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/20190910-toolforge-kubernetes

< Incident documentation

document status: draft

Tracked in Phabricator
Task T232536

Contents [hide]

Summary

Kubernetes API server process failed to start, causing the collapse of many Kubernetes managed services and disabling launching or managing of containers for tools while the server was down.

Toolforge Kubernetes API services were down for several hours due to puppet changes that caused a CA certificate mismatch with the etcd servers. Since it was precipitated by a restart of the service, the root cause took significant time to find and correct. While it was down, Toolforge webservices using Kubernetes would fail to launch or restart and likely many services failed while the API server was not responding.

Impact

Toolforge Kubernetes API server was unresponsive which caused the cluster to appear completely down for all monitoring and use purposes. Users of Toolforge could not launch tools with the `webservice` command with the kubernetes backend or use kubectl to check status or manipulate their tools. A larger impact was caused when kube2proxy was unable to watch for Kubernetes changes. It then "started from scratch" and crashed. Since prometheus shows that the containers (and thus the tools themselves) were unimpacted, this seems to have taken all Kubernetes webservices offline at the proxy level.

Detection

The first notice was icinga alerting on the check against http://checker.tools.wmflabs.org/k8s/nodes/ready was our first notice.

Timeline

- 18:30 maintain-kubeusers process restarts the Kubernetes API server, which fails to come back up -- kube2proxy also fails here
- 18:45 Alarms start going off for the toolschecker icinga check around "All k8s nodes healthy"
- 18:55 It is theorized by Bryan and Andrew that puppet certs could somehow be involved, but this uses project internal certs, which complicates the possibility.
- 18:59 Bryan tries restarting the kube-apiserver process (which fails). Puppet certs are possibly written off because of the timing of the outage as well because those were changed the night before, not now.
- 19:01 Bryan notices the log message indicating the ultimate failure is: "Failed to list *storage.StorageClass: client: etcd cluster is unavailable or misconfigured"

- 19:03 The team decides to be Brooke involved and contacts her.
- 19:07 Brooke notes that an etcd healthcheck reports success (also checks flannel in case there's some overall etcd issue--it's fine)
- 19:10 Brooke reboots the k8s master in case that helps here.
- 19:17 As Brooke comes up to speed on what's going on and has been done, she validates that the etcd database is all there, just in case.
- 19:19 An annoying behavior of the etcdctl command of the old version used on the tools-k8s-etcd servers is that it times out in 1 second. This fools the team into thinking the service is flapping or downward spiraling.
- 19:33 Brooke validates that the etcd and k8s certs never changed during puppet changes.
- 19:44 Brooke reproduces the same error message using etcd and in the etcd logs and wrongly presumes that the issue is in etcd because of this.
- 19:48 Toolforge homepage alerts as down
- 19:54 Because timeouts have been noticed, Brooke starts checking firewall stuff. The changes to the firewall don't line up with the outage and are about new puppetmasters, though.
- 19:55 Hieu starts working with Brooke on the etcd servers. They notice that timeouts do show up in the etcd logs as well as frequent raft elections.
- 20:18 Brooke reboots etcd cluster in case that might stop the churn
- 20:30 Brooke reboots the k8s master server again just to restart all services there.
- 20:50 Brooke stops etcd on tools-k8s-etcd-01 and takes a backup.
- 21:06 With a backup in place, the team reduces the cluster down to 1 node in order to elminate raft election flaps as a cause.
- 21:09 Hieu enables debugging on etcd.
- 21:24 Hieu notices that the etcdctl tool has tricked us into thinking that etcd was dropping off when really it was just slower than that tool's default timeouts.
- 21:25 The team begins trying to return etcd to its original size.
- 21:52 Brooke adds back the first node after checking a few things and discovers that she made a mistake. This collapses the cluster (because in a 2 node etcd cluster, both nodes have to work).
- 22:09 After several experiments trying to get that node out of the cluster, Jason offers to take over for Brooke to get some rest (she was sick today).
- 00:25 Jason supports Brooke (who failed to log out anyway) with technical advice and general idiot-checks until she gets etcd working again at full capacity. This required several experiments, but eventually called for a full disaster recovery from the backup (TODO: document this).
- 00:26 Jason and Brooke resume combing over the API server to find what changed or what is not working.
- 01:02 Brooke and Jason notice that the CA certificate did change the night before in some way that was hard to determine.
- 01:05 Brooke uses curl to highlight a reproducable error with CA trust on SSL calls from the kube API server that does not happen between etcd servers.
- 01:12 Hieu has rejoined the effort and starts looking for options to specify a CA in for etcd communication in case that's useful.
- 01:20 Alex Monk shows up and helps with methodically comparing the CA that might be used by the two servers.
- 01:28 After analyzing things a bit, Brooke swaps the CA with the one from the etcd servers at Alex's suggestion. This works and restores communication and services.
- 01:31 Recovery alerts noted.
- 01:33 Brooke finds that puppet reverts the change and breaks things again.
- 01:36 Alex provides the possibility that we are running into the situation described here: https://phabricator.wikimedia.org/T148929#2817428
- 01:43 Based on the information and direction of Alex, Brooke places the etcd puppet CA cert in /var/lib/puppet/client/ssl/certs/ca.pem, which puppet picks up and fixes things in a stable fashion. Outage over.

Conclusions

"Typically badly-behaved things" like etcd confused the solutions here. Puppet certs are dangerous. Very old Kubernetes versions have terrible log errors.

What went well?

- automated monitoring detected the incident

- team worked together
- things recovered quickly once the problem was solved

What went poorly?

- team made things worse by trying to fix the wrong thing (etcd instead of CA certs)
- Brooke was sick and was not thinking very clearly
  - cross training might have helped here (so Brooke wouldn't need to do as much), but this really needed some collaboration to solve anyway
- the logs on tools-k8s-master-01 are too verbose and kind of awful
- it took quite a while to find the fix due to multiple factors obscuring it
  - timing of the event (which was precipitated by maintain-kubeusers restarting the api server--thus forcing a reconnect to etcd instead of during the puppet changes)
  - long-standing quirks in our etcd cluster that aren't documented
  - no DR procedures or node-replacement procedures currently documented for etcd

Where did we get lucky?

- Alex Monk showed up with key information toward the end

How many people were involved in the remediation?

- 4 SREs were helping in one way or another, 1 team manager and 1 technical volunteer

Links to relevant documentation

https://wikitech.wikimedia.org/wiki/Portal:Toolforge/Admin/Kubernetes

Actionables

NOTE: Please add the #wikimedia-incident Phabricator project to these follow-up tasks and move them to the "follow-up/actionable" column.

- Document etcd cluster processes Task T232769
- Upgrade Toolforge Kubernetes (already underway) to improve etcd and Kubernetes error reporting -- and also not require restarting the API server to add a tool Task T214513
- Attempt to make kube2proxy more resilient to API server failures Task T232770
- Audit puppet CA certs. There really shouldn't be more than one within the tools project. Task T232772
- Make Kubernetes control plane HA (already in PoC in toolsbeta, so the task is closed and waiting on the full upgrade deployment) Task T142862

Categories: Incident documentation | Incident documentation drafts