openQA Project - action #92179

`make coverage` on openSUSE Leap 15.2 can fail with "Bad Sereal header: Not a valid Sereal document"

2021-05-05 14:59 - okurz

Status:	Resolved		**Start date:**	2021-05-05
Priority:	Low		**Due date:**	
Assignee:	tinita		**% Done:**	0%
Category:	Concrete Bugs		**Estimated time:**	0.00 hour
Target version:	Ready			
Difficulty:				

Description

Observation

See [#92164#note-2](#) :

```
make coverage KEEP_DB=1 TESTS=t/01-test-utilities.t
```

fails with

```
Can't read …/cover_db/runs/1620218733.13161.31806/cover.14 with Sereal: Sereal: Error: Bad Sereal
header: Not a valid Sereal document. at offset 1 of input at srl_decoder.c line 600 at /usr/lib/pe
rl5/vendor_perl/5.26.1/x86_64-linux-thread-multi/Devel/Cover/DB/IO/Sereal.pm line 34, <$fh> chunk
1.
```

on openSUSE Leap 15.2

Workaround

See https://github.com/os-autoinst/openQA/pull/3885 or remove perl-Sereal perl-Sereal-Decoder perl-Sereal-Encoder

Further details

See https://chat.suse.de/group/qa-tools?msg=nHxLEavqJPA5bFR3k

It seems like currently the DEVEL_COVER_DB_FORMAT=JSON is only used when running tests, and when producing the report, it is trying to use Sereal, and Sereal complains about JSON
the former "fix" made DEVEL_COVER_DB_FORMAT not set (or passed to the processes) at all. Sereal is the format, and Sereal::Encoder is the perl module that encodes a perl data structure to this format, and Sereal::Decoder decodes it. The Sereal perl module is just a kind of a placeholder requiring Encoder and Decoder.
Devel::Cover uses this format as the default format to enode the coverage data.
We switched to using JSON for codecov. We can say whenever we use Sereal then it's wrong because we use JSON within CI, don't want to support something else in other cases and it fails like it does for us.
e should just make sure that DEVEL_COVER_DB_FORMAT=JSON is correctly set for every target. If it's getting too hard we can move stuff into an external script.

checking if sereal is loaded might break if Sereal is used for something else then coverage in our code. If that would ever happen we could think about it at that time.

it seems like we do not really rely on perl-Sereal elsewhere within the openQA context.

History

#1 - 2021-05-05 15:44 - okurz

- Description updated

- Status changed from New to In Progress

- Assignee set to tinita

https://github.com/os-autoinst/openQA/pull/3887

#2 - 2021-05-05 15:58 - okurz

- Description updated

removing perl-Sereal perl-Sereal-Decoder perl-Sereal-Encoder also prevents the problem. This is why I suggested that maybe we should also have a check if these packages are loaded and then fail with an explicit error message.

#3 - 2021-05-05 17:07 - okurz

- Status changed from In Progress to Feedback

PR merged. I think it can be beneficial if we can improve further and have e.g. the additional error check. But if you don't have a idea how to do that it's ok to go without. Thanks already so much.

#4 - 2021-05-18 08:45 - tinita

- Status changed from Feedback to Resolved

#5 - 2021-05-18 08:48 - tinita

I just think hat it doesn't make sense to check if Sereal is loaded.
Also that apparently happens when Devel::Cover is reading the coverage results to generate a report.
We'd have to hook into that process that doesn't have anything to do with our code anymore.