openQA Tests - action #50345

[functional][u] Usage of kernel parameter kernel.softlockup_panic

2019-04-12 11:01 - SLindoMansilla

<table>
<thead>
<tr>
<th>Status:</th>
<th>Resolved</th>
<th>Start date:</th>
<th>2019-04-12</th>
</tr>
</thead>
<tbody>
<tr>
<td>Priority:</td>
<td>High</td>
<td>Due date:</td>
<td>2019-05-21</td>
</tr>
<tr>
<td>Assignee:</td>
<td>szarate</td>
<td>% Done:</td>
<td>0%</td>
</tr>
<tr>
<td>Category:</td>
<td>Enhancement to existing tests</td>
<td>Estimated time:</td>
<td>42.00 hours</td>
</tr>
<tr>
<td>Target version:</td>
<td>Milestone 30</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Difficulty:</td>
<td>easy</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Description**

**Motivation**

We get recently a lot of failures with error message "workqueue lockup" (see bsc#1126782). This is a symptom and we need to find the different causes for it.

There is a suggestion to enable kernel panic debugging: [https://bugzilla.suse.com/show_bug.cgi?id=1126782#c12](https://bugzilla.suse.com/show_bug.cgi?id=1126782#c12)

We need to determine if we can use this parameter in general without affecting the tests and how to properly use it. Or if we need to manually reproduce the issue with this kernel parameter.

**Related issues:**

- Related to openQA Tests - action #47849: [functional][u] test if system_worka...  Resolved  2019-02-13
- Blocks openQA Tests - action #66607: [functional][u] Execute "SysRq t" when w...  Resolved  2019-02-13

**History**

#1 - 2019-04-12 11:01 - SLindoMansilla

To be discussed in the next grooming meeting.

#2 - 2019-04-23 13:50 - SLindoMansilla

Related PRs:

- [https://github.com/os-autoinst/os-autoinst-distri-opensuse/pull/7177](https://github.com/os-autoinst/os-autoinst-distri-opensuse/pull/7177)
- [https://github.com/os-autoinst/os-autoinst-distri-opensuse/pull/7260](https://github.com/os-autoinst/os-autoinst-distri-opensuse/pull/7260)

#3 - 2019-05-13 09:37 - SLindoMansilla

- Status changed from New to Workable
- Assignee set to szarate

#4 - 2019-05-13 15:33 - szarate

- Status changed from Workable to In Progress

Trying out here: https://openqa.suse.de/tests/overview?build=rogue_workqueue_bsc1126782_investigation&distri=sle&version=15-SP1. Doesn't seem like it's going to be a problem for the tests, however it's better to use kernel.softlockup_panic=1 after GMC is out on sle15, Next step will be enabling it for SLE12 SP5, first via EXTRABOOTPARAMS and then via test code once we're sure enough that we want this.

#5 - 2019-05-14 19:48 - szarate

Doesn't look like it's gonna hurt - so far -, will do more tests tomorrow and see

#6 - 2019-05-17 14:44 - szarate

- Due date set to 2019-05-21
- Status changed from In Progress to Feedback
- Priority changed from High to Normal
- Target version set to Milestone 25

For the time being, I've created a new test suite: create_hdd_minimal_base+sdk+softlockup for SLE12-SP5 for the time being. It's in the development
job group so matter to wait for the next build, and see how it goes. I did trigger few runs and none of them seemed to fail with this.

I’ll look a bit later next week to see how to manually trigger softlockups and see what’s the behaviour...

Plan for this ticket is to simply once the HDD’s are created, schedule some 100 runs of modules that triggered the problem in the past, see if it can be reproduced and then move on.

But for now, look like we can get the parameter in shape to put in the test code.

Setting due date so that I get a reminder to look at this next week.

#7 - 2019-05-21 14:55 - SLindoMansilla
waiting for next build

#8 - 2019-05-28 09:14 - szarate
So according to: create_hdd_minimal_base+sdk+softlockup in the last run: https://openqa.suse.de/tests/overview?distri=sle&version=12-SP5&build=0170&groupid=132 Enabling the parameter does not hurt.

#9 - 2019-05-28 09:25 - szarate
- Status changed from Feedback to In Progress

Next up trigger a softlockup and see what happens :)

Potential problem with this approach is: https://www.suse.com/support/kb/doc/?id=7023049

#10 - 2019-06-04 07:14 - szarate
So it definitely does produce a panic, which is what we want, however... the job gets stuck (possibly detecting the serial error, would suffice to get the machine to shutdown) My main concern is that if we follow this path... there might be a lot of tests that would fail due to how often these softlockup could end up happening...

I used https://github.com/foursixnine/softlockup_test this kernel module

```
susetest login: [ 1095.966023] watchdog: BUG: soft lockup - CPU#0 stuck for 22s! [softlockup_thre:15996] [ 1095.966836] Kernel panic - not syncing: softlockup: hung tasks [ 1095.967377] CPU: 0 PID: 15996 Comm: softlockup_thre Tainted: G 5.1.4-1-default #1 openSUSE Tumbleweed (unreleased) [ 1095.968465] Hardware name: QEMU Standard PC (i440FX + PIIX, 1996), BIOS rel-1.12.0-0-ga698c89-prebuilt.qemu.org 04/01/2014 [ 1095.969471] Call Trace: [ 1095.969505] <IRQ> [ 1095.969896] panic+0x85/0xc0 [ 1095.970207] ? ret_from_fork+0x11/0x50 [ 1095.970834] ? ret_from_fork+0x45/0x50 [ 1095.971184] watchdog_timer_fn.cold.6+0x16/0x1e [ 1095.971603] ? softlockup_fn+0x40/0x40 [ 1095.972077] __hrtimer_run_queues+0x66a/0x140 [ 1095.972874] hrtimer_interrupt+0x100/0x220 [ 1095.973026] kthread+0x116/0x130 [ 1095.973479] <IRQ> [ 1095.973791] RIP: 0010:console_unlock+0x41f/0x550 [ 1095.974218] Code: fc ff ff c7 05 0a d0 0b 01 00 00 00 00 48 83 c4 08 5b 5d 41 5c 41 5d 41 5e 41 5f e9 8b f7 ff ff e8 26 35 09 00 4c 89 df 57 9d <DF> 1f 44 00 00 e9 25 ff ff ff ff ff 65 8b 05 60 b8 f1 54 89 c0 48 0f a3 [ 1095.975399] vprintk_emit+0x1ab/0x250 [ 1095.975939] RSP: 0018:console_unlock+0x41f/0x550 [ 1095.976362] Code: fc ff ff c7 05 0a d0 0b 01 00 00 00 00 48 83 c4 08 5b 5d 41 5c 41 5d 41 5e 41 5f e9 8b f7 ff ff e8 26 35 09 00 4c 89 df 57 9d <DF> 1f 44 00 00 e9 25 ff ff ff ff ff 65 8b 05 60 b8 f1 54 89 c0 48 0f a3 [ 1095.976632] RX: 0000000000000002 RX: ffffffff0000000f [ 1095.977286] RDX: 0000000000000001 RSI: 0000000000000000 [ 1095.977939] RBP: 00000000004d171eb R08: 00000000004d171eb R09: 00000000004d171eb R10: 00000000004d171eb [ 1095.978593] R10: 0000000000000000 R11: 0000000000000000 R12: 0000000000000000 [ 1095.979248] R13: 00000000004d171eb R14: 0000000000000000 [ 1095.979916] vprintk_emit+0x1ab/0x250 [ 1095.980261] ? 0xffffffffc0a3b000 [ 1095.980571] printk+0x48/0x4a [ 1095.980851] task+0x62/0x6b [softlockup_test] [ 1095.981257] kthread+0x116/0x130 [ 1095.981559] ? kthread_bind+0x30/0x30 [ 1095.981903] ret_from_fork+0x4a/0x50 [ 1095.982382] Kernel Offset: 0x2a000000 from 0xffffffff81000000 (relocation range: 0xffffffff80000000-0xffffffff)
```

2021-05-22
After talking with Takashi for a brief moment, he suggested to enable the parameter along with kdump, so I guess that the workflow would be something like:

enable kdump as early as possible in the test if it's available (Could start with extratests), and enable the kernel parameter... I'm guessing that I'd have to look for a certain timeout and on top of that figure out how to get more information as requested on: https://bugzilla.suse.com/show_bug.cgi?id=1130701

So as agreed in an offline meeting, I'm creating a new test module to enable boot parameters via sysctl, and on the other hand creating a ticket so that boot_to_desktop is able to enable EXTRABOOTPARAMS if they are set, which it currently doesn't.

Ideally, for at least one scenario, https://github.com/os-autoinst/os-autoinst-distri-opensuse/pull/7771 the early_test_setup is enabled right after the system has booted, on all of our architectures, duplicating at least one of the extra_tests scenarios (the big ones) and let it run for two or three builds, and see if nothing breaks too bad. Or Simply merge my pr, and see what breaks :D

Ideally that module could be extended further ahead in the future.
#25 - 2020-05-11 09:10 - SLindoMansilla
- Priority changed from Normal to High

#26 - 2020-05-11 09:10 - SLindoMansilla
- Blocks action #66607: [functional][u] Execute "SysRq t" when workqueue lockup is detected and publish kernel logs added

#27 - 2020-05-11 09:45 - szarate
- Category changed from Spike/Research to Enhancement to existing tests

#28 - 2020-05-11 09:45 - szarate
szarate wrote:

I'm unassigning from this for now, however I'd appreciate if anybody wants to pick this up...

Ideally, for at least one scenario, https://github.com/os-autoinst/os-autoinst-distri-opensuse/pull/7771 the early_test_setup is enabled right after the system has booted, on all of our architectures, duplicating at least one of the extra_tests scenarios (the big ones) and let it run for two or three builds, and see if nothing breaks too bad. Or Simply merge my pr, and see what breaks :D

Ideally that module could be extended further ahead in the future.

This can be easily implemented by adding a new branch at bootloader_setup::specific_bootmenu_params

```plaintext
# Kernel softlockup panic should be enabled, unless explicitly disabled
# See bsc#1126782
if (!get_var("SOFTLOCKUP_PANIC_DISABLED", 0)) {
    push @params, "kernel.softlockup_panic=1";
}
```

In order to verify that it properly works:

A create_hdd job should be triggered, and a corresponding extra_test job should follow

#29 - 2020-05-11 09:53 - szarate
- Difficulty set to easy

#30 - 2020-05-18 13:15 - szarate
- Status changed from Workable to In Progress

Well I'm picking this one, gonna see if I can piggyback triggering tests from GH automatically

#31 - 2020-06-04 10:09 - zluo
- Status changed from In Progress to Workable

unassigned but in progress. change it now to workalbe

#32 - 2020-06-04 10:13 - szarate
- Status changed from Workable to In Progress
- Assignee set to szarate

Better?

#33 - 2020-08-04 07:19 - szarate
- Status changed from In Progress to Resolved

PR finally merged since few days ago, no failing jobs so far