

Software-based energy savings

POWER DOWN!

Save power by choosing the right system and software configuration. **BY KARSTEN REICH**

joachim061974, photocase.com

A computer's power consumption is driven by a number of more or less power-hungry components: In addition to electronic components such as the CPU, RAM, and motherboard are mechanical components such as optical drives and the hard disk. The biggest power hogs are the monitor, the CPU, the mechanical components, and the graphics card. Notebooks are less power hungry in general because vendors tend to use more economical hardware to improve battery life. Whereas a portable PC requires about 20 watts for normal operation, even the most frugal desktop will want 80 watts – not counting the monitor (see the box titled “Desktop Consumption”).

Measuring Consumption

The Linux kernel and many applications offer a range of features that let users reduce power consumption, either to improve battery life or save money. Up until recently, a desktop PC's power consumption was typically ignored. This was never true of laptops because the battery is a very limited resource. It stands to reason that power-saving func-

tionality and software is almost exclusively the reserve of notebooks.

To measure your notebook's power consumption in battery-powered mode, you can either opt for the desktop's integrated tools or install add-on software. On Gnome, the power manager displays a battery pictogram in the panel and warns you if you are about to lose battery power (Figure 1). On KDE, you can build and install a tool such as the KThinkBat monitor [1], which displays the residual charge for up to two batteries, as well as your notebook's current power consumption in watts.

If you have a desktop, the only way to measure consumption – and to test how effective your attempted savings are – is to attach an ammeter (Figure 2). The formula *consumption in watts * daily use in hours * 0.365 * price of electricity*

per kWh gives you a fair approximation of what your machine is adding to your electricity bill each year.

Saving Power

The display and the CPU are typically the biggest power hogs, although it is fairly simple to minimize the consumption of both components. The Gnome and KDE power-saving applets allow users to switch off external LCD displays and CRTs after a specified idle time. This functionality relies on **DPMS**.

On a notebook, KPowerSave (Figure 3) or the PowerManager (the kde-guidance-powermanager package) and the Gnome brightness applets can do far more than just switch the display on and off, including the ability to dim LCD displays. Reducing the brightness saved around

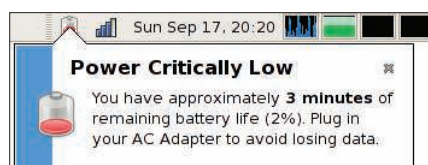


Figure 1: Gnome's power manager gives you a warning when the power is low.

GLOSSARY

DPMS: Display Power Management Signaling is a method of using the graphics card to trigger monitor power state changes with three energy-saving states: Standby, Suspend, and Power off. Both devices need to support this approach.

10 percent on our lab machine (Lenovo X61s).

At the Heart of the Machine

The processor can turn out to be a power hog. For many years, CPUs consumed the same amount of power whether idle or fully loaded. Power-saving technologies such as AMD's PowerNow and Cool'n'Quiet and Intel's Speedstep have introduced a more sophisticated approach. Under the hood, these technologies are basically a combination of several switchable clock speeds and processor voltages (P States) depending on the CPU – the PowerTOP program will tell you which states your CPU supports, but more on that later. Distributions with the current kernel support this function both on notebooks and on desktops without the need to modify the configuration.

Frequency scaling is also supported by many legacy processors on Linux. When the computer is idle, the Linux kernel will slow down the CPU. The graphical power-saving applets in Gnome and KDE call this *Ondemand* and *Dynamic*.

In addition, you can toggle your computer between various modes: *Performance* (CPU does not slow down) and *Power saving* (CPU always runs at the lowest clock speed). The latter state is not always the best approach to saving power. Again, this depends on the processor, which will consume a fair amount of power at high speed. If the CPU does not switch to a higher clock speed to handle the tasks faster, it will consume more power for a longer period of time. Dynamic frequency scaling is a better solution: The CPU completes its tasks at maximum speed and then goes back to sleep again.

Some bad guys wake up the computer on its power-saving breaks between production states. In fact, some of them wake up the computer several hundred times per second, even in idle state. This costs power, too. Up until kernel 2.6.21 (or 2.6.23 for 64-bit architecture), the kernel itself continually woke up the CPU because of its 1,000-Hz timer interrupt cycle. Many Userspace programs and kernel modules also use high-



Figure 2: A simple plug ammeter measures power consumption.

frequency timers that regularly check for keyboard input.

The PowerTOP [2] tool (Figure 4), which runs on any Linux distribution with kernel 2.6.21 or newer, will help you identify the culprits. For older distributions, you might need to build the current kernel yourself, then copy the Pow-

The 40 Watt Rackmount Server

Our new R410-EE-1U rackmount server uses an average of just 40 watts. That's less power consumption than most incandescent light bulbs.

With prices starting from just **£548** + VAT, they are easy on your budget and easy on the environment.

- Dual-Core AMD Athlon 64-bit platform
- 1U rackmount
- Low power Serial ATA disks
- Cool & quiet
- Choice of Linux OS



Digital Networks
United Kingdom

Prices as of 1st July 2008. Goto www.dnuk.com for full details.

www.dnuk.com sales@dnuk.com 0161 343 5333

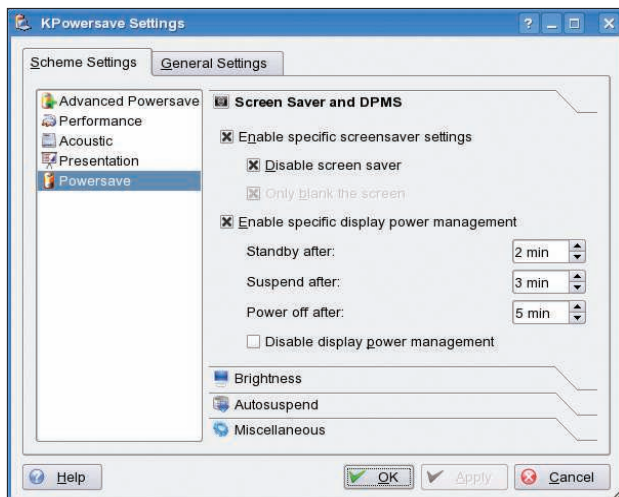


Figure 3: KPowerSave lets you configure a variety of power-saving functions.

erTOP archive onto your machine and unpack. If you have installed a full build environment, just a few commands are all it takes to build, install, and launch PowerTOP. Become root and type:

```
make
make install
powertop
```

PowerTOP shows your CPU's P States and C States. C States are only available for mobile processors. On top of this,

you see how much time the processor spends in each C State, and this is significant for energy efficiency. The aim is to keep the CPU in one of the two top C States for 90 percent of all idle time. PowerTOP not only shows you how many times per second the CPU has to wake up from its deep sleep, but also which programs and modules are responsible.

To find out which tangible piece of software is hiding behind the cryptic label, check out the "Tips & Tricks" sec-

tion for PowerTOP on the LessWatts [3] power-saving website.

To apply the suggestions that PowerTOP displays at the bottom of the screen immediately, press the key displayed by the program. This will substantially optimize your system. To make sure your computer starts saving power immediately the next time you boot it, without needing to load PowerTOP first, you need to add the commands to your */etc/rc.local* file. The system automatically executes the script whenever it changes

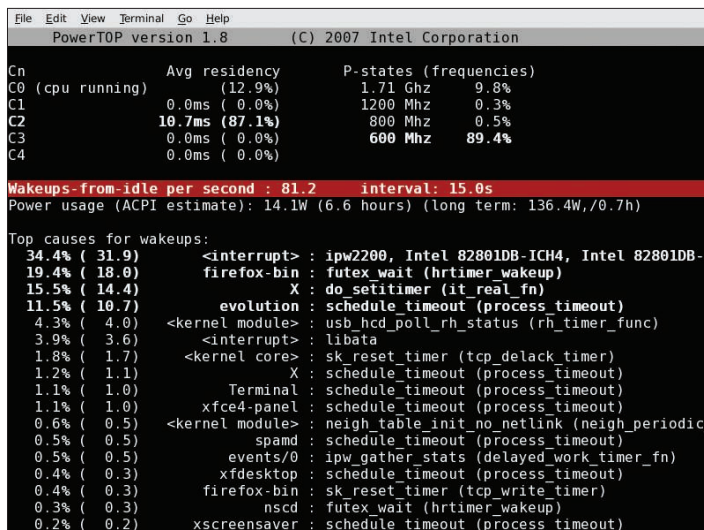


Figure 4: PowerTOP analyzes power usage.

Desktop Consumption

Most of the attention for power savings goes to little laptops and big server rooms, but you can also save power on an ordinary desktop PC. Systems that consume less than 80 watts are often difficult to optimize, but computers that consume 100 watts (not including the monitor) have some room for improvement.

The easiest way of saving power on your system is to buy a new thin-film transistor (TFT) display. Older CRT monitors consume 60 to 80 watts when in use, whereas flat-screen monitors use just 25 to 35 watts depending on their brilliance and contrast. If you prefer to keep your legacy CRT, don't forget to enable the screen saver and select a blank screen. This reduced our lab monitor's power consumption to 65 watts. Even if the computer does not transmit an image signal (in sleep mode, for example) a CRT monitor will still consume a few watts of power.

Blank screen mode does not have much effect on the power consumption of a TFT display, but suspend mode does: Modern flat screens require less than 1 watt in this

state. Assuming an average use of four hours a day, the CRT monitor will consume 4x365x80 watts per annum – that is, 116kWh. Standby mode (20 hours at 6 watts x 365 days) adds a further 44kWh. If power costs 20 cents per kilowatt hour, the CRT will add US\$ 30 a year to your electricity bill, compared with US\$ 9 for a TFT.

The CPU and graphics card are also major consumers. If you don't use 3D programs at all, or rarely use them, an on-board graphics chip is your best option. The graphics card driver is also important. A driver specially designed to support a card often reduces power consumption. If you have an additional plugin graphics card in your computer and don't actually need it, just remove the card. Most motherboards will automatically enable the integrated graphics card as soon as you remove the plugin card. Removing the plugin graphics card from a fairly ancient machine – with an integrated Intel GM865 graphics chip and a Radeon 9200 card (AGP) – and using the Intel chipset saved 5 watts.

A hard disk drive consumes between 8 and

12 watts. Many legacy machines use two or more disks because users add disks to increase storage capacity. If you replace three disks of low capacity with a state-of-the-art 500GB hard disk, you can save around 20 watts.

Avoid leaving CDs or DVDs in drives unless you need them. Some programs access these drives at boot time to collect information about the media. KDE also creates a directory for optical drives at login time. Most DVD drives consume about 20 watts of power when they power up, and this value can increase to 30 watts during normal operations.

The exercise of reducing power consumption is an interesting way to learn about your computer, and many users believe that saving energy is intrinsically important regardless of the monetary effects, but just to keep it all in perspective: A reduction of 20 watts will probably save around US\$ 5 to US\$ 20 per year depending on your usage and power service costs.

Marcel Hilzinger

the runlevel. If the file does not exist on your favorite distribution, create a file called *local* and populate it with the required power-saving commands. Then, working as *root*, make the file executable, copy it to */etc/init.d*, and link it to each runlevel in which you want to execute the script:

```
# chmod 755 local
# cp local /etc/init.d
# ln -s /etc/init.d/local /etc/rc2.d/S99local
```

The last line in this example just links the *local* script with runlevel 2, the standard runlevel for most distributions. If you frequently work in other runlevels, you will need to set up more links along the same lines.

To take some of the load off a power-hungry CPU, it is a good idea to check out process-hungry software. By disabling the 3D desktop and extensions, such as Superkaramba or the gDesklets, you can take some of the load off the graphics card.

The Strigi and Beagle search agents regularly stress the CPU.

Uncommunicative

Instead of patching the modules that regularly wake up the processor, you can simply unload them – of course, this assumes they are not needed.

If you do not need the computer to communicate in the next few minutes, just unload the modules for Bluetooth, WLAN, and Ethernet.

For Bluetooth, check first, by running the *hciconfig* command, to see whether a Bluetooth device is active.

Then stop the Bluetooth daemon and remove the module (you need to be root to do so):

```
# /etc/init.d/bluetooth stop
# modprobe -r hci_usb
```

The same thing applies to WLAN adapters. To find out which module handles the wireless connection, run *lsmod*, and run *iwconfig* to discover the wireless device's name.

The next two commands down the adapter and unload the module for an Intel 4965AGN WLAN card:

```
# ifdown wlan0
# modprobe -r iwl4965
```

If you rely on a wireless connection, why not make use of the WLAN card's power-saving feature. For cards that use the *ipw2100* or *ipw2200* driver, you can enable the feature with the following command:

```
# iwpriv wlan0 set_power 7
```

To discover whether your card supports the *set_power* option, you can type *iwpriv* without parameters. More recent WLAN adapters that use the *iwl3945* or

GLOSSARY

C States: Sleep or idle states for ACPI-capable processors. Five levels range from C0 to C4. The higher the number, the less power the CPU uses.

1

2

3

4

5

6

Easy MPI Debugging

PGDBG is an easy-to-use graphical parallel MPI, OpenMP and hybrid MPI+OpenMP debugger for Linux and Windows clusters.

PGI CDK compilers and tools are available directly from most cluster suppliers. Take a free test drive today at

www.pgroup.com/reasons

PGI CDK® Cluster Development Kit®

iwl4965 drivers can be switched to power-saving mode as follows:

```
# echo 5 > /sys/bus/pci/ 2
drivers/iwl4965/*/power_level
```

This setting does not affect the connection. The value that follows *echo* stands for the power-saving mode: 1 saves the least and 5 the most – 6 disables the function.

On our lab system (a Lenovo X61s), I saved around 1.4 watts (which is equivalent to 10 percent) by switching to the power-saving mode.

It might be worth your while to disable USB 1.1 on a notebook. USB mice and other USB 1.1 devices will not work after doing so, but integrated USB 2.0 devices such as touchpads or pointers will continue to provide normal service. To do this, unload the *uhci_hcd* module. On our lab machine, this reduced overall consumption by a further 10 percent, and it also helped to keep the heat down.

Deep Sleep

Unloading drivers switches off a computer's components individually. And if you will not be using the computer for an extended period of time, you can switch it to suspend mode. Suspend-to-RAM and Suspend-to-Disk typically work fine on more recent hardware. The advantage of saving the session data in RAM is that the computer boots extremely quickly when you need it. On the other hand, Suspend-to-Disk does not consume any power, and it prevents data loss in case of a power outage.

Hard Disk

The hard disk is another major power consumer. Depending on the amount of work it has to do, it would need a fair

amount of power if the drive motor runs continuously. Again, this is a component that you can stop: type *hdparm -S N /dev/sda*. The value *N* multiplied by five is the inactivity span in seconds, after which the hard disk automatically switches off. A value of 0 disables this feature.

This does not mean that the hard disk automatically stops spinning: Just as in CPU sleep mode, some standard system settings keep on waking up the disk. One reason is the speed at which the kernel writes the disk cache out to the disk. To discover the value for your machine in hundredths of a second, you can check the */proc/sys/vm/dirty_writeback_centisecs* file.

Set kernel access to 10 minutes with the following command:

```
# echo 60000 > 2
/proc/sys/vm/dirty_writeback_centisecs
```

Assuming your cache is big enough, the system will not even need to trouble the disk to write a file. But remember that buffered data will be lost in case of a power failure.

The syslog daemon can sabotage hard disk sleep mode. It regularly writes logs directly to disk without using the buffer, but you can add a *-* sign at the start of each entry in your */etc/syslog.conf* file to tell the daemon to use the disk cache in the future. Again, remember the danger of data loss in case of power failure.

Mounting Ext3 filesystems with the *noatime* parameter – or for files that require the *atime* option, with the *nodiratime* or *relatime* parameters – avoids further power-hungry disk accesses [4]. By default, Linux uses the *atime* parameter when mounting Ext3 filesystems. This tells Linux to log the last file access times and record timestamps at short intervals, which in turn means regular disk access and power consumption.

The developers of Laptop Mode tools [5] advise against using this power-saving feature on desktop PCs: Hard disks for desktops differ from laptop models in that they are designed for non-stop use, but not for being powered on and off at short intervals [6].

The Laptop Mode tools are a big help if you need to enable the kernel's notebook support features and are easily cus-

tomized by means of a file (see the "Laptop Mode Tools" box). The tool, which you will find in the repositories of many distributions, helps you save power by focusing mainly on the hard disk and its power-saving modes. If necessary, it will change system settings that cause the disk to start unnecessarily.

The tools support a number of additional power-saving options: they automatically enable the power-saving configuration when you switch to battery power and enable the Centrino WLAN adapter power-saving feature, support hibernation in case of a low battery state, start or stop daemons depending on the power supply, and modify the display brightness.

Conclusions

Tweaking the right controls lets users fight power hogs on laptops and desktops. Some tips are fairly easy to implement, but for others, you will need to delve more deeply into your system settings. In all cases, the reward for your work will be vastly improved battery life on notebooks and power savings in the long run. ■

INFO

- [1] KThinkBat: <http://lepetitfou.dyndns.org/projects/kthinkbat>
- [2] PowerTOP: <http://linuxpowertop.org>
- [3] PowerTOP tips and tricks: <http://www.lesswatts.org/projects/powertop/known.php>
- [4] Mount options: <http://www.lesswatts.org/tips/disks.php>
- [5] Laptop Mode tools: http://samwel.tk/laptop_mode/
- [6] Desktop PC hard disks: http://samwel.tk/laptop_mode/faq

Laptop Mode Tools

Configuration file: */etc/laptop-mode/laptop-mode.conf*

Activity monitor: */proc/sys/vm/laptop-mode* (1 = aktiv)

The Laptop Mode Tools are disabled by default on Ubuntu. To enable, change the value *false* in the *ENABLE_LAPTOP_MODE=false* line of the */etc/default/acpi-support* file to *true*. The Powersave daemon provides functionality similar to the Laptop Mode tools on openSUSE.

THE AUTHOR

Karsten Reich studied history and computer science at the Technical University of Dresden, Germany. He's been interested in computers since he was a child, and got started with Linux on Debian 3.0 (Woody).

He covers software, with the occasional hardware test as a Contributing Editor for our sister publications EasyLinux and LinuxUser.

