Closed (moved) Opened 2 years ago by Ilya Frolov

Routing outage 2017-09-13

Context

At 18:51 we detected increased 500 errors followed by full outage.

Slack log: https://gitlab.slack.com/archives/C101F3796/p1505328698000244

Graphs of the incident: https://performance.gitlab.net/dashboard/db/fleet-overview?orgId=1&from=1505328503598&to=1505330095434

Timeline

On date: 2017-09-13

- 18:51 UTC - people reporting 500ies in slack, we start investigating the issue
- 18:55 UTC - pagerduty starts calling
- 19:02 UTC - we're seeing no routes between frontend LBs and the rest of the fleet
- 19:07 UTC - we're rebooting HAP boxes as the only viable option
- 19:08 UTC - connectivity is back after reboots
- 19:14 UTC - service is fully online

Incident Analysis

- How was the incident detected?

People reporting in slack, blue-moon and pager duty.

- Is there anything that could have been done to improve the time to detection?

TBD

- How was the root cause discovered?

TBD, we're opening a ticket with upstream provider.

- Was this incident triggered by a change?

No.

- Was there an existing issue that would have either prevented this incident or reduced the impact?

No.

Root Cause Analysis

Rest is TBD after we receive info from upstream provider.

Follow the the 5 whys in a blameless manner as the core of the post mortem.

For this it is necessary to start with the production incident, and question why this incident happen, once there is an explanation of why this happened keep iterating asking why until we reach 5 whys.

It's not a hard rule that it has to be 5 times, but it helps to keep questioning to get deeper in finding the actual root cause. Additionally, from one why there may come more than one answer, consider following the different branches.

A root cause can never be a person, the way of writing has to refer to the system and the context rather than the specific actors.

For Ex:

At 00:00 UTC something happened that led to downtime

- Why did X caused downtime?

...

What went well

- Identify the things that worked well

What can be improved

- Using the root cause analysis, explain what things can be improved.

Corrective actions

- - Issue labeled as corrective action

Guidelines

- Blameless Postmortems Guideline
- 5 whys

Linked issues 1

Relates to

Gitlab system wide outage on 2017-09-11
#2744 WoW ending ...

Pablo Carranza [GitLab] @pcarranza-gitlab added network label 2 years ago

Daniele Valeriani [GitLab] @omame-gitlab mentioned in issue #2744 (moved) 2 years ago

Daniele Valeriani [GitLab] @omame-gitlab · 2 years ago

According to Microsoft there is a bug in the WALinuxAgent that's running on our load balancers, as well as 47 more nodes.

I'm adding https://gitlab.com/gitlab-com/infrastructure/issues/2771 as a corrective action.

Daniele Valeriani [GitLab] @omame-gitlab assigned to @omame 2 years ago

Daniele Valeriani [GitLab] @omame-gitlab marked this issue as related to #2744 (moved) 2 years ago

Pablo Carranza [GitLab] @pcarranza-gitlab mentioned in issue #2694 (closed) 2 years ago

Daniele Valeriani [GitLab] @omame-gitlab changed milestone to %WoW ending 2017-09-19 2 years ago

Daniele Valeriani [GitLab] @omame-gitlab · 2 years ago

All virtual machines in production, canary and staging have been upgraded to walinuxagent `2.2.17`.

Therefore, I'm closing this issue as Microsoft confirms we won't have any more trouble with this.

Daniele Valeriani [GitLab] @omame-gitlab closed 2 years ago

Andrew Newdigate @andrewn moved to production#242 (closed) 1 year ago

Please register or sign in to reply