Linux Suspend/Resume

Len Brown, Rafael Wysocki, Todd Brandt - Intel
Rafael Wysocki

Linux Kernel PM Maintainer
Todd Brandt
Sleepgraph Maintainer
Today's Session

1. Level Set
2. sleepgraph tutorial
3. Discussion / Q&A Topics
Part 1 - Level Set
Linux System Suspend Types

$ cat /sys/power/state
disk mem standby freeze

Power savings → Speed
Linux System Suspend to "mem" Default

```bash
$ cat /sys/power/mem_sleep
s2idle [deep]
```
System Suspend Options

- Busy
- Active Idle
- System Suspend to Low-Power Idle
- System Suspend to ACPI S3

* Other names and brands may be claimed as property of others
Part 2 - Sleepgraph
Get & Install Sleepgraph

$ cd ~/src/linux
$ cd tools/power/pm-graph
$ git clone http://github.com/intel/pm-graph
$ cd pm-graph
$ sudo make install
Run Sleepgraph

$ sleepgraph -m freeze -dev -rtcwake 5 -o outdir

Documentation


Usage


README

https://github.com/intel/pm-graph/blob/master/README

* Other names and brands may be claimed as property of others
Run Sleepgraph - Screenshot

```
lenb@lenb-Dell-XPS-13-9315:~/lpc23$ sudo sleepgraph -m freeze -dev -rtcwake 5 -o outdir
Checking this system (lenb-Dell-XPS-13-9315)...
    have root access: YES
    is sysfs mounted: YES
    is "freeze" a valid power mode: YES
    is ftrace supported: YES
    are kprobes supported: YES
    timeline data source: FTRACE (all trace events found)
    is rtcwake supported: YES
    optional commands this tool may use for info:
        turbostat: FOUND
        mcelog: FOUND
        lspci: FOUND
        lsusb: FOUND
        netfix: FOUND
os-version: Ubuntu 23.04
baseboard-manufacturer: Dell Inc.
baseboard-product-name: 021965
baseboard-serial-number: /2YZP6M3/CNCMK0022F0035/
baseboard-version: X03
bios-release-date: 04/12/2023
```
View Results

$ sudo sleepgraph -m freeze -dev -rtcwake 5 -o outdir

$ cd outdir

$ google-chrome *.html

$ chromium-browser *.html

$ firefox *.html
Other names and brands may be claimed as property of others.
Basic html timeline shows driver callbacks, trace marks

Blocks are driver callbacks during each suspend phase. Trace marks show common calls such as ksys_sync and CPUon/off. Click them to see more detailed info.

* Other names and brands may be claimed as property of others
Using -dev mode to visualize function calls with kprobes

Gray bars show function calls, hover to see function args, return values, and parent caller. Enabled with “-dev”.

* Other names and brands may be claimed as property of others
Using -proc mode to visualize user processes

Rounded bars show processes with CPU usage in red. Enabled with the “-proc” arg. “-x2” allows two back to back suspends.
Run & View Sleepgraph Stress Test Results

```bash
$ sleepgraph -dev -m freeze -rtcwake 5 -o stress.out
  -multi 3 0 -wifi
$ cd stress.out
$ google-chrome summary.html
$ chromium-browser summary.html
$ firefox summary.html
```
Other names and brands may be claimed as property of others.
### lenb-Dell-XPS-13-9315 6.5.0 freeze 3 (3 tests: 3 pass)

<table>
<thead>
<tr>
<th>#</th>
<th>Mode</th>
<th>Host Kernel</th>
<th>Test Time</th>
<th>Result</th>
<th>Issues</th>
<th>Suspend</th>
<th>Resume</th>
<th>Worst Suspend Device</th>
<th>SD Time</th>
<th>Worst Resume Device</th>
<th>RD Time</th>
<th>PkgPC10</th>
<th>SysLPI</th>
<th>Wifi</th>
<th>Detail</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>freeze</td>
<td>lenb-Dell-XPS-13-9315</td>
<td>2023/10/07 13:24:35</td>
<td>pass</td>
<td>INVALID S2WAKEx2</td>
<td>493.597 ms</td>
<td>504.443 ms</td>
<td>0000:00:02.0 {i915} (async)</td>
<td>291.606 ms</td>
<td>0000:00:02.0 {i915} (async)</td>
<td>292.731 ms</td>
<td>27.65%</td>
<td>20.66%</td>
<td>wlp0s20f3:iwlwifi:8086:51F0 reconnected 0.19</td>
<td>html</td>
</tr>
<tr>
<td>2</td>
<td>freeze</td>
<td>lenb-Dell-XPS-13-9315</td>
<td>2023/10/07 13:24:48</td>
<td>pass</td>
<td>INVALID</td>
<td>477.435 ms</td>
<td>510.939 ms</td>
<td>0000:00:02.0 {i915} (async)</td>
<td>288.625 ms</td>
<td>0000:00:02.0 {i915} (async)</td>
<td>288.074 ms</td>
<td>28.51%</td>
<td>20.25%</td>
<td>wlp0s20f3:iwlwifi:8086:51F0 reconnected 0.30</td>
<td>html</td>
</tr>
<tr>
<td>3</td>
<td>freeze</td>
<td>lenb-Dell-XPS-13-9315</td>
<td>2023/10/07 13:25:01</td>
<td>pass</td>
<td>INVALID</td>
<td>465.786 ms</td>
<td>501.072 ms</td>
<td>0000:00:02.0 {i915} (async)</td>
<td>279.016 ms</td>
<td>0000:00:02.0 {i915} (async)</td>
<td>291.770 ms</td>
<td>28.10%</td>
<td>20.93%</td>
<td>wlp0s20f3:iwlwifi:8086:51F0 reconnected 0.20</td>
<td>html</td>
</tr>
</tbody>
</table>

* Other names and brands may be claimed as property of others.
Output: `sudo ./sleepgraph -multi 10 0`

```bash
├── summary-devices.html
├── summary.html
├── summary-issues.html
├── suspend-220906-094933
│   ├── lenb-Dell-XPS-13-9360_freeze_dmesg.txt.gz
│   ├── lenb-Dell-XPS-13-9360_freeze_ftrace.txt.gz
│   └── lenb-Dell-XPS-13-9360_freeze.html
```
Trend Charts

Pm-graph repo supports a “stressreport” tool that lets us create datasets and spreadsheets for charting purposes. We have a web UI that employs html graphing tools to visualize kernel performance history.
Endurance Testing Gotchas

System PM config

  auto suspend, auto screen blank, fstrim.timer strikes at midnight

Network can fail, and stay failed

  -netfix added to repair ethernet/wifi after testing through NetworkManager
sleepgraph -wifi
Part 3 - Learnings / Discussion / Questions
Goals Evolve...

job #1: must be 100% reliable -- users must trust it

job #2: must be fast enough to be attractive to users

Today:

1-second laptop resume is considered "good enough"

touch screen needs "instant on/off" perception

Dark resume requires low tax for suspend+resume

Future: ?
How much Endurance is "Enough"?

<table>
<thead>
<tr>
<th>Iterations</th>
<th>Duration</th>
<th>Issues Found</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>20 sec</td>
<td>Gross functional issues</td>
</tr>
<tr>
<td>10</td>
<td>3 min</td>
<td>Most common functional issues</td>
</tr>
<tr>
<td>1,000</td>
<td>8 hr</td>
<td>Device transient issues</td>
</tr>
<tr>
<td>3,000</td>
<td>24 hr</td>
<td>Typical Endurance Test</td>
</tr>
<tr>
<td>10,000</td>
<td>3.5 days</td>
<td>Product Quality</td>
</tr>
</tbody>
</table>
Linux Suspend Architecture: Weakness #1

device drivers can prevent suspend -- but they generally should NOT

(e1000e -fixed)

suspend driver callbacks error return will PREVENT suspend

Generally better to not terminate a suspend in progress
Linux Suspend Architecture: Weakness #2

Run time Device Suspend can make resume FASTER

If resumed system keeps device suspended (eg. display)

but...

Run time Device Suspend can make suspend SLOWER

If run-time resume required before system suspend
Android Uconf Session on Monday:

Improving suspend/resume
Saravana Kannan <saravanak@google.com>
Endurance Lab Lessons

consistency is difficult to maintain (WIFI router, BIOS changes, device FW, distro updates etc. -- TEMPERATURE, time of day!)

Long endurance tests are beneficial, but more **more machines** are **more beneficial**

In lab: remote/automatic power reset is invaluable

USB ethernet is a PITA -- wifi more reliable

Not all WIFI is created equal -- repeated instability with some chips

wired ethernet: 1 device could prevent suspend (driver fixed)

performance regressions happen: console_suspend/printk, SATA...

Majority of issues are device specific. need a WIDE net to test all devices
Recent kernel bugs caught & fixed by stress testing

**Bug 217955** - Performance regression: `resume_console` takes 100ms longer in S2idle/S3 resume in v6.6-rc1
  - Discovered bump in resume time by charting median resume on all machines

**Bug 217804** - REGRESSION WITH BISECT: TPM patch breaks S3 on some Intel systems
  - Discovered AMD specific TPM commit that interfered with S3 on Intel systems

**Bug 216216** - [BISECT INCLUDED] PNP0501 serial driver takes almost 2 seconds to suspend/resume
  - Discovered bump in suspend time by charting median suspend on all machines

**Bug 216600** - RTC wakealarm file is missing in v6.1-rc1
  - Discovered wakealarm file missing which `sleepgraph` uses to wake from suspend.

**Bug 208733** - s2idle freeze wakes from `timekeeping_freeze` at least 100 times with `ec_no_wakeup` enabled
  - Discovered `ec_no_wakeup` kernel arg ceased to work correctly in S2idle
Hibernate to Disk - Future Revival?

1st reliable system suspend method

Stable, but disabled by default in Fedora, Ubuntu, configurable in SuSE

default (bare metal) images are not encrypted

Renewed Interest?

Migration use-case on VMM

filling crypto gap may lead to broader enablement
Future Trends

Endurance achieved, must assure it doesn't regress

Dark suspend/resume latency remains particularly important

On Intel, Low-Power Idle residency is mandatory

ACPI S3 use dwindling

Hibernate of interest again?
What should happen in the next 10 years?

- Legacy is still in production: hibernate and S3
- Must continuously regression test (function, power and performance)
- Must test on more different systems and peripherals
  - (Drivers for new devices must support suspend/resume)