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/20170831-Zookeeper

< Incident documentation

Contents [hide]
1 Summary
2 Timeline
3 Conclusions
4 Actionables

Summary

From 15:18 to 15:52 UTC the Zookeeper Main codfw cluster was down (not able to accept any new data or respond to clients) due to a firewall rule preventing the nodes of the cluster to communicate between each other via their network ports.

Background: https://gerrit.wikimedia.org/r/#/c/366228/ was merged in July 2017 as part of https://phabricator.wikimedia.org/T114815. Zookeeper ports were open to the whole Prod network and a more selective access control was needed, especially in light of the fact that the new Kafka Topics ACLs that the Analytics team will deploy will be stored in Zookeeper. The main flaw of this code review was that while all the clients were carefully reviewed and whitelisted in the Ferm rules, the Zookeeper cluster nodes were not, meaning that they would have been unable to communicate between each other via their ports (2181 2182 2183). The new Ferm rules were applied only to new TCP connections, leaving the long running ones open. When the new Ferm rules were deployed (incrementally node by node) nothing strange was registered, and the task was closed accordingly, without realizing that any simple restart of the Zookeeper daemons would have generated new connections that would have been blocked by Ferm.

As part of the last openjdk-7 security update Luca had to restart all the Zookeeper daemons, and when he reached 2 out of 3 nodes the cluster was not able anymore to make any progress in its distributed consensus algorithm.

Impact: While the Zookeeper cluster was down, all its clients were unable to read/write anything to it. The major impact was registered to MirrorMaker, a Kafka old consumer still using Zookeeper to store its offsets. We use it to mirror topics among Kafka clusters, and it stopped working in the timeframe (but didn't loose any data). The Kafka main codfw cluster, used by Eventbus and hence by ChangeProp, is a client of the Zookeeper main codfw cluster but it did not stop to accepting messages, since it was probably only in a degraded status (not able to store new Topic metadata changes to Zookeeper for example).

Timeline

2017-07-19 - https://gerrit.wikimedia.org/r/#/c/366228/ was merged to tighten the access to the main Zookeeper clusters in eqiad and codfw.

2017-08-31T13:09 - Zookeeper on conf2001 restarted (2 out of three nodes in the cluster able to communicate, cluster available).

2017-08-31T15:18 - Zookeeper on conf2002 restarted (2 out of three nodes in the cluster not able to communicate, cluster down). From this moment Kafka main codfw (kafka200[123]) and MirrorMaker were unable to communicate with Zookeeper.

2017-08-31T15:52 - Luca stops ferm on conf200[123] nodes via Cumin, cluster back into available mode after few seconds. MirrorMaker starts to work again from the last offset committed. Kafka main codfw is able to report Topic metadata changes to Zookeeper.

2017-08-31T16:31 - Permanent fix (https://gerrit.wikimedia.org/r/#/c/375023/) merged and rolled out to all the conf[12]00[123] nodes.

Conclusions

A couple of big lessons learned:

- New ferm rules are applied to new connections, so it is not sufficient to simply deploy them to be sure that they are working properly as the author expects.
- Zookeeper offers commands to check its state, and they have subtle differences. The command "**ruok**" only checks if the daemon is up and running, it doesn't report anything about the cluster status, meanwhile "**stats**" reports information about clients connected and role of the node (follower/leader). After updating conf2001 Luca only checked "**ruok**" and not "**stats**" (or the Zookeeper logs), deciding to proceed with conf2002 because nothing weird was reported in two hours. A simple additional check would have avoided the outage, since two out of three nodes would have been enough to guarantee Zookeeper availability.

Actionables

[DONE] - Document this outage in Service restarts in order to prevent more people to step into this issue.

Category: Incident documentation