

## openQA Project - action #73231

[microos]job incompletes with auto\_review:"backend died: Virtio terminal and svirt serial terminal do not support send\_key. Use"

2020-10-12 10:39 - Xiaojing\_liu

<b>Status:</b>	Resolved	<b>Start date:</b>	2020-10-12
<b>Priority:</b>	High	<b>Due date:</b>	
<b>Assignee:</b>	ggardet_arm	<b>% Done:</b>	0%
<b>Category:</b>	Concrete Bugs	<b>Estimated time:</b>	0.00 hour
<b>Target version:</b>	Ready		
<b>Difficulty:</b>			
<b>Description</b> <a href="https://openqa.opensuse.org/tests/1428196">https://openqa.opensuse.org/tests/1428196</a> is incompletes, the reason is 'Virtio terminal and svirt serial terminal do not support send_key. Use' More details please see: <a href="https://openqa.opensuse.org/tests/1428196/file/autoinst-log.txt">https://openqa.opensuse.org/tests/1428196/file/autoinst-log.txt</a>			
<b>Related issues:</b> Copied to openQA Project - action #75388: Explicit error feedback to test rev... <b>Workable</b>			

### History

#### #1 - 2020-10-12 11:15 - okurz

- Project changed from openQA Project to openQA Infrastructure
- Assignee set to ggardet\_arm
- Priority changed from Normal to High
- Target version set to Ready

interesting, could be linked to [#72079](#) , [ggardet\\_arm](#) can you also take a look into this while working on [#72079](#)? Maybe a different qemu version helps here as well?

#### #2 - 2020-10-12 11:32 - ggardet\_arm

- Assignee deleted (ggardet\_arm)

I have seen this error very sporadically, even *before* the upgrade.

#### #3 - 2020-10-12 11:55 - okurz

- Project changed from openQA Infrastructure to openQA Project
- Category set to Concrete Bugs
- Status changed from New to Workable

Problem seems to be caused by previous test modules failing, see the rollback ending in the log followed by log messages about a virtio console and then the error:

```
[0m[37m[2020-10-11T17:17:35.867 CEST] [debug] QEMU: Copyright (c) 2003-2019 Fabrice Bellard and the QEMU Project developers
[0m[2020-10-11T17:17:35.878 CEST] [debug] <<< consoles::virtio_terminal::open_pipe(pipe_prefix="/var/lib/openqa/pool/3/virtio_console")
[33m[2020-10-11T17:17:35.879 CEST] [info] ::: consoles::virtio_terminal::open_pipe: Set PIPE_SZ from 65536 to 1048576
[0m[33m[2020-10-11T17:17:35.879 CEST] [info] ::: consoles::virtio_terminal::open_pipe: Set PIPE_SZ from 65536 to 1048576
[0m[1;34m[2020-10-11T17:17:35.879 CEST] [debug] ||| starting health_check tests/transactional/health_check.pm
[0m[2020-10-11T17:17:35.881 CEST] [debug] tests/transactional/health_check.pm:42 called testapi::script_run
[2020-10-11T17:17:35.881 CEST] [debug] <<< testapi::script_run(cmd="rpm -q health-checker", quiet=undef, output=" ", timeout=undef)
[2020-10-11T17:17:35.882 CEST] [debug] tests/transactional/health_check.pm:42 called testapi::script_run
[2020-10-11T17:17:35.882 CEST] [debug] <<< testapi::type_string(string="rpm -q health-checker", max_interval=250, wait_screen_changes=0, wait_still_screen=0, timeout=30, similarity_level=47)
[2020-10-11T17:17:35.883 CEST] [debug] <<< consoles::serial_screen::type_string(cmd="backend_type_string", max_interval=250, json_cmd_token="QjmuvdRC", text="rpm -q health-checker")
```

```
[2020-10-11T17:17:35.883 CEST] [debug] tests/transactional/health_check.pm:42 called testapi::script_run
[2020-10-11T17:17:35.883 CEST] [debug] <<< testapi::type_string(string=""; echo Jhv3u-\$?- > /dev/ttyAMA0\n", max_interval=250, wait_screen_changes=0, wait_still_screen=0, timeout=30, similarity_level=47)
[2020-10-11T17:17:35.884 CEST] [debug] <<< consoles::serial_screen::type_string(text=""; echo Jhv3u-\$?- > /dev/ttyAMA0\n", json_cmd_token="vheCfIqe", max_interval=250, cmd="backend_type_string")
[2020-10-11T17:17:35.884 CEST] [debug] tests/transactional/health_check.pm:42 called testapi::script_run
[2020-10-11T17:17:35.885 CEST] [debug] <<< testapi::wait_serial(timeout=30, quiet=undef, no_regex=0, expect_not_found=0, record_output=undef, buffer_size=undef, regexp=qr/Jhv3u-\d+\/)
[2020-10-11T17:17:35.886 CEST] [debug] <<< consoles::serial_screen::read_until(json_cmd_token="CdDITowF", regexp="(?^:Jhv3u-\d+\/)", pattern="(?^:Jhv3u-\d+\/)", timeout=60, record_output=undef, expect_not_found=0, buffer_size=undef, cmd="backend_wait_serial", quiet=undef, no_regex=0)
[32m[2020-10-11T17:18:37.028 CEST] [debug] >>> testapi::wait_serial: (?^:Jhv3u-\d+\/): fail
[0m[2020-10-11T17:18:37.029 CEST] [debug] tests/transactional/health_check.pm:48 called testapi::validate_script_output
[2020-10-11T17:18:37.029 CEST] [debug] <<< testapi::type_string(string="cat - > /tmp/scriptIWeHA.sh;\n", max_interval=250, wait_screen_changes=0, wait_still_screen=0, timeout=30, similarity_level=47)
[2020-10-11T17:18:37.030 CEST] [debug] <<< consoles::serial_screen::type_string(cmd="backend_type_string", json_cmd_token="klzuekgN", max_interval=250, text="cat - > /tmp/scriptIWeHA.sh;\n")
[2020-10-11T17:18:37.031 CEST] [debug] tests/transactional/health_check.pm:48 called testapi::validate_script_output
[2020-10-11T17:18:37.031 CEST] [debug] <<< testapi::type_string(string="health-checker\n", max_interval=250, wait_screen_changes=0, wait_still_screen=0, timeout=30, similarity_level=47)
[2020-10-11T17:18:37.032 CEST] [debug] <<< consoles::serial_screen::type_string(cmd="backend_type_string", text="health-checker\n", max_interval=250, json_cmd_token="wEobsqaz")
[2020-10-11T17:18:37.032 CEST] [debug] tests/transactional/health_check.pm:48 called testapi::validate_script_output
[2020-10-11T17:18:37.032 CEST] [debug] <<< testapi::send_key(key="ctrl-d", wait_screen_change=0, do_wait=0)
[37m[2020-10-11T17:18:37.034 CEST] [debug] Backend process died, backend errors are reported below in the following lines:
Virtio terminal and svirt serial terminal do not support send_key. Use
type_string (possibly with an ANSI/XTERM escape sequence), or switch to a
console which sends key presses, not terminal codes.
at /usr/lib/os-autoinst/backend/baseclass.pm line 787.
```

compared to the last green <https://openqa.opensuse.org/tests/1426590/file/autoinst-log.txt> showing

```
[0m[37m[2020-10-10T07:08:51.154 UTC] [debug] ||| finished rebootmgr transactional at 2020-10-10 07:08:51 (785 s)
[0m[1;34m[2020-10-10T07:08:51.158 UTC] [debug] ||| starting health_check tests/transactional/health_check.pm
[0m[2020-10-10T07:08:51.159 UTC] [debug] tests/transactional/health_check.pm:42 called testapi::script_run
[2020-10-10T07:08:51.159 UTC] [debug] <<< testapi::script_run(cmd="rpm -q health-checker", timeout=undef, quiet=undef, output="")
[2020-10-10T07:08:51.160 UTC] [debug] tests/transactional/health_check.pm:42 called testapi::script_run
[2020-10-10T07:08:51.160 UTC] [debug] <<< testapi::type_string(string="rpm -q health-checker", max_interval=250, wait_screen_changes=0, wait_still_screen=0, timeout=30, similarity_level=47)
[2020-10-10T07:08:54.188 UTC] [debug] tests/transactional/health_check.pm:42 called testapi::script_run
[2020-10-10T07:08:54.189 UTC] [debug] <<< testapi::type_string(string=""; echo Jhv3u-\$?- > /dev/ttyAMA0\n", max_interval=250, wait_screen_changes=0, wait_still_screen=0, timeout=30, similarity_level=47)
[2020-10-10T07:08:59.296 UTC] [debug] tests/transactional/health_check.pm:42 called testapi::script_run
[2020-10-10T07:08:59.296 UTC] [debug] <<< testapi::wait_serial(quiet=undef, regexp=qr/Jhv3u-\d+\/, no_regex=0, expect_not_found=0, buffer_size=undef, timeout=30, record_output=undef)
[32m[2020-10-10T07:09:00.402 UTC] [debug] >>> testapi::wait_serial: (?^:Jhv3u-\d+\/): ok
[0m[2020-10-10T07:09:00.402 UTC] [debug] tests/transactional/health_check.pm:48 called testapi::validate_script_output
[2020-10-10T07:09:00.403 UTC] [debug] <<< testapi::type_string(string="cat - > /tmp/scriptIWeHA.sh;\n", max_interval=250, wait_screen_changes=0, wait_still_screen=0, timeout=30, similarity_level=47)
[2020-10-10T07:09:04.938 UTC] [debug] tests/transactional/health_check.pm:48 called testapi::validate_script_output
[2020-10-10T07:09:04.938 UTC] [debug] <<< testapi::type_string(string="health-checker\n", max_interval=250, wait_screen_changes=0, wait_still_screen=0, timeout=30, similarity_level=47)
[2020-10-10T07:09:07.101 UTC] [debug] tests/transactional/health_check.pm:48 called testapi::validate_script_output
[2020-10-10T07:09:07.101 UTC] [debug] <<< testapi::send_key(key="ctrl-d", wait_screen_change=0, do_wait=0)
[2020-10-10T07:09:07.874 UTC] [debug] tests/transactional/health_check.pm:48 called testapi::validate_script_output
[2020-10-10T07:09:07.875 UTC] [debug] <<< testapi::type_string(string="(echo IWeHA; bash -eox pipefail /tmp/scriptIWeHA.sh ; echo SCRIPT_FINISHEDIWeHA-\$?-) | tee /dev/ttyAMA0\n", max_interval=250, wait_screen_changes=0, wait_still_screen=0, timeout=30, similarity_level=47)
[2020-10-10T07:09:25.366 UTC] [debug] tests/transactional/health_check.pm:48 called testapi::validate_script_output
[2020-10-10T07:09:25.367 UTC] [debug] <<< testapi::wait_serial(timeout=30, record_output=1, no_regex=0, quiet=undef, expect_not_found=0, regexp="SCRIPT_FINISHEDIWeHA-\d+\/", buffer_size=undef)
[32m[2020-10-10T07:09:27.477 UTC] [debug] >>> testapi::wait_serial: SCRIPT_FINISHEDIWeHA-\d+\/: ok
```

both jobs have VIRTIO\_CONSOLE=1

The `send_key` call seems to come from `valid_script_output` that calls `script_output` which has the code:

```
if (testapi::is_serial_terminal) {
  my $heretag = 'EOT_' . $marker;
  my $cat = "cat > $script_path << '$heretag'; echo $marker-\$?-";
  testapi::wait_serial($self->{serial_term_prompt}, no_regex => 1, quiet => $args{quiet});
  bmwgemu::log_call("Content of $script_path :\n \"\$cat\" \n");
  testapi::type_string($cat . "\n");
  testapi::wait_serial("$cat", no_regex => 1, quiet => $args{quiet});
  # Wait for input prompt of here tag before typing $script. This avoids
  # messy output, like duplicate output of $script. We do this in a second
  # wait_serial() call, to avoid issues during new line detection.
  testapi::wait_serial('> ', no_regex => 1, quiet => $args{quiet});
  testapi::type_string("$script\n$heretag\n");
  testapi::wait_serial("> $heretag", no_regex => 1, quiet => $args{quiet});
  testapi::wait_serial("$marker-0-", quiet => $args{quiet});
}
elseif ($args{type_command}) {
  my $cat = "cat - > $script_path;\n";
  testapi::type_string($cat);
  testapi::type_string($script . "\n");
  testapi::send_key('ctrl-d');
}
```

so it seems that `testapi::is_serial_terminal` evaluates to false here. Maybe the problem is related to the test module `tests/transactional/health_check.pm` does not have an explicit `select_console` call. Now I am asking myself: Clearly this is not a "product bug" but should the job incomplete or fail? The "incomplete" is not commonly seen as a problem that would need to be fixed in the test distribution. This might be even more so because in the reason it says "backend died" and not "tests died". I suggest we change that to say "tests died".

#### #4 - 2020-10-16 04:23 - Xiaojing\_liu

[okurz](#) Can we change `send_key('ctrl-d')` to `type_string('exit\n')` ?

#### #5 - 2020-10-16 07:16 - ggardet\_arm

PR to make sure we have selected the root console for `health_check` and `openqa-review`:

<https://github.com/os-autoinst/os-autoinst-distrib-opensuse/pull/11215>

#### #6 - 2020-10-19 09:02 - okurz

Xiaojing\_liu wrote:

[okurz](#) Can we change `send_key('ctrl-d')` to `type_string('exit\n')` ?

no, we should not. That is more characters to type and can potentially more likely fail and I don't see it as necessary. This is not the problem here, the code should not have been called in this case at all. Clearly the test code is wrong and `ggardet_arm` fixed that. What we should do here is what I suggested in [#73231#note-3](#) at the very end, improve the error output and make it explicit and obvious to the test reviewers that they need to act and not that `openQA` code is broken.

[ggardet\\_arm](#) thx, PR merged.

#### #7 - 2020-10-26 06:50 - Xiaojing\_liu

- Status changed from *Workable* to *In Progress*

- Assignee set to *Xiaojing\_liu*

#### #8 - 2020-10-27 08:06 - Xiaojing\_liu

- Status changed from *In Progress* to *Workable*

- Assignee deleted (*Xiaojing\_liu*)

#### #9 - 2020-10-27 08:18 - okurz

- Copied to action [#75388](#): *Explicit error feedback to test reviewers on wrong test API usage added*

#### #10 - 2020-10-27 08:19 - okurz

- Status changed from *Workable* to *Resolved*

- Assignee set to *ggardet\_arm*

As the actual error in tests was fixed by `ggardet_arm` we do not need to regard this as high anymore. Also I tried to clarify better what we could do with an explicit feature request in [#75388](#)