

Power Optimization Linux

Cursor blink off

Blinking cursor uses an interrupt and increased power usage. There are 2 options to turn this off:

1. Framebuffer configuration

```
# echo 0 > /sys/class/graphics/fbcon/cursor_blink
```

This command can be executed on startup by adding the command to a systemd unit. It can be included in the powertop optimization service ("powersave.service").

2. Kernel parameter

Setting Kernel parameter vt.global_cursor_default=0

```
# nano /etc/default/grub
```

Add vt.global_cursor_default=0 to the following line in:
GRUB_CMDLINE_LINUX_DEFAULT="quiet vt.global_cursor_default=0"

Update grub and reboot
update-grub

Powertop

Measure

PowerTop has two methods to investigate power consumption improvement options: To find out which ones are suggested, proceed as follows:

```
# powertop  
# powertop --html=powerreport.html
```

With the normal powertop command you can see the tuning options in the "Tuning" tab. By pressing enter you can tune on or off an option. The used command line command is shown in the header of the screen. Using the html option provides a more easy way to see all the tuning commands. In the "Tuning" tab of the report all parameters and commands are shown. If you experience inaccurate measurement, then it is likely that you need to calibrate powertop first. Calibration to prevent inaccurate measurement can be done by:

```
# powertop --calibrate
```

Note: Calibration will toggle various functions like backlight or wifi. Thus, it may turn your screen black for some time, lose your connection, and so on. Do not touch the machine during the calibration.

Tune

1. Automatic

This can be done by having “powertop -auto-tune” be started as a service.

```
# nano /etc/systemd/system/powertop.service

[Unit]
Description=Powertop tunings
[Service]
ExecStart=/usr/bin/powertop --auto-tune
RemainAfterExit=true
[Install]
WantedBy=multi-user.target
```

Install and enable the service unit:

```
# systemctl daemon-reload
# systemctl enable powersave.service
# systemctl start powersave.service
```

2. Customized

These can be done by creating a script with optimization commands that is started by a service. The script will look like:

```
# cat /usr/local/bin/powersave.sh

#!/bin/sh
# Is invoked by systemd powersave.service
# Need a delay because of race condition with framebuffer init
(/sys/class/graphics/fbcon/cursor_blink)
/bin/sleep 5
echo min_power > /sys/class/scsi_host/host0/link_power_management_policy
echo min_power > /sys/class/scsi_host/host5/link_power_management_policy
echo min_power > /sys/class/scsi_host/host3/link_power_management_policy
echo min_power > /sys/class/scsi_host/host1/link_power_management_policy
echo min_power > /sys/class/scsi_host/host4/link_power_management_policy
echo min_power > /sys/class/scsi_host/host2/link_power_management_policy
echo 0 > /proc/sys/kernel/nmi_watchdog
```

```

echo 1500 > /proc/sys/vm/dirty_writeback_centisecs
echo auto > /sys/bus/i2c/devices/i2c-1/device/power/control
echo auto > /sys/bus/i2c/devices/i2c-2/device/power/control
echo auto > /sys/bus/i2c/devices/i2c-3/device/power/control
# USB Mouse
#echo auto > /sys/bus/usb/devices/1-3/power/control
echo auto > /sys/bus/pci/devices/0000:00:14.0/power/control
echo auto > /sys/bus/pci/devices/0000:00:00.0/power/control
echo auto > /sys/bus/pci/devices/0000:00:02.0/power/control
echo auto > /sys/bus/pci/devices/0000:00:17.0/power/control
echo auto > /sys/bus/pci/devices/0000:00:1f.2/power/control
echo auto > /sys/bus/pci/devices/0000:00:08.0/power/control
echo auto > /sys/bus/pci/devices/0000:00:16.0/power/control
echo auto > /sys/bus/pci/devices/0000:00:14.2/power/control
echo auto > /sys/bus/pci/devices/0000:00:1f.0/power/control
echo auto > /sys/bus/pci/devices/0000:01:00.0/power/control
/sbin/ethtool -s enp1s0 wol d
echo 0 > /sys/class/graphics/fbcon/cursor_blink

# Enable L1 and L0s on PCIe Intel Bridge and Realtek network adapter
# Does not seem to differ anything
setpci -s 00:1c.0 0x50.B=0x43
setpci -s 01:00.0 0x80.B=0x43

#SSD Optimization the deadline scheduler ensures bulk transactions won't
slow down small transactions
echo deadline > /sys/block/sda/queue/scheduler

```

Make the script executable:

```
# chmod +x /usr/local/bin/powersave.sh
```

Create the service unit that will look like:

```

# nano /etc/systemd/system/powersave.service

[Unit]
Description=Power Saving tunings
Requires=getty.target
After=getty.target
[Service]
Type=idle
RemainAfterExit=true
ExecStart=/usr/local/bin/powersave.sh
[Install]
WantedBy=multi-user.target

```

Install and enable the service unit:

```

# systemctl daemon-reload
# systemctl enable powersave.service

```

```
# systemctl start powersave.service
```

From:

<https://wiki.oscardegroot.nl/> - HomeWiki

Permanent link:

<https://wiki.oscardegroot.nl/doku.php?id=linux:powersave:powertop&rev=1565945831>

Last update: **2022/01/15 11:38**

