

openQA Project - action #103605

Most tests time out with perl-Mojo-IOLoop-ReadWriteProcess 0.28

2021-12-07 10:40 - dimstar

Status:	Resolved	Start date:	2021-12-07
Priority:	Normal	Due date:	
Assignee:	cdywan	% Done:	0%
Category:	Concrete Bugs	Estimated time:	0.00 hour
Target version:	Ready		
Difficulty:			
Description			
Related issues:			
Blocked by openQA Project - action #103611: test failure: os-autoinst: 29-bac...		Resolved	2021-11-30

History

#1 - 2021-12-07 10:41 - dimstar

- Subject changed from *Pretty much all tests end 'timeout exceeded'*, e.g to *Pretty much all tests end 'timeout exceeded'*

<https://openqa.opensuse.org/tests/2073993> - that test runs like 20 minutes, and ran into a timeout of 2h (on ow7) (edited)

```
[2021-12-07T08:38:56.350654+01:00] [debug] done with backend process
37495: EXIT 0
[2021-12-07T10:32:53.258690+01:00] [info] +++ worker notes +++
[2021-12-07T10:32:53.258871+01:00] [info] End time: 2021-12-07 09:32:53
[2021-12-07T10:32:53.258923+01:00] [info] Result: timeout
```

there are a bunch more that are 'all done' - then timeout exceeded (e.g <https://openqa.opensuse.org/tests/2074158>)

if one catches a test 'stalled' in that situation and opens it in the web-browser, it finishes the upload and makes the test reach its final state

#2 - 2021-12-07 10:42 - dimstar

- Priority changed from *Normal* to *Immediate*

#3 - 2021-12-07 11:16 - okurz

- Target version set to *Ready*

#4 - 2021-12-07 11:18 - favogt

strace of a worker process with a stuck job shows this repeating activity:

```
poll([{fd=5, events=POLLIN|POLLPRI}], 1, 10000) = 0 (Timeout)
stat("/var/lib/openqa/pool/9/autoinst-status.json", {st_mode=S_IFREG|0644, st_size=85, ...}) = 0
geteuid() = 486
geteuid() = 486
openat(AT_FDCWD, "/var/lib/openqa/pool/9/autoinst-status.json", O_RDONLY) = 7
ioctl(7, TCGETS, 0x7ffdc720010) = -1 ENOTTY (Unpassender IOCTL (I/O-Control) für das Gerät)
lseek(7, 0, SEEK_CUR) = 0
fstat(7, {st_mode=S_IFREG|0644, st_size=85, ...}) = 0
fcntl(7, F_SETFD, FD_CLOEXEC) = 0
read(7, "{\n  \"current_test\" : \"\", \n  \"status\" : \"finished\", \n  \"test_execution_paused\" : 0\n}\n", 131072) = 85
read(7, "", 131072) = 0
close(7) = 0
stat("/var/lib/openqa/pool/9/testresults/test_order.json", {st_mode=S_IFREG|0644, st_size=800, ...}) = 0
openat(AT_FDCWD, "/var/lib/openqa/pool/9/testresults/test_order.json", O_RDONLY) = 7
ioctl(7, TCGETS, 0x7ffdc720010) = -1 ENOTTY (Unpassender IOCTL (I/O-Control) für das Gerät)
lseek(7, 0, SEEK_CUR) = 0
fstat(7, {st_mode=S_IFREG|0644, st_size=800, ...}) = 0
fcntl(7, F_SETFD, FD_CLOEXEC) = 0
read(7, "{\n  {\n    \"category\" : \"boot\", \n    \"flags\" : {\n      \"fatal\" : 1, \n      \"milestone\" : 1\n    }, \n    \"name\" : \"boot_to_desktop\", \n    \"script\" : \"tests/boot/boot_to_desktop.pm\"\n  }, \n  {\n    \"category\" : \"nfs\", \n    \"flags\" : {\n      \"fatal\" : 1\n    }, \n
```

```

"name" : "install",\n      "script" : "tests/nfs/install.pm"\n    },\n    {\n      "category" : "nfs",\n      "name" : "run",\n      "script" : "tests/nfs/run.pm"\n    },\n    {\n      "category" : "nfs",\n      "flags" : {\n        "no_rollback" : 1\n      },\n      "name" : "generate_report",\n      "script" : "tests/nfs/generate_report.pm"\n    },\n    {\n      "category" : "shutdown",\n      "flags" : {\n        "fatal" : 1\n      },\n      "name" : "shutdown",\n      "script" : "tests/shutdown/shutdown.pm"\n    }\n  ], 131072) = 800
read(7, "", 131072) = 0
close(7) = 0
getpid() = 18473
poll([{fd=5, events=POLLIN|POLLPRI}], 1, 0) = 0 (Timeout)
socket(AF_UNIX, SOCK_STREAM|SOCK_CLOEXEC|SOCK_NONBLOCK, 0) = 7
connect(7, {sa_family=AF_UNIX, sun_path="/var/run/nscd/socket"}, 110) = 0
sendto(7, "\2\0\0\0\16\0\0\0\21\0\0\0\0openqal-opensuse\0", 29, MSG_NOSIGNAL, NULL, 0) = 29
poll([{fd=7, events=POLLIN|POLLEERR|POLLHUP}], 1, 5000) = 1 ({fd=7, revents=POLLIN})
read(7, "\2\0\0\0\1\0\0\0\1\0\0\0\4\0\0\0\0\0\0\0\0\0\0\0", 24) = 24
read(7, "\300\250pd\2openqal-opensuse.openqanet.opensuse.org\0", 45) = 45
close(7) = 0
socket(AF_INET, SOCK_STREAM, IPPROTO_TCP) = 7
ioctl(7, TCGETS, 0x7ffdce7200d0) = -1 ENOTTY (Unpassender IOCTL (I/O-Control) für das Gerät)
lseek(7, 0, SEEK_CUR) = -1 ESPIPE (Nicht erlaubter Seek)
ioctl(7, TCGETS, 0x7ffdce7200d0) = -1 ENOTTY (Unpassender IOCTL (I/O-Control) für das Gerät)
lseek(7, 0, SEEK_CUR) = -1 ESPIPE (Nicht erlaubter Seek)
fcntl(7, F_SETFD, FD_CLOEXEC) = 0
fcntl(7, F_GETFL) = 0x2 (flags O_RDWR)
fcntl(7, F_SETFL, O_RDWR|O_NONBLOCK) = 0
connect(7, {sa_family=AF_INET, sin_port=htons(80), sin_addr=inet_addr("192.168.112.100")}, 16) = -1 EINPROGRESS (Die Operation ist jetzt in Bearbeitung)
fcntl(7, F_GETFL) = 0x802 (flags O_RDWR|O_NONBLOCK)
poll([{fd=5, events=POLLIN|POLLPRI}, {fd=7, events=POLLOUT}], 2, 77885) = 1 ({fd=7, revents=POLLOUT})
getsockopt(7, SOL_SOCKET, SO_ERROR, [0], [256->4]) = 0
connect(7, {sa_family=AF_INET, sin_port=htons(80), sin_addr=inet_addr("192.168.112.100")}, 16) = 0
getpeername(7, {sa_family=AF_INET, sin_port=htons(80), sin_addr=inet_addr("192.168.112.100")}, [256->16]) = 0
setsockopt(7, SOL_TCP, TCP_NODELAY, [1], 4) = 0
getsockname(7, {sa_family=AF_INET, sin_port=htons(59694), sin_addr=inet_addr("192.168.112.12")}, [256->16]) = 0
getsockname(7, {sa_family=AF_INET, sin_port=htons(59694), sin_addr=inet_addr("192.168.112.12")}, [256->16]) = 0
getpeername(7, {sa_family=AF_INET, sin_port=htons(80), sin_addr=inet_addr("192.168.112.100")}, [256->16]) = 0
poll([{fd=7, events=POLLIN|POLLPRI|POLLOUT}, {fd=5, events=POLLIN|POLLPRI}], 2, 77884) = 1 ({fd=7, revents=POLLIN|POLLOUT})
write(7, "POST /api/v1/jobs/2073563/status HTTP/1.1\r\nAccept-Encoding: gzip\r\nX-API-Key: asdf\r\nHost: openqal-opensuse\r\nUser-Agent: Mojolicious (Perl)\r\nAccept: application/json\r\nContent-Type: application/json\r\nContent-Length: 662\r\nX-API-Hash: asdf\r\nX-API-Microtime: 1638870488\r\n\r\n{\"status\": \"cmd_srv_url\": \"http://localhost:20093/SeeoqM4ua7lrsAXO\", \"test_execution_paused\": 0, \"test_order\": {\"category\": \"boot\", \"flags\": {\"fatal\": 1, \"milestone\": 1}, \"name\": \"boot_to_desktop\", \"script\": \"tests/boot/boot_to_desktop.pm\"}, {\"category\": \"nfs\", \"flags\": {\"fatal\": 1}, \"name\": \"install\", \"script\": \"tests/nfs/install.pm\"}, {\"category\": \"nfs\", \"flags\": {}, \"name\": \"run\", \"script\": \"tests/nfs/run.pm\"}, {\"category\": \"nfs\", \"flags\": {\"no_rollback\": 1}, \"name\": \"generate_report\", \"script\": \"tests/nfs/generate_report.pm\"}, {\"category\": \"shutdown\", \"flags\": {\"fatal\": 1}, \"name\": \"shutdown\", \"script\": \"tests/shutdown/shutdown.pm\"}}, \"worker_hostname\": \"192.168.112.12\", \"worker_id\": 190}}", 977) = 977
poll([{fd=7, events=POLLIN|POLLPRI}, {fd=5, events=POLLIN|POLLPRI}], 2, 77884) = 1 ({fd=7, revents=POLLIN})
read(7, "HTTP/1.1 200 OK\r\nDate: Tue, 07 Dec 2021 09:48:09 GMT\r\nServer: Mojolicious (Perl)\r\nContent-Type: application/json; charset=UTF-8\r\nContent-Length: 69\r\nStrict-Transport-Security: max-age=31536000; includeSubDomains\r\n\r\n{\"job_result\": \"passed\", \"known_files\": [], \"known_images\": [], \"result\": 1}", 131072) = 283
close(7) = 0
pipe([7, 9]) = 0
ioctl(7, TCGETS, 0x7ffdce7200d0) = -1 ENOTTY (Unpassender IOCTL (I/O-Control) für das Gerät)
lseek(7, 0, SEEK_CUR) = -1 ESPIPE (Nicht erlaubter Seek)
ioctl(9, TCGETS, 0x7ffdce7200d0) = -1 ENOTTY (Unpassender IOCTL (I/O-Control) für das Gerät)
lseek(9, 0, SEEK_CUR) = -1 ESPIPE (Nicht erlaubter Seek)
fcntl(7, F_SETFD, FD_CLOEXEC) = 0
fcntl(9, F_SETFD, FD_CLOEXEC) = 0
rt_sigprocmask(SIG_SETMASK, ~[RTMIN RT_1], [], 8) = 0
clone(child_stack=NULL, flags=CLONE_CHILD_CLEARTID|CLONE_CHILD_SETTID|SIGCHLD, child_tidptr=0x7f77a6e8c450) = 22205
rt_sigprocmask(SIG_SETMASK, [], strace: Process 22205 attached
NULL, 8) = 0
[pid 22205] set_robust_list(0x7f77a6e8c460, 24) = 0
[pid 22205] rt_sigprocmask(SIG_SETMASK, [], NULL, 8) = 0
[pid 18473] getpid() = 18473
[pid 18473] close(9) = 0
[pid 18473] poll([{fd=5, events=POLLIN|POLLPRI}, {fd=7, events=POLLIN|POLLPRI|POLLOUT}], 2, 0 <unfinished ...>
[pid 22205] write(9, "0-[\\"\\",\\"\\"]", 9 <unfinished ...>
[pid 18473] <... poll resumed> = 0 (Timeout)

```

```

[pid 22205] <... write resumed>          = 9
[pid 18473] poll([{fd=5, events=POLLIN|POLLPRI}, {fd=7, events=POLLIN|POLLPRI|POLLOUT}], 2, 77828 <unfinished ...>
[pid 22205] exit_group(0 <unfinished ...>
[pid 18473] <... poll resumed>           = 1 ({fd=7, revents=POLLIN})
[pid 22205] <... exit_group resumed>     = ?
[pid 18473] read(7, "0-[\\"", 131072) = 9
[pid 18473] poll([{fd=5, events=POLLIN|POLLPRI}, {fd=7, events=POLLIN|POLLPRI|POLLOUT}], 2, 77828 <unfinished ...>
[pid 22205] +++ exited with 0 +++
<... poll resumed>                       = 1 ({fd=7, revents=POLLHUP})
--- SIGCHLD {si_signo=SIGCHLD, si_code=CLD_EXITED, si_pid=22205, si_uid=486, si_status=0, si_utime=0, si_stime=0} ---
rt_sigreturn({mask=[]})                  = 1
rt_sigprocmask(SIG_BLOCK, [CHLD], [], 8) = 0
rt_sigaction(SIGCHLD, NULL, {sa_handler=0x5578ac776980, sa_mask=[], sa_flags=SA_RESTORER, sa_restorer=0x7f77a62f4050}, 8) = 0
wait4(-1, [{WIFEXITED(s) && WEXITSTATUS(s) == 0}], WNOHANG, NULL) = 22205
wait4(-1, 0x7ffdce71fe94, WNOHANG, NULL) = -1 ECHILD (Keine Kind-Prozesse)
rt_sigprocmask(SIG_UNBLOCK, [CHLD], NULL, 8) = 0
read(7, "", 131072)                      = 0
getpid()                                  = 18473
wait4(22205, 0x7ffdce720324, 0, NULL)     = -1 ECHILD (Keine Kind-Prozesse)
close(7)                                  = 0
poll([{fd=5, events=POLLIN|POLLPRI}], 1, 0) = 0 (Timeout)
poll([{fd=5, events=POLLIN|POLLPRI}], 1, 10000^Cstrace: Process 18473 detached
<detached ...>

```

#5 - 2021-12-07 11:34 - cdywan

dimstar wrote:

if one catches a test 'stalled' in that situation and opens it in the web-browser, it finishes the upload and makes the test reach its final state

Do you mean you managed to open a test before it tiemd out and that "fixed" it? Do you have a link, to compare what happened in that case?

#6 - 2021-12-07 11:40 - cdywan

dimstar wrote:

<https://openqa.opensuse.org/tests/2073993> - that test runs like 20 minutes, and ran into a timeout of 2h (on ow7) (edited)

The "next" one just passed: <https://openqa.opensuse.org/tests/2074422>

Note the upload logs which the "passed+timed out" job doesn't have:

```

[2021-12-07T12:34:02.021998+01:00] [info] Uploading baseline-lynis-audit-system-nocolors-Tumbleweed-x86_64-textmode
[2021-12-07T12:34:02.022342+01:00] [info] Uploading baseline-lynis-audit-system-nocolors-Tumbleweed-x86_64-textmode using multiple chunks
[2021-12-07T12:34:02.024258+01:00] [info] baseline-lynis-audit-system-nocolors-Tumbleweed-x86_64-textmode: 1 chunks
[2021-12-07T12:34:02.024402+01:00] [info] baseline-lynis-audit-system-nocolors-Tumbleweed-x86_64-textmode: chunks of 1000000 bytes each
[2021-12-07T12:34:02.115405+01:00] [info] baseline-lynis-audit-system-nocolors-Tumbleweed-x86_64-textmode: Processing chunk 1/1, avg. speed ~30.967 KiB/s
[2021-12-07T12:34:02.115870+01:00] [info] Uploading lynis_audit_system_current_file
[2021-12-07T12:34:02.116082+01:00] [info] Uploading lynis_audit_system_current_file using multiple chunks
[2021-12-07T12:34:02.117448+01:00] [info] lynis_audit_system_current_file: 1 chunks
[2021-12-07T12:34:02.117751+01:00] [info] lynis_audit_system_current_file: chunks of 1000000 bytes each
[2021-12-07T12:34:02.230059+01:00] [info] lynis_audit_system_current_file: Processing chunk 1/1, avg. speed ~30.521 KiB/s

```

and also

```
[2021-12-07T12:34:01.843496+01:00] [info] Isotovideo exit status: 0
```

#7 - 2021-12-07 11:51 - dimstar

cdywan wrote:

dimstar wrote:

if one catches a test 'stalled' in that situation and opens it in the web-browser, it finishes the upload and makes the test reach its final state

Do you mean you managed to open a test before it timed out and that "fixed" it? Do you have a link, to compare what happened in that case?

e.g. just happened on <https://openqa.opensuse.org/tests/2074403#>

Found it as '100% progressed' in the all tests view; open it in new tab, within seconds, it is 'passed'

```
[2021-12-07T11:44:19.275642Z] [debug] stopping backend process 18287
[2021-12-07T11:44:19.275820Z] [debug] done with backend process
18250: EXIT 0
[2021-12-07T11:50:25.409349Z] [info] Isotovideo exit status: 0
[2021-12-07T11:50:25.465881Z] [info] +++ worker notes +++
[2021-12-07T11:50:25.466265Z] [info] End time: 2021-12-07 11:50:25
[2021-12-07T11:50:25.466443Z] [info] Result: done
```

Here it indicates a 6 min delay (which is when I must have opened it)

#8 - 2021-12-07 11:52 - dimstar

cdywan wrote:

dimstar wrote:

<https://openqa.opensuse.org/tests/2073993> - that test runs like 20 minutes, and ran into a timeout of 2h (on ow7) (edited)

The "next" one just passed: <https://openqa.opensuse.org/tests/2074422>

Note: it did not 'just' pass: this is another test which I opened in a tab once it reached 100% complete. At that point, it moved.

#9 - 2021-12-07 11:57 - mkittler

It looks very much like isotovideo exits just fine (within reasonable time) but the worker is completely unaware of it. We're using RWP to spawn isotovideo so it is likely related to <https://github.com/mudler/Mojo-IOLoop-ReadWriteProcess/issues/29>. So likely related to recent RWP changes. (I've wrote this comment before I've seen @fvogt's comment.)

#10 - 2021-12-07 12:09 - cdywan

- Related to action #103611: test failure: os-autoinst: 29-backend-driver.t:31 Failed test 'exit logged' added

#11 - 2021-12-07 13:25 - kraih

After reviewing the RWP commits from the past few months i suspect these two PRs.

<https://github.com/mudler/Mojo-IOLoop-ReadWriteProcess/pull/22><https://github.com/mudler/Mojo-IOLoop-ReadWriteProcess/pull/24>

#12 - 2021-12-07 13:28 - mkittler

It is very likely 1e0addb6cee6ffd74e9e611270fa9eb8759a6976.

#13 - 2021-12-07 13:38 - kraih

Thanks to the test case from #103611 it could be bisected to

<https://github.com/mudler/Mojo-IOLoop-ReadWriteProcess/commit/1e0addb6cee6ffd74e9e611270fa9eb8759a6976>. We should probably deploy an older version of RWP until the issue has been fixed upstream.

#14 - 2021-12-07 14:08 - favogt

Adding some conversation from #eng-testing for completeness:

13:12 Fabian Vogt

```
[info] isotovideo has been started (PID: 29829)
[info] 29829: WORKING 2074471
```

(visiting job page)

```
>> Mojo::IOLoop::ReadWriteProcess::_fork_collect_status(): Forked code Process Exit status: 0
>> Mojo::IOLoop::ReadWriteProcess::_fork_collect_status(): Forked code Process Exit status: 0
[info] Isotovideo exit status: 0
```

The mentioned pid is already dead before, so the child death went unnoticed

13:16 Marius Kittler

@Fabian Vogt This supports my theory that RWP is the culprit. It looks like when one is calling `is_running` (which we do when activating the live view) that explicitly that invokes `$self->session->consume_collected_info`; and hence the missing SIGCHLD handling is worked around.

13:24 Fabian Vogt

Yeah, I suspect that RWP needs `consume_collected_info` called regularly
Otherwise the death recorded in the SIGCHLD handler is never emitted

13:28 Marius Kittler

Having to call it regularly defeats the point of having an event for the process termination. I'm currently looking at the RWP code and it seems a bit broken, indeed. However, I'd have to come up with a good test for reproducing the problem.

13:28 Fabian Vogt

I hotpatched `Session.pm` to collect in SIGCHLD again and it fixed it

So this is caused by `1e0adbb6cee6ffd74e9e611270fa9eb8759a6976`. I see different ways to address that

a) emit from the signal handler if the process was registered, and if the process was not registered yet add it to a list and emit on registration

b) block SIGCHLD between fork and registration

Oh, forgot about c) In the SIGCHLD handler, register a timer event to collect when it reaches the main loop again

14:05 Marius Kittler

@Fabian Vogt

a) Seems to be simple enough. I don't see a problem with running the handler code from the signal handler. After all, Perl signal handlers are not "normal C signal handlers" so you can actually do anything within them.

b) Should work as well but is likely a bit harder to implement than a).

c) This requires that the IO loop is actually running which might not be the case in all uses of the library (e.g. in the `os-autoinst` backend). So I would favor a) and b). Note that Mojolicious' events are actually distinct from using the IO loop and so far these events worked without needing to run the IO loop.

14:16 Fabian Vogt

@Marius Kittler a) Only downside is that there is a theoretical race condition: Between the forked process exiting (and being waited for by the parent) and the registration, the PID could get recycled

b) Might only need the signalblocker code we already have in `os-autoinst`

c) Ok, that wouldn't work then

14:19 Marius Kittler

Then it is likely the best to go for b).

#15 - 2021-12-07 17:09 - cdywan

- Status changed from *New* to *Feedback*

- Assignee set to *cdywan*

- Priority changed from *Immediate* to *Urgent*

I don't know why nobody took this. And I'm obviously not going to claim I'm "the" person working this, but this should not stay an unprocessed ticket.

#16 - 2021-12-08 08:22 - cdywan

I realized there's no mention of packages on OBS. Marius dropped the packages from `devel:openQA`, meaning on 15.3 you get an older version and on 15.2 you get an **outdated version** (meaning it breaks before you can run into errors due to the changed behavior). And this is also why [osd-deployment](#) won't currently succeed.

#17 - 2021-12-08 10:18 - cdywan

cdywan wrote:

I realized there's no mention of packages on OBS. Marius dropped the packages from `devel:openQA`, meaning on 15.3 you get an older version and on 15.2 you get an **outdated version** (meaning it breaks before you can run into errors due to the changed behavior). And this is also why [osd-deployment](#) won't currently succeed.

[mkittler](#) did this in the meanwhile:

```
osc linkpac openSUSE:Leap:15.3 perl-Mojo-IOLoop-ReadWriteProcess devel:openqa:Leap:15.2
```

#18 - 2021-12-08 10:39 - cdywan

cdywan wrote:

I don't know why nobody took this. And I'm obviously not going to claim I'm "the" person working this, but this should not stay an unprocessed ticket.

^ Read: We don't want an immediate ticket with no assignee because 1. otherwise it looks to most stakeholders like we're not on top of it 2. there's no contact person 3. and nobody to ensure that clean-up is being done after the fact.

#19 - 2021-12-09 14:23 - cdywan

- Related to deleted (action #103611: test failure: os-autoinst: 29-backend-driver.t:31 Failed test 'exit logged')

#20 - 2021-12-09 14:23 - cdywan

- Blocked by action #103611: test failure: os-autoinst: 29-backend-driver.t:31 Failed test 'exit logged' added

#21 - 2021-12-09 14:32 - cdywan

- Status changed from Feedback to Blocked

- Priority changed from Urgent to Normal

I'm lowering the **Priority** here since the acute issue has been addressed, but **Blocked** on [#103611](#) since that includes the long-term fix.

#22 - 2021-12-09 14:37 - cdywan

- Subject changed from Pretty much all tests end 'timeout exceeded' to Most tests time out with perl-Mojo-IOLoop-ReadWriteProcess 0.28

#23 - 2021-12-13 11:55 - mkittler

- Status changed from Blocked to Workable

Since [#103611](#) has been resolved you could go ahead now.

#24 - 2021-12-17 15:33 - okurz

- Project changed from openQA Infrastructure to openQA Project

- Category set to Concrete Bugs

- Status changed from Workable to Resolved

not an "infrastructure" problem per se. This is just where we saw the symptoms. Links in devel repos have been reset.