

Root Cause Analysis

Hazelcast Related Outage



The following is a detailed accounting of the service outage that Agile Central users experienced on <date> at Jan 19, 2016 8:52am.

Root Cause Analysis Summary

Event Date	Jan 19, 2016
Event Start	8:52am
Downtime Start Time	8:52am
Time Detected	8:55am
Time Resolved	9:00am
Downtime End Time	9:00am
Event End Time	9:00am
Root Cause	Hazelcast problem uncovered by poor performing user query. The user query cause high load on a single app server, and user threads on other app servers backed up trying to communicate with the highly loaded server over hazelcast.
Duration	Total Downtime: 8 minutes Total Impaired Availability: N/A Time to Detect: 3 minutes Time to Resolve: 5 minutes

Future Preventative Measures

Actions that should be taken to prevent this Event in the future.

Actions	Description
Turn on Session Replication again	After fixing the Hazelcast issues

Work with Customer to avoid that query	Reached out to customer to work on custom app
Can we limit ad hoc query damage to systems?	To bring up to Ops and the Architecture Council for consideration of a feature that can limit a user's ability to severely degrade system performance in these situations
Detecting ad hoc query hammering	Looking at a way to find out via Splunk query when a user is hammering ALM with queries
Any additional monitoring / alerting to detect faster?	Might require tuning of Nagios. Look at alerts to see if we can tweak to detect faster
Investigate Hazelcast timeout	Look to a shorter timeout than the 60 second default