

openQA Infrastructure - action #96713

Slow grep in openqa-label-known-issues leads to high CPU usage

2021-08-10 12:05 - tinita

Status:	Resolved	Start date:	
Priority:	Normal	Due date:	2021-09-10
Assignee:	okurz	% Done:	0%
Category:		Estimated time:	0.00 hour
Target version:	Ready		

Description

Observation

On 2021-08-10 we experienced problems on osd like a high load, and one of the reasons seemed to be slow grep commands called by openqa-label-known-issues.

For example this regex seemed to be problematic:

```
grep -qPzo (?s)2021-.*T.*Error connecting to <root@s390p.*.suse.de>: No route to host /tmp/tmp.gy3iDKQDrV
```

Running over 4 minutes sometimes. (Removing the (?s) made it run in only a few milliseconds on a file with about 500kb.)

We are now lowering the timeout for the post fail hook

https://gitlab.suse.de/openqa/salt-states-openqa/-/merge_requests/538

But that will mean that the whole process can timeout because of one bad regex.

Suggestions

- Study current grep logic in [openqa-label-known-issues](#)
- The mentioned regex comes from this issue: [#93119](#)
- Maybe the regex can be improved
- Add a timeout to the grep call
- Make sure we are informed in some way of timed out grep calls so we can improve the regexes
- Reduce limit=200 (maximum number of jobs to check)

Related issues:

Related to openQA Infrastructure - action #96807: Web UI is slow and Apache R...	Resolved	2021-08-12	2021-10-01
Copied from openQA Infrastructure - action #96710: Error `Can't call method "...	Resolved	2021-08-10	2021-08-31
Copied to openQA Infrastructure - action #97943: Increase number of CPU cores...	Resolved		

History

#1 - 2021-08-10 12:05 - tinita

- Copied from action #96710: Error `Can't call method "write" on an undefined value` shows up in worker log leading to incompletes added

#2 - 2021-08-10 12:16 - mkittler

The SR for the timeout unfortunately needs a follow up: https://gitlab.suse.de/openqa/salt-states-openqa/-/merge_requests/539

#3 - 2021-08-10 13:04 - cdywan

- Description updated

#4 - 2021-08-10 13:17 - cdywan

Add a timeout to the grep call

Thinking out loud here. The script has some hard-coded regexes where grep is called individually, but results from progress are passed in one go. So I assume to allow a timeout per title we'd have to add the overhead of calling grep every single time.

It might be more beneficial to join other grep calls and speed up the whole thing?

#5 - 2021-08-10 13:29 - tinita

cdywan wrote:

Thinking out loud here. The script has some hard-coded regexes where grep is called individually, but results from progress are passed in one go.

Not sure what you mean.
We currently call grep for every title.

#6 - 2021-08-10 13:58 - cdywan

https://gitlab.suse.de/openqa/salt-states-openqa/-/merge_requests/541

tinita wrote:

We currently call grep for every title.

Nevermind, seems I was misreading the code. `label_on_issues_from_issue_tracker` loops through the issues individually.

#7 - 2021-08-10 14:16 - tinita

- Status changed from New to In Progress
- Assignee set to tinita

#8 - 2021-08-10 14:16 - tinita

- Status changed from In Progress to Feedback

PR: <https://github.com/os-autoinst/scripts/pull/99>

#9 - 2021-08-12 09:44 - tinita

PR was merged, scripts on osd updated.
I'll watch the logs.

#10 - 2021-08-17 08:16 - tinita

- Target version set to Ready

#11 - 2021-08-17 08:58 - cdywan

- Related to action #96807: Web UI is slow and Apache Response Time alert got triggered added

#12 - 2021-08-25 10:56 - okurz

- Due date set to 2021-09-03
- Priority changed from Urgent to High

High load seems to be solved meanwhile but the current timeout of just 5s is rather low. Please also see my related MR https://gitlab.suse.de/openqa/salt-states-openqa/-/merge_requests/550 where I want to configure a much longer timeout along with running scripts with nice

#13 - 2021-08-26 09:23 - okurz

- Assignee changed from tinita to okurz
- Priority changed from High to Normal

discussed with tinita and cdywan, noted down one idea for improvement in [#65271#note-70](#)

will merge https://gitlab.suse.de/openqa/salt-states-openqa/-/merge_requests/550 as soon as pipelines succeed.

Will watch logs in some days with `ssh osd 'sudo journalctl -u openqa-gru | grep timed'`

#14 - 2021-09-02 07:20 - okurz

- Due date changed from 2021-09-03 to 2021-09-10

- Status changed from Feedback to In Progress

Since the MR was merged I don't see problems. On OSD I could confirm that sometimes there are grep processes running with nice level 19 as expected. Monitoring in htop I see that telegraf pops up often but only with nice-level of 10. Mostly it's "openqa prefork" processes so that's fine. I don't think my change had negative impact. However we see that we book all 10 cores often so we should ask for increase in number of cores. Also we see alerts on CPU load lately. We can consider bumping the CPU load alert limits a bit as well.

#15 - 2021-09-02 07:38 - okurz

- Copied to action #97943: Increase number of CPU cores on OSD VM due to high usage size:S added

#16 - 2021-09-02 07:39 - okurz

- Status changed from In Progress to Feedback

Created https://gitlab.suse.de/openqa/salt-states-openqa/-/merge_requests/568 as well [#97943](#) to increase cores on OSD

#17 - 2021-09-02 20:56 - okurz

- Status changed from Feedback to Resolved

https://gitlab.suse.de/openqa/salt-states-openqa/-/merge_requests/568 merged, rest for [#97943](#)