

openQA Infrastructure - coordination #68923

coordination # 103938 (New): [saga][epic] Scale up: Efficient handling of large storage on o3

[epic] Use external videoencoder in production auto_review:"External encoder not accepting data"

2020-07-14 06:53 - mkittler

Status: New	Start date: 2020-11-13
Priority: Normal	Due date:
Assignee:	% Done: 50%
Category:	Estimated time: 0.00 hour
Target version: future	

Description

Motivation

os-autoinst supports "external videoencoder" where we can configure command lines to call other commands. With more efficient video encoders we can save storage space and hence avoid space issues and can store results for a longer time which helps especially for bug investigation discovered by openQA tests.

Acceptance criteria

- **AC1: DONE** One worker runs with an external video encoder configured -> [#77839](#)
- **AC2: DONE** The configured external video encoder provides significantly smaller video files instead of the default ones -> [#77839](#)
- **AC3: DONE** The worker is not overstressed by the external video encoder, i.e. not more jobs failing or incompleting due to an overstressed worker -> [#77839](#)
- **AC4:** All o3 workers run the same (exceptions allowed if required and documented) -> [#77842](#)
- **AC5:** The same as above for OSD -> [#77845](#)
- **AC6:** No more workarounds in OSD due to inefficient video encoder -> [#77848](#)

Suggestions

- Install ffmpeg on the worker.
- Try to find a good encoder setting. Likely VP9 would make sense so `ffmpeg -hide_banner -nostats -r 24 -f image2pipe -vcodec ppm -i - -pix_fmt yuv420p -c:v libvpx-vp9 -crf 35 -b:v 1500k -cpu-used 0` would be a start. Also consider AV1 (see [#75256](#)).
- Check whether the resulting videos work (including seeking).
- Try it on one worker host first. As not all worker hosts have the same hardware the setting might need to be tweaked per worker host.
- Monitor the CPU usage of ffmpeg e.g. via `htop` and also the overall performance of the worker, e.g. check for incomplete jobs due to performance issues.
- I'd refrain from using non-royalty free formats to avoid any legal problems.
- Adopt any special handling of videos in our production instances:
 - <https://gitlab.suse.de/openqa/salt-states-openqa/-/blob/master/etc/master/cron.d/SLES.CRON>

Subtasks:

action # 77839: Use external videoencoder in production on one o3 worker	Resolved
action # 77842: Use external videoencoder in production on all o3 machines	New

Related issues:

Related to openQA Project - action #67342: Support using a "generic" video en...	Resolved	2020-05-27	
Related to openQA Project - action #70873: Test fails because auto_review:"En...	Resolved	2020-09-02	2020-11-14
Related to openQA Project - action #75256: Try out AV1 video codec as potenti...	Resolved	2020-10-25	
Related to openQA Infrastructure - action #76987: re-encode some videos from ...	Resolved	2020-11-04	

History

#1 - 2020-07-14 06:57 - mkittler

- Related to action #67342: Support using a "generic" video encoder like `ffmpeg` instead of only relying on the one provided by `os-autoinst` itself added

#2 - 2020-07-24 09:03 - mkittler

- Assignee set to `mkittler`

I've installed ffmpeg on the o3 workers imagetester and power8 and enabled VP9 encoding in the worker configuration. It should be active on the next reboot.

#3 - 2020-07-24 13:33 - okurz

Careful, weekend incoming ;) So depending on your availability it can be safer to apply changes whenever you are available to monitor the effect more closely. Keep in mind that openSUSE never sleeps! ;)

#4 - 2020-07-24 14:00 - mkittler

I know. I plan to check the job history of these workers tomorrow. (Should only take one minute and disabling the setting again is easy as well.)

#5 - 2020-07-25 08:38 - okurz

- Status changed from New to In Progress

- Target version set to Ready

<https://openqa.opensuse.org/admin/workers/41> shows many incompletes, e.g. <https://openqa.opensuse.org/tests/1342147> with

```
[37m[2020-07-25T08:23:20.109 UTC] [debug] Backend process died, backend errors are reported below in the following lines:
```

```
External encoder not accepting data at /usr/lib/os-autoinst/backend/baseclass.pm line 157.
```

```
[0m[33m[2020-07-25T08:23:20.109 UTC] [info] :: OpenQA::Qemu::Proc::save_state: Saving QEMU state to qemu_state.json
```

```
[0m[37m[2020-07-25T08:23:22.137 UTC] [debug] flushing frames
```

```
[0m[37m[2020-07-25T08:23:22.146 UTC] [debug] QEMU: QEMU emulator version 3.1.1.1 (openSUSE Leap 15.1)
```

```
[0m[37m[2020-07-25T08:23:22.147 UTC] [debug] QEMU: Copyright (c) 2003-2018 Fabrice Bellard and the QEMU Project developers
```

```
[0m[37m[2020-07-25T08:23:22.147 UTC] [debug] QEMU: qemu-system-x86_64: terminating on signal 15 from pid 6965 (/usr/bin/isotovideo: backen)
```

```
[0m[37m[2020-07-25T08:23:22.148 UTC] [debug] sending magic and exit
```

```
[0m[37m[2020-07-25T08:23:22.148 UTC] [debug] received magic close
```

```
[0m[37m[2020-07-25T08:23:22.149 UTC] [debug] THERE IS NOTHING TO READ 15 4 3
```

```
[0m[37m[2020-07-25T08:23:22.149 UTC] [debug] terminating command server 6939 because test execution ended
```

```
[0m[37m[2020-07-25T08:23:22.149 UTC] [debug] isotovideo: informing websocket clients before stopping command server: http://127.0.0.1:20013/RieBgBe4MIpGFwPw/broadcast
```

```
[0m[37m[2020-07-25T08:23:22.160 UTC] [debug] backend process exited: 0
```

```
[0m[37m[2020-07-25T08:23:22.167 UTC] [debug] commands process exited: 0
```

```
[0m[37m[2020-07-25T08:23:23.268 UTC] [debug] done with command server
```

```
[0m[37m[2020-07-25T08:23:23.268 UTC] [debug] stopping autotest process 6942
```

```
[0m[37m[2020-07-25T08:23:23.279 UTC] [debug] [autotest] process exited: 1
```

```
[0m[37m[2020-07-25T08:23:24.380 UTC] [debug] done with autotest process
```

```
[0m[37m[2020-07-25T08:23:24.380 UTC] [debug] isotovideo failed
```

```
[0m[37m[2020-07-25T08:23:24.381 UTC] [debug] stopping backend process 6965
```

```
[0m[37m[2020-07-25T08:23:24.382 UTC] [debug] done with backend process
```

```
[0mCan't locate object method "is_running" via package "/usr/lib/os-autoinst" (perhaps you forgot to load "/usr/lib/os-autoinst") at /usr/bin/isotovideo line 157.
```

```
END failed--call queue aborted.
```

I disabled again the settings in /etc/openqa/workers.ini again on imagetester:

```
#EXTERNAL_VIDEO_ENCODER_CMD=ffmpeg -y -hide_banner -nostats -r 24 -f image2pipe -vcodec ppm -i - -c:v libvpx-vp9 -crf 35 -b:v 1500k -cpu-used 0
```

```
#EXTERNAL_VIDEO_ENCODER_OUTPUT_FILE_EXTENSION=webm
```

... and restarted the systemd service. power8 is not a problem as all jobs are started with NOVIDEO=1.

Using <https://github.com/os-autoinst/scripts/blob/master/openqa-restart-incompletes-on-worker-instance> I restarted incomplete jobs with

```
env WORKER=imagetester openqa-restart-incompletes-on-worker-instance
```

#6 - 2020-07-25 10:31 - okurz

- Subject changed from Use external videoencoder in production to Use external videoencoder in production auto_review:"External encoder not accepting data"

#7 - 2020-07-30 12:36 - mkittler

Looks like apparmor (which is running on imagetester) prevented ffmpeg to start:

```
[2020-07-25T08:23:17.559 UTC] [debug] Launching external video encoder: ffmpeg -y -hide_banner -nostats -r 24 -f image2pipe -vcodec ppm -i - -c:v libvpx-vp9 -crf 35 -b:v 1500k -cpu-used 0 '/var/lib/openqa/pool/1/video.webm'
```

sh: /usr/bin/ffmpeg: Permission denied

PR: <https://github.com/os-autoinst/openQA/pull/3292>

#8 - 2020-08-04 14:29 - mkittler

I'm currently running a job with the mentioned settings on a dedicated worker slot on imagetester and it works. So the apparmor PR worked. Whether there are performance problems is yet to be determined (and likely depends on the worker host). Due to my extra worker slot there are currently 3 tests running in imagetester and one of them encodes VP9. The CPU usage of the ffmpeg process ranges from 0 to 100 percent. So far it seems it is fast enough and doesn't disturb the actual jobs.

~~After the test has finished with no further problems I'm going to enable the VP9 encoder on both regular worker slots and disable the extra worker slot again.~~ The video is truncated so I'm not enabling the encoder. (see <https://openqa.opensuse.org/tests/1353163>)

#9 - 2020-08-10 11:59 - mkittler

The truncated video can be locally reproduced as well. It can happen when the test execution just ends (isotovideo terminates itself; it is *not* aborted by the worker or Ctrl+C).

The problem is caused by not explicitly stopping (and waiting for) ffmpeg in that case. Of course I've thought about this problem when introducing the external video encoder and have even already implemented it:

<https://github.com/os-autoinst/os-autoinst/compare/master...Martchus:terminate-video-encoder?expand=1>

However, I didn't propose that change after all because:

1. When isotovideo is stopped by the worker or Ctrl+C SIGTERM/SIGINT is sent to the video encoder process without the extra need to do it from our side.
2. When ffmpeg receives a 2nd signal to exit it will immediately exit without further finalizing the file.

These points together create an annoying problem: When isotovideo terminates itself (and therefore sends SIGTERM to the backend) the backend would need to send SIGTERM to ffmpeg and wait for its termination. When isotovideo is stopped by the worker or Ctrl+C we must *not* send SIGTERM to ffmpeg because that would mean ffmpeg receives 2 signals to exit (because of 1.) and would possibly leave the file unfinalized (because of 2.).

It is currently not clear to me how to solve this. I haven't found anything in ffmpeg's documentation to prevent 2. (which would be the easiest solution). The backend process would need to determine whether it received SIGINT/SIGTERM from isotovideo simply because the test execution ended or from the worker/shell which sends the exit signal to the entire process group. Only in the first case it would terminate ffmpeg.

Maybe it also works to close the pipe we use to send data to ffmpeg. Closing the pipe might cause ffmpeg to exist but would not interfere with a signal sent to ffmpeg by the worker/shell.

Another solution would be to explicitly terminate (and wait for) the process group within the worker - even if isotovideo itself terminated on its own. (That would at least help when isotovideo is executed within the worker. Getting rid of any potential leftover process is likely a good idea anyways.)

#10 - 2020-08-10 12:20 - mkittler

According to a reply in #ffmpeg on freenode closing the pipe is supposed to work and there's really no option to change ffmpeg's behavior (2.). So I'll try closing the pipe. Nevertheless I'll also improve the worker code to take care of the whole process group in any case.

#11 - 2020-08-10 14:38 - mkittler

PR for closing pipes: <https://github.com/os-autoinst/os-autoinst/pull/1503>

Improving the worker code to take care of the whole process group would likely be a little bit more work because it currently lacks a way to track the process group ID and only relies on calling getppid for the PID of the immediate child process. Likely one needed to use vfork() to keep track of the process group ID from the parent process side. (see https://en.wikipedia.org/wiki/Process_group)

#12 - 2020-08-17 10:12 - cdywan

The PR got merged. Do you have more items in mind for this?

#13 - 2020-08-19 10:20 - mkittler

Yes - the PR was just for fixing an issue which came up when testing this in production. What I actually want to do is finally enabling this in production. Therefore I'm trying my luck on imagetester again.

#14 - 2020-08-19 11:12 - mkittler

- Status changed from In Progress to Feedback

It seems to work now: <https://openqa.opensuse.org/tests/1366465>

So I'll keep using VP9 enabled on imagetester.

I've noticed some Perl warnings produced by the code from my last PR so here's a fix: <https://github.com/os-autoinst/os-autoinst/pull/1511>

#15 - 2020-08-20 14:21 - mkittler

It seems to work nicely on both imagetester worker slots and my PR to fix warnings worked as well.

However, I've discovered that Firefox doesn't like the produced VP9 videos (although it is supposed to support VP9 within WebM in general). The videos themselves are definitely not truncated anymore. Maybe Firefox has problems with some particularity of the files produced via ffmpeg -y -hide_banner -nostats -r 24 -f image2pipe -vcodec ppm -i - -c:v libvpx-vp9 -crf 35 -b:v 1500k -cpu-used 0.

#16 - 2020-08-20 15:00 - mkittler

Luckily it was just the color space which wasn't supported by Firefox. I extended the example command to add -pix_fmt yuv420p to the ffmpeg arguments: <https://github.com/os-autoinst/os-autoinst/pull/1513>

So since it generally works I suppose we can enable it on more workers.

#17 - 2020-08-25 10:33 - mkittler

For the record: To enable the external video encoder on further workers one has to install ffmpeg¹ and add the following two lines to /etc/openqa/workers.ini (as it is already configured on imagetester right now):

```
EXTERNAL_VIDEO_ENCODER_CMD=ffmpeg -y -hide_banner -nostats -r 24 -f image2pipe -vcodec ppm -i - -pix_fmt yuv420p -c:v libvpx-vp9 -crf 35 -b:v 1500k -cpu-used 0
EXTERNAL_VIDEO_ENCODER_OUTPUT_FILE_EXTENSION=webm
```

I've tested ffmpeg 3.x and 4.x as provided by Leap 15.1 and Tumbleweed.

The change takes effect after restarting the worker service(s). After that I would recommend to closely monitor the workers, e.g. checkout the job history for the worker slots on the web UI for incomplete jobs and also check whether the video looks good and is seekable. Also check for failed jobs because they might have failed due to a performance regression caused by the additional CPU load.

The ffmpeg parameters can be tweaked of course as needed. However, I'd like to stick with VP9 because it is currently the most modern, production-ready and royalty-free format which is available. It can be encoded and decoded under openSUSE without having to enable additional repositories. (See my previous comment for why -pix_fmt is required. More VP9 options are documented in the [ffmpeg wiki](#).)

#18 - 2020-09-01 13:34 - mkittler

- Status changed from Feedback to Workable
- Assignee deleted (mkittler)

#19 - 2020-09-02 16:59 - okurz

- Related to action #70873: Test fails because auto_review:"Encoder not accepting data":retry, video is missing added

#20 - 2020-10-26 10:29 - mkittler

- Related to action #75256: Try out AV1 video codec as potential new default added

#21 - 2020-10-26 10:33 - mkittler

- Description updated

#22 - 2020-11-04 16:03 - mkittler

- Description updated

#23 - 2020-11-09 20:10 - okurz

- Related to action #76987: re-encode some videos from existing results to save space added

#24 - 2020-11-12 13:16 - okurz

- Parent task set to #64746

#25 - 2020-11-13 09:34 - cdywan

- Start date changed from 2020-07-14 to 2020-11-20

#26 - 2020-11-13 09:38 - cdywan

- Description updated

#27 - 2020-11-13 15:16 - okurz

- Tracker changed from action to coordination
- Subject changed from Use external videoencoder in production auto_review:"External encoder not accepting data" to [epic] Use external

videoencoder in production auto_review:"External encoder not accepting data"

- *Description updated*
- *Assignee set to okurz*

I realized that the "acceptance criteria" are actually "suggestions". I improved the description so that actually we can treat it as an epic and do it in smaller steps one by one.

#28 - 2020-11-13 15:39 - okurz

- *Description updated*
- *Status changed from Workable to Blocked*

#29 - 2021-07-15 10:04 - okurz

- *Target version changed from Ready to future*

#30 - 2021-12-14 10:00 - okurz

- *Parent task changed from #64746 to #103938*

#31 - 2021-12-14 10:01 - okurz

- *Status changed from Blocked to New*
- *Assignee deleted (okurz)*