

openQA Infrastructure - action #80834

[alert] osd reported 502 errors, unresponsive, failing alerts for CPU usage since 2020-12-08 0840Z and minion jobs failed

2020-12-08 09:00 - okurz

Status:	Resolved	Start date:	2020-12-08
Priority:	Immediate	Due date:	
Assignee:	okurz	% Done:	0%
Category:		Estimated time:	0.00 hour
Target version:	Ready		
Description			
Observation			
grafana alerts "CPU usage" on osd and "Minion Jobs" alerted since 2020-12-08 0840Z. Users in RC channel #testing reported problems to reach OSD.			
Related issues:			
Related to openSUSE admin - tickets #44060: Kind request for access to manage...		Rejected	2018-11-20

History

#1 - 2020-12-08 09:04 - okurz

During <https://stats.openqa-monitor.qa.suse.de/d/WebuiDb/webui-summary?orgId=1&editPanel=23&tab=alert&from=1607416466938&to=1607417457072> there was a sudden raise in "IOWait" CPU usage, i.e. processes are blocked trying to read/write I/O. ps reports processes in blocked state:

```
# ps auxf | grep '\<D\>'
root      670  0.3  0.0      0   0 ?        D   Dec06 10:51  \_ [xfssaid/vdd]
root     14335  0.0  0.0      0   0 ?        D   08:01  0:02  \_ [kworker/7:0]
root     23524  0.1  0.0      0   0 ?        D   09:23  0:02  \_ [kworker/u20:2]
root     30113  0.0  0.0      0   0 ?        D   09:39  0:00  \_ [kworker/0:3]
geekote+ 20532  4.4  1.1 414096 238168 ?        D   08:12  4:51 /usr/bin/perl /usr/share/openqa/script/openqa
prefork -m production --proxy -i 100 -H 400 -w 20 -c 1 -G 800
geekote+ 23015  5.0  1.1 422064 239708 ?        D   08:19  5:13 /usr/bin/perl /usr/share/openqa/script/openqa
prefork -m production --proxy -i 100 -H 400 -w 20 -c 1 -G 800
geekote+ 32140  3.2  1.1 417396 241620 ?        D   08:43  2:33 /usr/bin/perl /usr/share/openqa/script/openqa
prefork -m production --proxy -i 100 -H 400 -w 20 -c 1 -G 800
geekote+ 6023  5.3  1.0 391008 215356 ?        D   08:53  3:41 /usr/bin/perl /usr/share/openqa/script/openqa
prefork -m production --proxy -i 100 -H 400 -w 20 -c 1 -G 800
geekote+ 9891  5.0  1.1 416720 234860 ?        D   08:56  3:17 /usr/bin/perl /usr/share/openqa/script/openqa
prefork -m production --proxy -i 100 -H 400 -w 20 -c 1 -G 800
geekote+ 12971  5.8  1.1 413940 232012 ?        D   09:01  3:34 /usr/bin/perl /usr/share/openqa/script/openqa
prefork -m production --proxy -i 100 -H 400 -w 20 -c 1 -G 800
geekote+ 13764  4.7  1.1 419216 237660 ?        D   09:03  2:48 /usr/bin/perl /usr/share/openqa/script/openqa
prefork -m production --proxy -i 100 -H 400 -w 20 -c 1 -G 800
geekote+ 15173  4.3  1.0 396528 222460 ?        D   09:08  2:22 /usr/bin/perl /usr/share/openqa/script/openqa
prefork -m production --proxy -i 100 -H 400 -w 20 -c 1 -G 800
geekote+ 15967  4.8  1.0 398824 225012 ?        D   09:10  2:32 /usr/bin/perl /usr/share/openqa/script/openqa
prefork -m production --proxy -i 100 -H 400 -w 20 -c 1 -G 800
geekote+ 17182  4.5  1.1 399988 226096 ?        D   09:14  2:11 /usr/bin/perl /usr/share/openqa/script/openqa
prefork -m production --proxy -i 100 -H 400 -w 20 -c 1 -G 800
geekote+ 21583  2.8  1.0 388732 212756 ?        D   09:18  1:13 /usr/bin/perl /usr/share/openqa/script/openqa
prefork -m production --proxy -i 100 -H 400 -w 20 -c 1 -G 800
geekote+ 22135  4.6  1.1 409796 227904 ?        D   09:19  1:57 /usr/bin/perl /usr/share/openqa/script/openqa
prefork -m production --proxy -i 100 -H 400 -w 20 -c 1 -G 800
geekote+ 22762  4.0  1.0 390900 216960 ?        D   09:21  1:38 /usr/bin/perl /usr/share/openqa/script/openqa
prefork -m production --proxy -i 100 -H 400 -w 20 -c 1 -G 800
geekote+ 22794  4.6  1.0 384316 210212 ?        D   09:21  1:53 /usr/bin/perl /usr/share/openqa/script/openqa
prefork -m production --proxy -i 100 -H 400 -w 20 -c 1 -G 800
geekote+ 23177  1.9  0.9 362996 185532 ?        D   09:22  0:45 /usr/bin/perl /usr/share/openqa/script/openqa
prefork -m production --proxy -i 100 -H 400 -w 20 -c 1 -G 800
geekote+ 25175  4.2  1.0 380852 207000 ?        D   09:24  1:38 /usr/bin/perl /usr/share/openqa/script/openqa
prefork -m production --proxy -i 100 -H 400 -w 20 -c 1 -G 800
geekote+ 26111  1.7  0.9 364648 187208 ?        D   09:27  0:35 /usr/bin/perl /usr/share/openqa/script/openqa
prefork -m production --proxy -i 100 -H 400 -w 20 -c 1 -G 800
geekote+ 26165  3.1  1.0 378648 204840 ?        D   09:27  1:07 /usr/bin/perl /usr/share/openqa/script/openqa
prefork -m production --proxy -i 100 -H 400 -w 20 -c 1 -G 800
```

```

geekote+ 28975 1.4 0.9 368404 193120 ?      D   09:36   0:23 /usr/bin/perl /usr/share/openqa/script/openqa
prefork -m production --proxy -i 100 -H 400 -w 20 -c 1 -G 800
geekote+ 29942 0.4 0.8 353396 174544 ?      D   09:39   0:05 /usr/bin/perl /usr/share/openqa/script/openqa
prefork -m production --proxy -i 100 -H 400 -w 20 -c 1 -G 800
geekote+ 32733 0.0 0.8 350232 170508 ?      D   09:46   0:00 /usr/bin/perl /usr/share/openqa/script/openqa
prefork -m production --proxy -i 100 -H 400 -w 20 -c 1 -G 800
geekote+ 32757 0.0 0.8 350232 170604 ?      D   09:46   0:00 /usr/bin/perl /usr/share/openqa/script/openqa
prefork -m production --proxy -i 100 -H 400 -w 20 -c 1 -G 800

```

seems like the storage does not respond.

Reported ticket <https://infra.nue.suse.com/SelfService/Display.html?id=182058> and mentioned in [#suse-it-ama](#)

EDIT: If I attach to one process, e.g. `strace -f -p 20532` I see nothing showing up.

All of the following still works:

```

touch /tmp/foo
touch /foo
touch /assets/foo
touch /space-slow/foo
touch /results/foo
touch /srv/foo
rm -f /tmp/foo /foo /assets/foo /space-slow/foo /results/foo /srv/foo
for i in /tmp/foo /foo /assets/foo /space-slow/foo /results/foo /srv/foo; do echo $i; dd bs=1M count=1 if=/dev
/zero of=$i; done

```

top says

```
%Cpu(s):  0.1 us,  0.1 sy,  0.0 ni,  0.0 id, 99.7 wa,  0.0 hi,  0.0 si,  0.1 st
```

iotstat -x 2 5 shows that the first report has high numbers for all statistics since the last boot but all four subsequent reports show near to no I/O action, so nearly nothing is going on. Also in iotop I don't see anything really reading or writing.

cat /proc/20532/io shows that the openQA webui process mentioned above was reading+writing but is not anymore, the report is static on:

```

rchar: 1700440441
wchar: 1175769606
syscr: 36578
syscw: 91864
read_bytes: 213331968
write_bytes: 1039572992
cancelled_write_bytes: 642121728

```

I don't see anything interesting in `lsdf -p 20532`, no files open from any external storage as it seems

```

#
COMMAND  PID    USER   FD    TYPE    DEVICE  SIZE/OFF      NODE NAME
openqa   20532  geekotest cwd    DIR      253,1    4096          2 /
openqa   20532  geekotest rtd    DIR      253,1    4096          2 /
openqa   20532  geekotest txt    REG      253,1    2099808      81338 /usr/bin/perl
openqa   20532  geekotest mem    REG      253,1    26408      40882 /lib64/libnss_dns-2.26.so
openqa   20532  geekotest mem    REG      253,1    27192      177731 /usr/lib/perl5/5.26.1/x86_64-linux-th
read-multi/auto/PerlIO/encoding/encoding.so
...
openqa   20532  geekotest mem    REG      253,1    18904      178243 /usr/lib/perl5/5.26.1/x86_64-linux-th
read-multi/auto/mro/mro.so
openqa   20532  geekotest mem    REG      253,1    2038456     39871 /lib64/libc-2.26.so
openqa   20532  geekotest mem    REG      253,1    144048     40951 /lib64/libpthread-2.26.so
openqa   20532  geekotest mem    REG      253,1    57448     40462 /lib64/libcrypt-2.26.so
openqa   20532  geekotest mem    REG      253,1    18400     40529 /lib64/libdl-2.26.so
openqa   20532  geekotest mem    REG      253,1    1355760     40535 /lib64/libm-2.26.so
openqa   20532  geekotest mem    REG      253,1    180056     40288 /lib64/ld-2.26.so
openqa   20532  geekotest mem    REG      253,1    330604     182205 /usr/lib/locale/en_US.utf8/LC_CTYPE
openqa   20532  geekotest mem    REG      253,1    26244     27665 /usr/lib64/gconv/gconv-modules.cache
openqa   20532  geekotest 0r     CHR      1,3        0t0         1029 /dev/null
openqa   20532  geekotest 1u     unix 0xffff880623993c00    0t0         39941 type=STREAM
openqa   20532  geekotest 2u     unix 0xffff880623993c00    0t0         39941 type=STREAM
openqa   20532  geekotest 3u     IPv6      63106972    0t0         TCP localhost:9526->localhost:47556 (CLOS
E_WAIT)
openqa   20532  geekotest 4u     unix 0xffff880625edd000    0t0         37304 type=STREAM
openqa   20532  geekotest 5r     REG      253,1    14590     199991 /usr/lib/perl5/vendor_perl/5.26.1/Moj
olicious/Plugin/AssetPack/Store.pm
openqa   20532  geekotest 6r     REG      253,1    38688     178066 /usr/share/openqa/assets/images/logo.
svg

```

```

openqa 20532 geekotest 7w FIFO 0,12 0t0 35294 pipe
openqa 20532 geekotest 15u IPv4 35296 0t0 TCP localhost:9526 (LISTEN)
openqa 20532 geekotest 16u IPv6 35298 0t0 TCP localhost:9526 (LISTEN)
openqa 20532 geekotest 17u unix 0xfffff880238523400 0t0 61487507 type=STREAM
openqa 20532 geekotest 18u unix 0xfffff88011bdea800 0t0 61577791 type=STREAM
openqa 20532 geekotest 19u IPv4 63106970 0t0 TCP localhost:56552->localhost:9527 (CLOS
E_WAIT)
...
openqa 20532 geekotest 170r REG 253,1 578 177997 /usr/share/openqa/assets/images/logo-
16.png

```

likely the kernel threads are more interesting:

```

root 14335 0.0 0.0 0 0 ? D 08:01 0:02 \_ [kworker/7:0]
root 23524 0.0 0.0 0 0 ? D 09:23 0:02 \_ [kworker/u20:2]
root 30113 0.0 0.0 0 0 ? D 09:39 0:00 \_ [kworker/0:3]

```

```

# for i in 14335 23524 30113 ; do echo "## PID: $i" && echo "### stack:" && cat /proc/$i/stack && echo "### io
:" && cat /proc/$i/io; done

```

```

## PID: 14335
### stack:
[<ffffffff810b9d82>] io_schedule+0x12/0x40
[<ffffffff814207b3>] wbt_wait+0x1b3/0x3c0
[<ffffffff813f10a6>] blk_mq_make_request+0xd6/0x570
[<ffffffff813e46e2>] generic_make_request+0x182/0x3e0
[<ffffffff813e49ac>] submit_bio+0x6c/0x140
[<ffffffffffa089f04a>] _xfs_buf_ioapply+0x2fa/0x4a0 [xfs]
[<ffffffffffa08a0b21>] xfs_buf_submit+0x61/0x210 [xfs]
[<ffffffffffa08c31d4>] xlog_bdstrat+0x24/0x50 [xfs]
[<ffffffffffa08c4eaf>] xlog_sync+0x2bf/0x3b0 [xfs]
[<ffffffffffa08c5ac0>] xlog_write+0x490/0x680 [xfs]
[<ffffffffffa08c74f8>] xlog_cil_push+0x258/0x3a0 [xfs]
[<ffffffffff810a794a>] process_one_work+0x1da/0x400
[<ffffffffff810a7d8f>] worker_thread+0x21f/0x3f0
[<ffffffffff810ae34d>] kthread+0x10d/0x130
[<ffffffffff81800242>] ret_from_fork+0x22/0x40
[<ffffffffffxffffffff>] 0xffffffffffffffff
### io:
rchar: 0
wchar: 0
syscr: 0
syscw: 0
read_bytes: 16384
write_bytes: 0
cancelled_write_bytes: 0
## PID: 23524
### stack:
[<ffffffff810b9d82>] io_schedule+0x12/0x40
[<ffffffff814207b3>] wbt_wait+0x1b3/0x3c0
[<ffffffff813f10a6>] blk_mq_make_request+0xd6/0x570
[<ffffffff813e46e2>] generic_make_request+0x182/0x3e0
[<ffffffffff813e49ac>] submit_bio+0x6c/0x140
[<ffffffffffa0899734>] xfs_submit_ioend+0x84/0x1e0 [xfs]
[<ffffffffffa089a14a>] xfs_vm_writepages+0x6a/0x80 [xfs]
[<ffffffffff811d681c>] do_writepages+0x3c/0xd0
[<ffffffffff8128e60d>] __writeback_single_inode+0x3d/0x320
[<ffffffffff8128eea5>] writeback_sb_inodes+0x185/0x480
[<ffffffffff8128f1fd>] __writeback_inodes_wb+0x5d/0xb0
[<ffffffffff8128f54e>] wb_writeback+0x23e/0x2c0
[<ffffffffff812901e6>] wb_workfn+0x216/0x400
[<ffffffffff810a794a>] process_one_work+0x1da/0x400
[<ffffffffff810a7b9b>] worker_thread+0x2b/0x3f0
[<ffffffffff810ae34d>] kthread+0x10d/0x130
[<ffffffffff81800242>] ret_from_fork+0x22/0x40
[<ffffffffffxffffffff>] 0xffffffffffffffff
### io:
rchar: 0
wchar: 0
syscr: 0
syscw: 0
read_bytes: 1519616
write_bytes: 0
cancelled_write_bytes: 0
## PID: 30113
### stack:

```

```
[<ffffffff810a83cd>] flush_work+0x10d/0x1c0
[<ffffffffffa08c7e08>] xlog_cil_force_lsn+0x68/0x1e0 [xfs]
[<ffffffffffa08c5dfc>] _xfs_log_force+0x7c/0x280 [xfs]
[<ffffffffffa08c609f>] xfs_log_worker+0x2f/0xf0 [xfs]
[<ffffffffff810a794a>] process_one_work+0x1da/0x400
[<ffffffffff810a7b9b>] worker_thread+0x2b/0x3f0
[<ffffffffff810ae34d>] kthread+0x10d/0x130
[<ffffffffff81800242>] ret_from_fork+0x22/0x40
[<ffffffffffffffff>] 0xffffffffffffffff
### io:
rchar: 0
wchar: 0
syscr: 0
syscw: 0
read_bytes: 0
write_bytes: 0
cancelled_write_bytes: 0
```

so stuck on xfs? But for which filesystem?

further

```
# for i in 14335 23524 30113 ; do echo "## PID: $i" && echo "### syscall:" && cat /proc/$i/syscall; done
## PID: 14335
### syscall:
0 0x0 0x0 0x0 0x0 0x0 0x0 0x0 0x0
## PID: 23524
### syscall:
0 0x0 0x0 0x0 0x0 0x0 0x0 0x0 0x0
## PID: 30113
### syscall:
0 0x0 0x0 0x0 0x0 0x0 0x0 0x0 0x0
```

what about the "xfsaild" process?:

```
# for i in 670 ; do echo "## PID: $i" && echo "### stack:" && cat /proc/$i/stack && echo "### io:" && cat /proc/$i/io && echo "### lsof:" && lsof -p $i && echo "### syscall:" && cat /proc/$i/syscall ; done
## PID: 670
### stack:
[<ffffffff810a83cd>] flush_work+0x10d/0x1c0
[<ffffffffffa08c7e08>] xlog_cil_force_lsn+0x68/0x1e0 [xfs]
[<ffffffffffa08c5dfc>] _xfs_log_force+0x7c/0x280 [xfs]
[<ffffffffffa08d33f6>] xfsaild+0x1a6/0x780 [xfs]
[<ffffffffff810ae34d>] kthread+0x10d/0x130
[<ffffffffff81800242>] ret_from_fork+0x22/0x40
[<ffffffffffffffff>] 0xffffffffffffffff
### io:
rchar: 0
wchar: 0
syscr: 0
syscw: 0
read_bytes: 41573335040
write_bytes: 0
cancelled_write_bytes: 0
### lsof:
COMMAND  PID USER  FD      TYPE DEVICE SIZE/OFF NODE NAME
xfsaild/v 670 root  cwd     DIR    253,1  4096    2 /
xfsaild/v 670 root  rtd     DIR    253,1  4096    2 /
xfsaild/v 670 root  txt     unknown                /proc/670/exe
### syscall:
0 0x0 0x0 0x0 0x0 0x0 0x0 0x0 0x0
```

#2 - 2020-12-08 09:44 - okurz

I pinged again in [#suse-it-ama](#) and [esujskaja](#) responded "Oliver Kurz noted".

#3 - 2020-12-08 10:51 - okurz

- Related to tickets #44060: Kind request for access to management interface of ariel VM, e.g. ssh-access, libvirt, "ssh user access to atreju1 to be able to view cscreen" added

#4 - 2020-12-08 11:04 - okurz

Finally got in contact with gschlotter:

- 1014Z gschlotter contacted me over RC
- 1015Z I responded with suggestion to force a reset of the VM
- 1017Z gschlotter forced shutdown and boot, I could login over ssh. All services started up correctly within 1m

So unfortunately there was bit of waiting time between me communicating the problem to EngInfra until their helpful reaction 0904Z-1014Z so 50m. That should be still considered ok but still something to improve :)

gschlotter could see in the serial output that the last messages were from the system shutdown:

```
[ OK ] Stopped Monitoring of LVM2 mirrors,...sing dmeventd or progress polling.
Stopping LVM2 metadata daemon...
[ OK ] Stopped LVM2 metadata daemon.
[ OK ] Reached target Shutdown.
[198553.813575] systemd-journald[537]: Failed to send WATCHDOG=1 notification message: Connection refused
[198643.816898] systemd-journald[537]: Failed to send WATCHDOG=1 notification message: Transport endpoint is not connected
```

so the machine was stuck in shutdown and rebooted fine after that. Maybe a hard echo b > /proc/sysrq-trigger would have worked out better than "reboot".

I asked again if we can improve the situation in <https://infra.nue.suse.com/SelfService/Display.html?id=182058>

Would it be possible for you to reconsider providing some controlling access to either the VM or the hypervisor? At least it should be possible to have access to the serial console over ssh (optionally with some restricted user accounts). Maybe over the serial console we would have been able to force a reboot using magic-sysrq.

Lessons learned

- Fast reaction is good. (Unfortunately) pinging people from EngInfra in RC helped better than creating a ticket.
- After the VM booted everything was coming up just fine without any further manual interaction needed. So we did good in developing reboot-safe setups including automatic retriggers of incomplete jobs that got abandoned, etc.
- We could have reduced the downtime significantly if we could have direct controlling access to the VM which we do not have, see [#44060](#) and [#43970](#) . Our proposals are unfortunately not accepted by EngInfra due to limitations in permission restrictions.

#5 - 2020-12-08 11:04 - okurz

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