

openQA Infrastructure - action #102062

check state of packages fails with test " = clean

2021-11-08 10:59 - cdywan

Status:	Resolved	Start date:	2021-11-08
Priority:	Normal	Due date:	2021-12-04
Assignee:	tinita	% Done:	0%
Category:		Estimated time:	0.00 hour
Target version:	Ready		

Description

Observation

This morning [check state of packages](#) failed:

```
.++ for i in {1..7}
++++ uname -i
2423+++ curl -s 'https://api.opensuse.org/public/build/devel:openQA/_result?repository=openSUSE_Le
ap_15.2&arch=x86_64'
[...]
++ echo 'Retry 7 after sleep ...'
2517Retry 7 after sleep ...
2518++ sleep 300
[...]
++ echo '$ test "$status" = "clean"'
2819$ test "$status" = "clean"
2820++ test '' = clean
2821+++ kill %1
```

Acceptance criteria

- **AC1:** Package check concludes with a meaningful result
- **AC2:** Package check succeeds

Suggestion

History

#1 - 2021-11-08 12:24 - mkittler

Judging by the log it is the same problem as <https://github.com/os-autoinst/openQA/pull/4351#issuecomment-962900388>. So this is already fixed and retrying the job helped.

Not sure whether it is worth improving the output. One had to scroll a little bit to the top but the source of the problem was actually apparent.

#2 - 2021-11-08 13:17 - cdywan

mkittler wrote:

Not sure whether it is worth improving the output. One had to scroll a little bit to the top but the source of the problem was actually apparent.

I suspect it was not apparent to me. Can you elaborate what one should have seen here?

#3 - 2021-11-08 13:22 - okurz

- Target version set to Ready

#4 - 2021-11-08 14:46 - mkittler

- Status changed from New to Feedback

- Assignee set to mkittler

- Target version deleted (Ready)

I saw the same dependency resolution error as [okurz](#) mentioned in the PR:

```
<status package="openQA" code="unresolvable">
  <details>have choice for libpq.so.5() (64bit) needed by perl-DBD-Pg: libpq5 postgresql14-devel-mini, have
choice for libpq.so.5() (64bit) needed by postgresql12-server: libpq5 postgresql14-devel-mini, have choice for
libpq5 &gt;= 12.6 needed by postgresql12-server: libpq5 postgresql14-devel-mini, have choice for libpq.so.5()
(64bit) needed by postgresql12: libpq5 postgresql14-devel-mini, have choice for libpq5 &gt;= 12.6 needed by po
stgresql12: libpq5 postgresql14-devel-mini</details>
</status>
```

And the following:

```
++ grep 'status package'
++ grep -v '\(disabled\|succeeded\|excluded\)'
  <status package="openQA" code="unresolvable">
  <status package="openQA:openQA-test" code="unresolvable">
```

Which makes it clear that these two packages were unresolvable (and they should be succeeded or at least disabled/excluded).

#5 - 2021-11-09 10:19 - okurz

- Target version set to Ready

#6 - 2021-11-09 14:02 - tinita

Regarding the scrolling up:

The dots come from this line (see the link above the line for what this is needed):

<https://gitlab.suse.de/openqa/osd-deployment/-/blob/master/gitlab-ci.yml#L49>

```
.gitlab_workaround: &gitlab_workaround |
(
  trap "kill %1" EXIT; while true; do sleep 1 && echo -n .; done &
  set -ex
```

(note that the -n for the echo did have an effect a while ago, and the dots were all on the same line, but someday gitlab added a linebreak itself)

I suggested that we increase the sleep time from 1 to 5 seconds. Since that is running in the background it won't affect the actual pipeline itself, it might just take 4 more seconds at the end (the pipeline in question took over 35 minutes).

https://gitlab.com/gitlab-org/gitlab-runner/-/issues/4119#note_257314431 suggests that a higher sleep time will still be good enough to workaround the bug.

That would considerably reduce the number of lines with a dot. I find it annoying how much I have to scroll to find actual error messages.

Alternatively we could check in which version(s) of gitlab and/or related software the bug was fixed, and which versions we have running. Then we could eventually remove the workaround completely.

#7 - 2021-11-11 11:09 - cdywan

- Description updated

- Assignee changed from mkittler to tinita

Discussed this briefly with marius/oli/tina/sebastian and as an improvement, the script could show a URL to the OBS web UI which would provide the context for what the script is also checking (maybe mapped from the API URL).

#8 - 2021-11-19 21:41 - tinita

- Status changed from Feedback to In Progress

#9 - 2021-11-20 04:10 - openqa_review

- Due date set to 2021-12-04

Setting due date based on mean cycle time of SUSE QE Tools

#10 - 2021-11-22 16:34 - tinita

- Status changed from In Progress to Feedback

MR to improve the error message: https://gitlab.suse.de/openqa/osd-deployment/-/merge_requests/40

#11 - 2021-11-24 10:06 - tinita

MR was approved and I just merged it

#12 - 2021-11-26 09:29 - tinita

- *Status changed from Feedback to Resolved*

Pipeline succeeded