Main page
Recent changes
Server admin log (Prod)
Server admin log (RelEng)
Deployments
SRE/Operations Help
Incident status

Cloud VPS & Toolforge
Cloud VPS documentation
Toolforge documentation
Request Cloud VPS project
Server admin log (Cloud VPS)

Tools
What links here
Related changes
Special pages
Permanent link
Page information
Cite this page

Print/export
Create a book
Download as PDF
Printable version

Incident documentation/20170223-AQS

< Incident documentation

Summary

As final step of the Analytics AQS Cassandra cluster, T157354 was meant to apply a non invasive change to the system_auth Cassandra keyspace to increase the replication factor from 6 to 12. The procedure did not proceed as planned since after the first execution of nodetool repair the aqs user was wiped out from some Cassandra instances, ending up in authentication errors for most of the HTTP queries coming from the Restbase AQS layer. The procedure was completed to avoid data consistency errors and the aqs user was re-created manually to force its replication to all the instances, ending the outage.

Timeline

This is a step by step outline of what happened to cause the incident and how it was remedied.

- 2017-02-23 09:37 - executed the following command on the aqs1004-a Cassandra instance: `ALTER KEYSPACE "system_auth" WITH REPLICATION = {'class': 'SimpleStrategy', 'replication_factor': '12'};`
- 2017-02-23 09:38 - executed the following command on aqs1004-a: `nodetool-a repair system_auth`
- 2017-02-23 09:39 - start of the 50X for the AQS API
- 2017-02-23 09:43 - first alarms showing in the #wikimedia-operations IRC channel, multiple opsens aware of the issue and working on it (elukey, godog). The mobile apps endpoint in eqiad seems to be in trouble as well due to failures to contact the AQS API.
- 2017-02-23 09:54 - from Logstash it is clear that it is an authentication problem between AQS Restbase and Cassandra - "*User aqs has no SELECT permission on <table local_group_default_T_pageviews_per_article_flat.data> or any of its parents*". The following cqlsh query was returning inconsistent results: `select * from system_auth.roles`
- 2017-02-23 09:56 - decision taken by elukey to proceed with the remaining nodetool repair actions on the rest of the Cassandra instances as an attempt to fix the inconsistency (11, taking 3/4 minutes each, to be executed sequentially).
- 2017-02-23 10:43 - the following command was executed on aqs1004 to recreate the aqs user: `CREATE USER IF NOT EXISTS aqs WITH PASSWORD 'thepassword' NOSUPERUSER;`
- 2017-02-23 11:03 - complete recovery

Conclusions

What weakness did we learn about and how can we address them?

We still don't have an exact root cause. The Cassandra documentation seems consistent with the procedure followed but for some reason system_auth users/roles were wiped in the process. There are some remarks to highlight:

- We don't run periodical nodetool repair actions in AQS (because not really needed), so the one that caused the outage was probably the first executed on the new AQS cluster (we have deprecated aqs100[123] running on rotating disks with aqs100[456789] running on SSDs). Since AQS now runs Cassandra 2.2, the repair was incremental by default and there might be some dark corner cases if the first incremental repair for a keyspace is executed after an alter statement.

- Replicating the system_auth keyspace to the number of Cassandra instances seemed to be an old recommendation that is not present anymore in the current Datastax guidelines. It is only mentioned to avoid it to be set at 1.
- Increasing the replication of a keyspace on a live cluster can lead to read failures while the repair actions are ongoing, so we should have followed a safer path (for example, disable authentication completely, make the change, re-enable authentication).
- We have permission_validity_ms set to 10 minutes to cache credentials on each node, it might have exacerbated the problem.

Actionables

- Review Restbase system_auth replication and update it if necessary following a safer procedure - T158908
- Document best practices for increasing system_auth replication factor on Wikitech and its pitfalls.

Category: Incident documentation