

openQA Infrastructure - action #72139

openQA services on OSD failed to connect to database

2020-09-30 11:38 - mkittler

Status:	Resolved	Start date:	2020-09-30
Priority:	High	Due date:	2020-10-20
Assignee:	okurz	% Done:	0%
Category:		Estimated time:	0.00 hour
Target version:	Ready		
Description			
<p>All openQA services which use the database showed connection errors. That's the first error logged by PostgreSQL:</p> <pre>2020-09-30 12:30:53.437 CEST openqa geekotest [7311]FATAL: remaining connection slots are reserved for non-replication superuser connections</pre>			
<p>From the openQA-side the errors look like:</p> <pre>Sep 30 12:47:45 openqa openqa[32459]: [error] [vJyMdc-a] DBIx::Class::Storage::DBI::catch {...} () : DBI Connection failed: DBI connect('dbname=openqa','geekotest',...) failed: FATAL: remaining connection slots are reserved for non-replication superuser connections at /usr/lib/perl5/vendor_perl/5.26.1/DBIx/Class/Storage/DBI.pm line 1517. at /usr/share/openqa/script/./lib/OpenQA/Schema.pm line 172</pre>			
<p>This lead to various alerts being triggered (Minion jobs alert, HTTP Response alert, Workers alert). A restart of the main openqa-webui service and postgresql service helped to fix the error. (Likely the restart of openqa-webui was unnecessary considering the other services could restore themselves without a restart.)</p> <p>I also retried the failed Minion jobs and all of them passed. So there shouldn't be any active warnings anymore.</p> <p>The question is what caused the connection limit to be exceeded. Theoretically we have a fixed number of services using a fixed number of connections.</p>			
Related issues:			
Related to openQA Project - action #72196: t/24-worker-jobs.t fails in OBS		Resolved	2020-10-02

History

#1 - 2020-09-30 13:11 - mkittler

It happened again

```
2020-09-30 14:40:59.072 CEST openqa geekotest [14311]FATAL: remaining connection slots are reserved for non-replication superuser connections
```

#2 - 2020-09-30 15:43 - okurz

- Status changed from New to In Progress
- Assignee set to mkittler
- Priority changed from Normal to Urgent
- Target version set to Ready

this is urgent and certainly you are working on it. Everyone can help of course but we should just make it obvious that this is actively tracked by at least one person.

#3 - 2020-09-30 16:18 - mkittler

PRs by Nick to improve the monitoring: https://gitlab.suse.de/openqa/salt-states-openqa/-/merge_requests/364, https://gitlab.suse.de/openqa/salt-states-openqa/-/merge_requests/365

In the meantime I've been restarting the database again to keep OSD operational. I also restarted the failed Minion jobs again. After the database restart I have been monitoring the number of database connections manually and saw a slow increase over time. At 62 connections I reverted the perl-Mojolicious package to version 5.59. The [diff](#) doesn't look *that* suspicious but it was the only lead. After the downgrade the number of connections is around 49 and doesn't seem to increase further. So *maybe* we've found the culprit.

#4 - 2020-09-30 17:37 - okurz

ok, I guess no upstream bug has been created yet but should be?

#5 - 2020-10-01 09:19 - kraith

This seems like something i should take a look at. Feel free to assign the ticket to me if Marius is done.

#6 - 2020-10-01 09:44 - okurz

- Assignee changed from mkittler to kraith

mkittler currently does not follow up with any active tasks. The monitoring change by nicksinger had been already mentioned.

[kraith](#) would be cool if you can take a look from mojo side.

#7 - 2020-10-01 12:29 - kraith

I've started running some tests and so far i can only confirm that there's a risk of leaks. For example Mojo::UserAgent does accumulate some garbage state information after a Mojo::IOLoop->reset call. But i've not been able to find anything in Minion or Mojo::Pg yet. There's a few more things i can try though to get to the root of it.

#8 - 2020-10-01 13:46 - kraith

I'm getting closer, the Mojo::IOLoop->reset leak was probably just a trigger for other bugs to become visible. Here's a first fix for an upstream leak in Mojo::Pg. <https://github.com/mojolicious/mojo-pg/compare/a546b0454d7a...29b22d36a953>

#9 - 2020-10-01 14:11 - kraith

And a followup. With one commit also reverting the upstream change Marius suspected of being the cause.

<https://github.com/mojolicious/mojo/compare/b181827b1a75...0ca361b75e38>

#10 - 2020-10-01 14:15 - kraith

I could not exactly replicate the problem from this ticket, but i think it wasn't actually the Minion jobs, but the subprocess from OpenQA::WebAPI::Controller::API::V1::Job. Subprocesses behave very similar to Minion jobs and also call Mojo::IOLoop->reset. And due to the calling context there is a lot more file descriptors to leak.

#11 - 2020-10-03 10:59 - okurz

- Related to action #72196: t/24-worker-jobs.t fails in OBS added

#12 - 2020-10-03 11:40 - okurz

- Status changed from In Progress to Blocked

- Assignee changed from kraith to okurz

- Priority changed from Urgent to High

It turns out [#72196](#) is a consequence of perl-Mojolicious 8.59->8.60 as well. So we have a clear reproducible test with t/24-worker-jobs.t showing that 8.60 is introducing problems. Of course we did not see that in circleci tests as we only run the old version until the dependency PR is accepted which also showed the corresponding problems but was even blocked by many circleCI tests failing due to the container image unable to be retrieved from registry.opensuse.org. [#69160](#) should help to prevent deployments going forward by running tests within Leap repos as well. As there had not been an automatic submit request for the new mojolicious version yet I have created <https://build.opensuse.org/request/show/839303> now with a manual update. You can also use the repo from <https://build.opensuse.org/package/show/home:okurz:branches:devel:languages:perl/perl-Mojolicious> if you need the new fixed version.

[kraith](#) I guess with this you do not plan further immediate work. I will take over the ticket waiting for the SR to be accepted. Also reducing prio as a fixed version is available and osd was hotpatched already some days ago.

EDIT: SR accepted, now <https://build.opensuse.org/request/show/839314> for openSUSE:Factory

#13 - 2020-10-06 10:09 - okurz

To prevent the broken version to be redeployed tomorrow on osd I will now try to link 8.61 into devel:openQA:Leap:15.1 overriding the current link from Factory:

```
for i in 1 2; do osc linkpac -f devel:languages:perl perl-Mojolicious devel:openQA:Leap:15.$i; done
```

after the SR to Factory <https://build.opensuse.org/request/show/839314> has been accepted we can revert this back with

```
for i in 1 2; do osc linkpac -f openSUSE:Factory perl-Mojolicious devel:openQA:Leap:15.$i; done
```

#14 - 2020-10-06 10:11 - okurz

- Status changed from Blocked to In Progress

I will monitor <https://build.opensuse.org/project/show/devel:openQA:Leap:15.1> and then see if I can retrigger the build dependency generation CI job that should create a new PR.

#15 - 2020-10-06 11:52 - okurz

- Due date set to 2020-10-13

- Status changed from In Progress to Feedback

<https://app.circleci.com/pipelines/github/os-autoinst/openQA/4417/workflows/1cd6b181-9298-48b6-8cf9-db9b12781c3c/jobs/42437> created <https://github.com/os-autoinst/openQA/pull/3446> which has now two approvals. Tests are currently running and unless there are tests failing we should merge this PR to unblock other PR tests.

Setting due date as reminder to check if perl-Mojolicious-8.61 reached Factory and we can remove the workaround and put the link to the Factory package again

#16 - 2020-10-08 18:07 - okurz

- Due date changed from 2020-10-13 to 2020-10-20

<https://build.opensuse.org/request/show/839314> for the submission of perl-Mojolicious to openSUSE:Factory is still pending and I am getting annoyed by all these failed tests and failed package builds. I now linked the fixed version directly into devel:openQA so that we also have that version available for any tests and builds based on Factory and/or Tumbleweed.

```
$ osc linkpac devel:languages:perl perl-Mojolicious devel:openQA
Sending meta data...
Done.
Creating _link... Done.
```

I hope this works. This as well should be reverted but here with a deletion of the (linked) package after the submission to openSUSE:Factory is accepted.

#17 - 2020-10-12 12:54 - okurz

SR was accepted after DimStar moved the SR out into another adi (together with the submissions for os-autoinst and openQA). All three are accepted now. I deleted the package link with `osc rdelete devel:openQA perl-Mojolicious`

#18 - 2020-10-15 08:15 - okurz

- Status changed from Feedback to Resolved

Packages are fine, we have better tests, no workarounds in place anymore, monitoring improved as well as tests, e.g. in OBS for Leap and more.