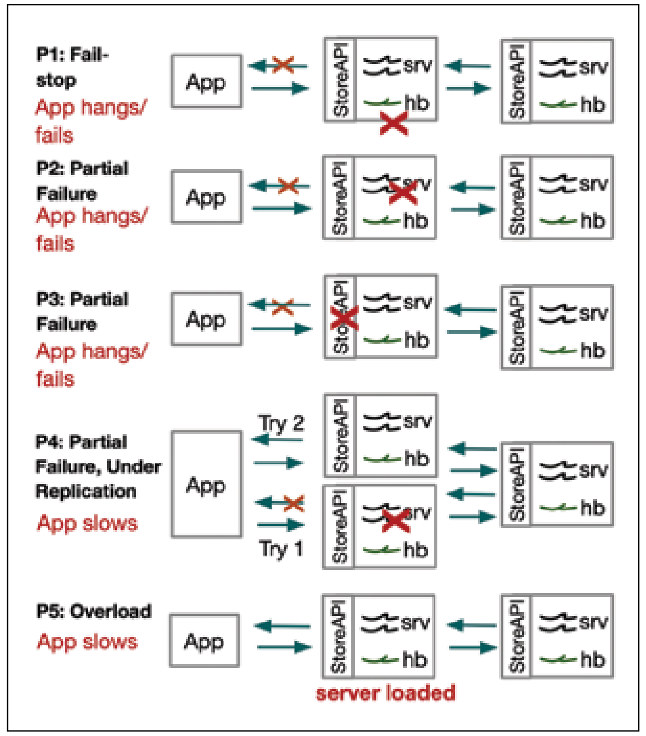

Figure 1: Common patterns of I/O failures. Notation: "hb" is heartbeat process, "srv" is service process; each box represents the storage components (*e.g.*, data servers).

The research team deployed Kaleidoscope on the Cray® Sonexion®; an evaluation showed that Kaleidoscope can run live forensics at five-minute intervals and pinpoint the root causes of 95.8% of real-world performance issues, with negligible monitoring overhead.

RESEARCH CHALLENGE

Large-scale storage services are typically implemented on top of clusters of servers and disk arrays to provide high performance (*e.g.*, load balancers and congestion control) as well as high availability (*e.g.*, RAID, and active–active high-availability server pairs). Component failures and resource contention are chronic problems that lead to I/O timeouts and slowdown in such systems. State-of-the-art solutions focus on reliability failures and, hence, do not attempt to distinguish between resource contention and component failures in storage systems, as highlighted in Fig. 1. Knowing whether a problem is due to resource contention or component/node/subsystem failure is critical in effectively coordinating a recovery strategy.

A combination of component failures and contention issues significantly degrades application performance in production settings. This project uses a mixture of proactive monitoring and machine learning to jointly address the above issues. The team has incorporated the proposed techniques into an automated tool called Kaleidoscope. This tool has been demonstrated in live traffic on a production system to: (1) locate components such as data servers and RAID devices causing I/O bottlenecks such as I/O slowdown or timeouts, (2) differentiate between a reliability failure and a resource contention issue, and (3) quantify the negligible impact on system performance while delivering high precision and recall.

METHODS & CODES

The research team used two years of production data in excess of one terabyte from Blue Waters including system-generated storage error logs and store read/write latency logs collected intelligently using Kaleidoscope. Kaleidoscope is a generic framework for supporting runtime detection and diagnosis of large-scale storage systems. The key components of the tools are:

- Proactive monitoring. Kaleidoscope monitors the end-to-end performance of a storage system using Store-Pings, a set of monitor primitives that covers all the storage operations involved in serving a client's I/O requests (*e.g.*, creating, reading, writing, and deleting files). Store-Ping monitors are strategically placed to provide both spatial and temporal differential observability in real time (steps 1–3 in Fig. 2).
- Modeling and inferring component health. The health of a component in a storage system such as a metadata server or a RAID device is modeled as a stochastic process that accounts for uncertainty (owing to performance variability and asynchrony) as well as nondeterminism in distributed storage systems. The research group built a system model by using the factor graph (FG) formalization, which infers component health by ingesting the monitoring data collected by Store-Pings. The inference on the model allows Kaleidoscope to localize unhealthy components in near real time (step 4 in Fig. 2).
- Methods to determine the cause of I/O failures. A set of statistical methods (including a local outlier factor algorithm run using data on server load, disk load, and disk bandwidth utilization) and clustering of storage system error logs are used to distinguish between component failures and resource overloads. The statistical methods are based on a comparison of reliability and performance metrics (such as the number of active processes on a data server) as they are collected for healthy and unhealthy components. Note that the distinction between healthy and unhealthy components is provided by the FG-based model (Step 5 in Fig. 2).

RESULTS & IMPACT

Deployment. Kaleidoscope has been deployed on Blue Waters' Cray® Sonexion®, a 36-petabyte production system that employs the Lustre file system. Lustre is used by more than 70% of the top 100 supercomputers and is offered by cloud service vendors such as Amazon and Azure. Its design resembles that of many other object-based POSIX storage systems such as the IBM GPFS, BeeGFS, Ceph, and GlusterFS. The team measured the overhead introduced by Store-Ping monitors on the production system and found the overhead to be less than 0.01% on the peak I/O throughput of the Cray® Sonexion®.

Forensic effectiveness. The evaluation was based on 843 production issues identified and resolved by Blue Waters system managers in a two-year period as the ground truth. Overall, Kaleidoscope correctly localized the component failures and resource overloads for 99.3% of the cases. In addition, Kaleidoscope accurately identified the likely root cause for 95.8% of the cases, *i.e.*, disambiguation between resource contention and component failures.

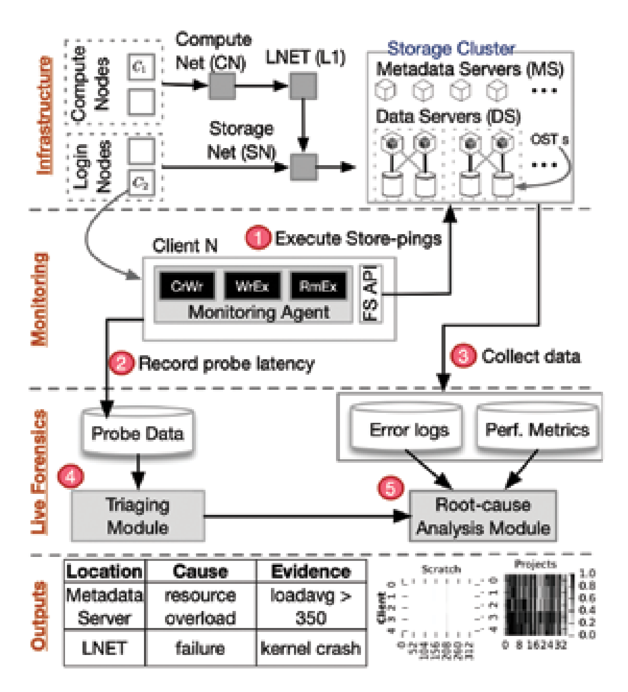

Figure 2: An overview of Kaleidoscope. Kaleidoscope consists of three components for monitoring failure localization and failure diagnosis (marked in gray).

WHY BLUE WATERS

Blue Waters is one of the few open-science capacity systems that provides a testbed for scaling computations to tens or hundreds of thousands of cores on CPUs and GPUs. It also enables the study of failures and performance degradations of applications in production petascale systems, thereby allowing researchers to understand the performance–fault–tolerance continuum in high-performance computing systems.

PUBLICATIONS & DATA SETS

S. Jha *et al.*, "Measuring congestion in high-performance datacenter interconnects," to be presented at *17th USENIX Symp. Networked Systems Design Implement.*, Santa Clara, CA, U.S.A. Feb. 25–27, 2020.

S. Jha *et al.*, "A study of network congestion in two supercomputing high-speed interconnects," presented at *26th IEEE Annual Symp. High-Performance Interconnects (HOTI)*, Santa Clara, CA, U.S.A. Aug. 14–16, 2019.

S. Jha *et al.*, "Live forensics for distributed storage systems," submitted, 2019, arXiv: https://arxiv.org/abs/1907.10203v1.