HPC State Anomaly Detection and Visualization with SaNSA

Megan Hickman Fulp - Coastal Carolina University

Mentors: Dr. Nathan DeBardeleben (LANL), Prof. William M. Jones (CCU)

Motivation

Goal: Run various statistics on SaNSA data to gather information that will assist in finding anomalous nodes and correlated failures.

Background

- SaNSA (Supercomputer and Node State Architecture) is a tool designed to help users visualize HPC states.
- After ingesting system and scheduler events from a machine, the following metrics are calculated:
  - Time in state
  - Number of events per state
  - Percent of time spent in state
- Since events from 26 different states are being captured from all nodes on a given machine, datasets can become very large. *Apache Spark* and *Elasticsearch* are utilized to perform calculations on these large datasets with relatively low overhead.

Scheduler vs. Hardware Conflict

- Events from different sources occasionally contradict one another due to their hardware or scheduler perspective.
- In the example below:
  - The scheduler lost connection to the node 3 minutes before the node hardware was considered “down.”
  - Likewise, the node hardware was declared “up” 9 minutes before the scheduler was able to reconnect.

State	Timestamp
SLURMCTLD: Not_Responding_Setting_Down	2019-01-01T03:15:44.000-07:00
QSTATS: DOWN	2019-01-01T03:18:55.000-07:00
SYSTEMD: Startup_Finished	2019-01-01T03:19:50.000-07:00
SLURMCTLD: Now_Responding	2019-01-01T03:28:55.000-07:00

User-centric view: It does not matter that a node is up if it is unreachable & unable to be scheduled by the resource manager.

Node-reliability view: It seems incorrect to count the node as down if the hardware was up at the time and some other system, like a software timeout, kept it from being accessible.

Both views have been incorporated into SaNSA’s data. This distinction is crucial when analyzing calculated results.

Anomaly Detection

Process

After calculating the average percentage of time spent in each state for all nodes, we can pick out nodes that have spent an anomalous amount of time in a certain state.

This is accomplished by calculating the z-score for percent of time spent in a state. Z-score is defined as the number of standard deviations a data point is from the mean.

$$x = raw\ score, \quad \mu = mean, \quad \sigma = standard\ deviation$$

$$\text{z Score} = (x - \mu)/\sigma$$

$$\text{z Score} < 1 \rightarrow not\ anomalous$$

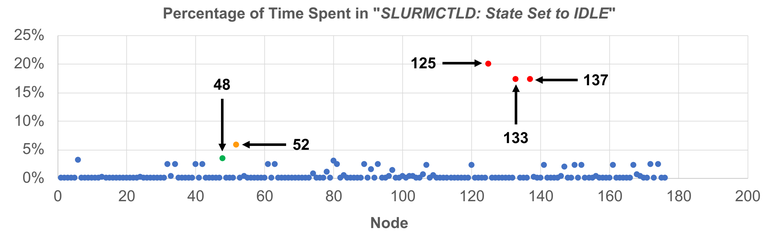
$$1 < \text{z Score} < 2 \rightarrow low\ severity$$

$$2 < \text{z Score} < 4 \rightarrow medium\ severity$$

$$4 < \text{z Score} \rightarrow high\ severity$$

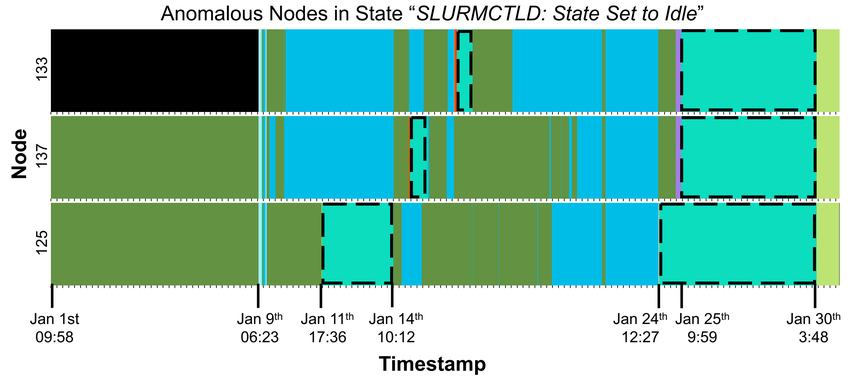
In the following example, one month of Woodchuck data is analyzed to find all nodes that spent an anomalous percentage of time in the *SLURMCTLD: State_set_to_IDLE* state.

Node	State	Percent in State	Z-Score	Severity
48	SLURMCTLD: State_set_to_IDLE	3.42%	1.1028	Low
52	SLURMCTLD: State_set_to_IDLE	5.73%	2.0350	Medium
133	SLURMCTLD: State_set_to_IDLE	17.20%	6.6710	High
137	SLURMCTLD: State_set_to_IDLE	17.20%	6.6712	High
125	SLURMCTLD: State_set_to_IDLE	19.94%	7.7797	High



Once the anomalous nodes have been identified, their events can be visualized. The following visualization can assist in analyzing node-state over time, finding the cause of the anomaly, and determining correlated failures.

State Visualization of Anomalies



Legend

UP	IDLE	DOWN	DRAIN	DT
SLURMCTLD: Now Responding	SYSTEMD: Startup Finished	QSTATS: MAINT	SLURMCTLD: State set to DRAIN	DST: Start
RP: Successfully Inserted Node	SLURMCTLD: State set to IDLE	SLURMCTLD: State set to MAINT	SLURMCTLD: State set to DRAINING	DST: End
KERNEL: Kernel Command Line	SLURMCTLD: State set to ALLOCATED	NTPD: Signal 15 Exit	QSTATS: DRAIN	Quick DST: Start
SLURM: Job Running	SLURM: Job Finished	SLURMCTLD: State set to DOWN	QSTATS: DRAINING	Quick DST: End
		QSTATS: DOWN		DAT: Start
		KERNEL: Kernel Panic		DAT: End
No Data		SLURMCTLD: Not Responding Setting Down		

Future Work

- Classify and plot events by their state category.
  - UP, IDLE, DOWN, DRAIN, DT
- After adding node topology data, positional correlations between nodes can be found.
- These results come from one month of Woodchuck data (a few hundred nodes). In the future, SaNSA will be used on 6 months of Grizzly (~1,500 nodes) data and eventually Trinity data (20,000 nodes).

Ultrascale Systems Research Center

LA-UR-19-27051