Make Linux faster and lighter

Posted at 12:43pm on Friday October 30th 2009

With just a few tweaks, your Linux box can be lighter, sprightlier and quicker than ever before. Read on for the best ways to speed up your boot sequence, optimise KDE and Gnome, and get better performance from your favourite apps. We've also got some top tips from our favourite free software gurus...

Gone are the days when you could make a cup of tea and drink it in the time it takes your computer's operating system to boot (with one notable exception). On that basis, you might think that your Linux machines are already performing at the fastest possible speed, right?

Sadly, this is not always the case. Communities developing mainstream Linux distributions have to appeal to the widest possible audience and ensure compatibility with the widest range of hardware. This means that someone running a mainstream distro on a netbook or a low-end PC may well be using many of the same settings as someone with a high-end gaming machine. But it doesn't have to be this way!

With just a few tweaks and some experimentation, your Linux system can realise your machine's untapped potential. Do you have a dual-core processor? Take advantage of this by running boot processes in parallel. Do you have more memory than you know what to do with? You could try caching data in memory rather than swapping space for faster access.

The great strength of everyone's favourite free OS is that it can be customised from the ground up, so Linux is the ideal tool to tailor to your needs. But once you have an ultrafast system, how can you become more productive? We've consulted prominent members of the free software community for their favourite tips that make their systems more productive - and they could do the same for you.

Make Linux boot faster

Any productive machine needs to be up and running as soon as possible, and a sluggish boot can hinder your efforts – which is why boot times were the first thing we thought about improving. One word of warning before you begin: we recommend that you make a backup before you make these alterations, as a bug in your bootloader can render your Linux box unbootable!

Remove the timeout

You may notice that each time you boot there's a small count-down from three to zero, which is great if you regularly select a different OS or Linux kernel at boot time, but useless if you always boot into the same distro. Fortunately, it's easy to remove by opening **/boot/grub/menu.lst** in a text editor with root permissions and finding the line showing:

```
timeout=3
```

Once you've found it, change the value to zero. Save and exit then reboot and you should notice you have just knocked three seconds off your boot time.

Improve disk performance

If you have a DMA-compatible (Direct Memory Access) hard drive, you can increase data throughput threefold with a simple tweak. This will improve boot times because read times will be reduced, and overall performance will increase whenever the hard disk is accessed.

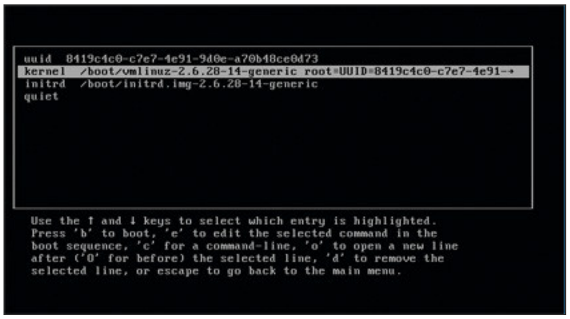
Start by installing **hdparm** through your package manager, then fire up a root terminal and type: **hdparm -d1 /dev/hda1** replacing **/dev/hda1** with the location of your boot partition to increase startup times or the root partition to increase general performance. Gnome users can have this run automatically each startup by heading to **System > Administration > Services**. You can then add this line as an entry with **gksudo** at the start to ensure it runs with root permissions without requiring additional authentication.

Run boot processes in parallel

Parallelism can lead to big performance boosts, because running two processes at once will take half the time of running them sequentially (at least in theory). You can take advantage of this technique in Grub by firing up **/etc/init.d/rc** in a text editor with root permissions and finding the following line:

```
CONCURRENCY=none
```

You would then replace **none** with **shell** before saving and closing your text editor. When you reboot you should see a noticeable decrease in your boot times (around one or two seconds in most cases). If you don't see an increase, this is because this tweak is aimed primarily at systems with multi-core processors. If you have a solo-core processor you could actually increase your boot time if you use this tweak, which was the case with our test system where we saw a 2.4-second increase.



```
uuid   8419c4c0-c7e7-4e91-9d0e-a70b48ce0d73
kernel   /boot/vmlinuz-2.6.28-14-generic root=UUID=8419c4c0-c7e7-4e91-→
initrd   /boot/initrd.img-2.6.28-14-generic
quiet

  Use the ↑ and ↓ keys to select which entry is highlighted.
  Press 'b' to boot, 'e' to edit the selected command in the
  boot sequence, 'c' for a command-line, 'o' to open a new line
  after ('O' for before) the selected line, 'd' to remove the
  selected line, or escape to go back to the main menu.
```

You could edit a text file and restart your machine to profile your system, or just click a few buttons in Grub.

Optimise memory

One great way to improve performance is to define how swap space is used. A swap partition is where the Linux kernel caches data in virtual memory to quickly swap into RAM as and when it's needed. A dedicated partition isn't compulsory, but the space is also used to store your machine state if you choose to hibernate. The effects of this tip depend on your system and whether you have a swap partition, but if you have plenty of RAM you'll find that reducing swappiness will give you a noticeable performance boost.

This will allow the kernel to cache data in memory for faster access and reduces the amount of data being swapped in and out of swap space at any given time. Simply open **/etc/sysctl.conf** in a text editor with root permissions, then append the following line to the bottom of the file:

```
vm.swappiness=10
```

(You will need to restart the session in order for your changes to take effect.) You can tweak this value to see how performance improves. The lower the value, the less you use swap space and the more data is cached to memory. This was the value that seemed to work best with our test system using 512MB RAM. However, this tip isn't just restricted to systems with large amounts of memory - systems with 256MB of RAM or less may see a performance boost if swappiness is increased, as this will cache more data to swap space and free up more memory for day-to-day applications.

The precise values vary from system to system so this will require some trial and error, especially as reading data from swap space still takes longer than reading it from memory, but the end results are usually worth the effort.

Speed up ext3

You can gain some significant performance benefits by enabling write-back operation in ext3. This tweak isn't restricted to systems running KDE, but it is disabled by default in almost every distro. This is primarily because older hard drives don't support this feature, though newer hard drives can achieve a minor speed boost. This won't affect your day-to-day disk operations, so it is not recommended you try this tweak on a typical home system, but you will see an improvement for the high intensity disk operations that are typically the preserve of servers.

After making a back up of the file, open **/etc/fstab** in a text editor with root permissions and look for a line with the following section of text:

```
relatime,errors=remount-ro
```

and replace it with this:

```
noatime,nodiratime,errors=remount-ro,data=writeback
```

If there is no text either side of this snippet, you have done it wrong! Save and exit this file then open **/boot/grub/menu.lst** and find the following two lines:

```
# defoptions quiet splash
#altoptions=(recoverymode) single
```

They won't be next to each other, but once you find them append the following at the end of both those lines:

```
rootflags=data=writeback
```

then save and exit. Open up a root terminal and run **update grub**. You then have the option of simply restarting to apply these changes to your file system, or you can apply them on the fly by installing tune2fs from your package manager. If you choose the latter option you would then type the following into a root terminal:

```
tune2fs -o journal_data_writeback /dev/hda1
```

substituting **/dev/hda1** with your root partition. Be warned: write-back mode puts you at a little risk of losing data if you machine suddenly powers off, ie if you kick the power cord out, if there's a major system crash or if you get a general power outage.

Voice of the Guru #1

Richard Stallman - founder of the GNU Project

"I recently learned about the M-. command in Bash (use Escape followed by a full stop if you don’t have a Meta key), which copies the last argument from the previous command."

Optimise Gnome

Along with KDE, Gnome is one of the two most widely used desktop environments in the Linux ecosystem, but it's quickly being overtaken by Xfce and other more lightweight alternatives when it comes to performance and speed. However, with just a few of our tweaks, Gnome can keep up with the rest of the pack...

Install preload

Preload is a daemon that analyses what you do on a day-to-day basis and fetches the binaries and files you're most likely to need to boost startup times and general performance. In Ubuntu you just need to search for 'preload' in the package manager, but on other distros it's worth checking that the service is running. To activate the service, type:

```
service preload on
```

After enabling automatic login and installing preload we initiated two restarts on our test machine to give the daemon a chance to monitor the startups. After comparing the two times, we found that preload had trimmed a second off the time it took to get a usable desktop. This doesn't sound a lot, but if it's used in conjunction with your new fast boot time then it's well worth the effort.

Enable automatic login

Yes, we know we're always telling you how important security is, but if you live on your own in a castle that doesn't have internet access then you're probably safe to enable automatic logins. The time saved is the time it would otherwise take you to type your password.

The GUI method is by far the easiest. Simply head to **System > Administration > Login Window**. After authenticating yourself, head to the Security tab and check the box next to Automatic Login. You can then choose from the list of users in the drop-down menu who gets logged in automatically.

Unfortunately for users of Fedora, this option isn't available, so you need to make the necessary changes manually. To enable automatic login you need to open **/etc/gdm/custom.conf** in a text editor with root permissions and then append the following to the bottom of the file:

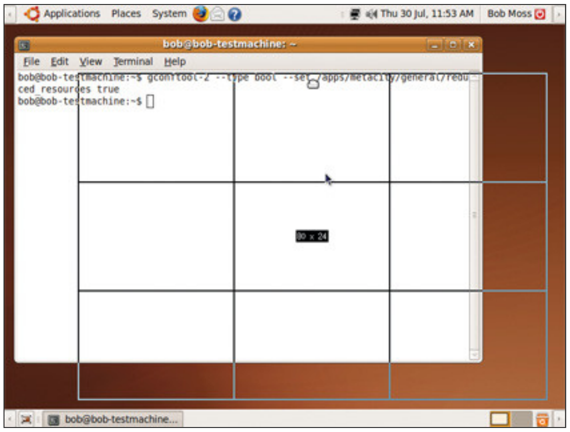
[daemon] TimedLoginEnable=True TimedLogin=UserName TimedLoginDelay=0
replacing **UserName** with the user you would like to log in.

Use wire frames

A common problem with low-powered systems is the lag generated when you click and drag a window. Gnome renders the window as it moves, which can slow performance down, but it has a built-in method to reduce the demand on system resources this causes, which you can activate by typing the following into a terminal:

```
gconftool-2 --type bool --set /apps/metacity/general/ reduced_resources true
```

Now when you click and drag windows you will see wire frames instead of the window contents, which drastically improves performance on under-powered systems (but has little or no effect on others). Note that this trick won't work if you are using Compiz or an alternative window decorator.



```
bob@bob-testmachine:~$ gconftool-2 --type bool --set /apps/metacity/general/redu
ced_resources true
bob@bob-testmachine:~$ 
```

Gnome's wireframe mode eliminates the overhead of showing window contents whilst resizing or moving them.

Make menus faster

If your menus are starting to feel sluggish, the most likely cause is the icon delay. Every time you open the Gnome menu for the first time the icons have to be loaded from their source files. On slower systems this can look a little messy, so there is a timeout deliberately set to ensure the icons are loaded in time for you to see them. However, with modern hardware this is not generally necessary, so the option can safely be adjusted. Create a new file in your home directory named **.gtkrc-2.0** then open this file in your favourite text editor to type:

```
gtk-menu-popup-delay=0
```

After saving the file, press **Ctrl+Alt+Backspace** to reinitialise xserver and your menus should open considerably faster.

Optimise KDE

KDE has received a mixed press lately. Though there's no doubting that its good looks and the new plasmoids set it apart from the competition, KDE 4 still can't match KDE 3 (or even Gnome) for speed and performance. Though this area is in heavy development, a few tweaks can trim valuable seconds off your startup time and make the desktop as a whole much more responsive.

Disable IPv6

Until IPv6 is more implemented, Konqueror has to translate between IPv4 and IPv6 - so if you can get a speed boost by turning off IPv6. The methods are slightly different between distros. For instance, in Kubuntu you need to edit **/etc/environment** as root and add the following line to the file:

```
KDE_NO_IPV6=True
```

In openSUSE, edit **/etc/sysconfig/windowmanager** with root permissions and find this line:

```
KDE_USE_IPV6="yes"
```

then change the answer to **no**. In both cases you will need to save the file and exit before restarting the session for the changes to take effect.

Enable automatic login

Gnome isn't the only desktop that can benefit from you setting up an automatic login: KDE 4 users can do much the same thing by heading to System Settings and clicking on the Advanced tab. Here you can run the login manager and enable automatic login from the Convenience tab. You'll also need to choose the user you want to log in automatically from the drop-down menu below the checkbox. When you reboot you should miss the login screen altogether and launch straight into the KDE 4 splash screen.

Start with an empty session

By default KDE 4 saves volumes of data about your current session, such as which windows are open and which processes were running so that next time you log in your session is exactly as you left it. This is very convenient, but it also slows your boot time down as this data has to be reloaded on each boot. We can avoid this problem by ensuring that we start with a fresh session. Select System Settings (or Configure Desktop in OpenSUSE) from the KDE menu and in the Advanced tab start the session manager. Towards the bottom of the form select the radio button next to Start With An Empty Session then click Apply.

Start Konqueror faster

We can make Konqueror run much faster by enabling preloading. This uses some memory, but means that the next time you fire up Konqueror it will open in around half the time and in the same place you left it. If you used the previous tweak to start KDE with an empty session then there's no need to panic, as we will configure preloading to run automatically when KDE starts.

To enable Konqueror preloading, open up the browser and head to **Settings > Configure Konqueror**. In the Performance section check the box next to **Preload An Instance After KDE Startup**. You can also set how many instances of Konqueror are preloaded at startup depending on how many Konqueror windows you tend to run at any one given time.

Voice of the Guru #2

Gael Duval - Founder of Mandriva and Ulteo

"I'm always using '!part_of_command' as a repeater of a shell command you want to repeat. For instance if you have typed 'ps auxww' in the near past and want to go with it again, then just type '!ps'. Of course it's also cool to use it jointly with a command number returned from the list of past commands provided by 'history'. Just type '!command_number' to repeat a command from the history list."

Boost your network

You've tweaked your router, optimised your server and replaced all your broadband filters in your house - and you still have a slow connection. Before you make an angry phone call to your ISP, there's a possibility that the client machine may be the bottleneck. With just a few of these tweaks you can improve your connections, or at the very least eliminate your client machine as the single point of failure.

Use hostname 'localhost'

This doesn't cause a drastic improvement on its own, but in conjunction with the other network tricks here can improve your local machine's network performance. Simply open **/etc/hosts** in a text editor with root permissions and change the top two lines to:

```
127.0.0.1 localhost yourhost
127.0.1.1 yourhost
```

where yourhost is the name of the machine (ie bobdesktop). Ensure you keep a backup of your **/etc/hosts** file, as not all distros are compatible with this!

Optimise TCP settings

Distros come preconfigured for "average" internet users, but in a world of 3G, dial-up and ADSL there is no such thing as an average user. Now edit **/etc/sysctl.conf** as root and append the following:

```
net.ipv4.tcp_timestamps = 0
net.ipv4.tcp_sack = 1
```

(This may take some trial and error, so keep a backup of the files you edit.) The top line removes timestamps and so relieves all incoming and outgoing packets of a 12k overhead. The bottom line enables selective acknowledgements, which means fewer checks are initiated on each packet so they are delivered quicker. We don't recommend the latter tweak for dial-up connections as this increases the number of packets which need to be resent.

Metrics and backlogs

You can further optimise your TCP settings by appending the following lines to the same text file:

```
net.ipv4.tcp_no_metrics_save = 1
net.core.netdev_max_backlog = 2500
```

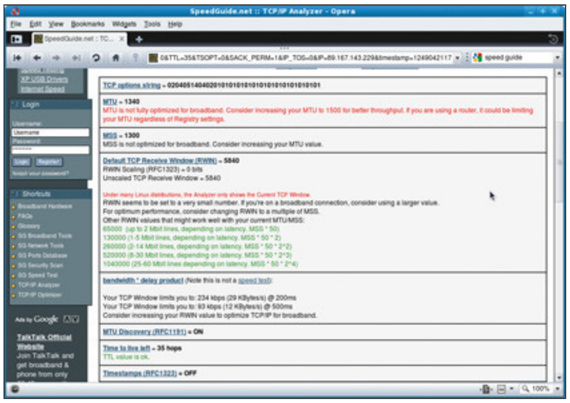
The top line speeds up connections by ensuring that TCP metrics are not saved for each individual packet. The second line dictates the backlog of packets allowed. To give an idea of the scale of the figure you should use for your backlog, around 2,000 is recommended for wireless or older Ethernet connections, and around 5,000 for a 1GB Ethernet cable to broadband connection. On the other hand, this figure can rise up to 30,000 for an expensive 10GB Ethernet cable. The slower the connection, the lower your backlog should be, as the aim of this tweak is to clear outstanding packets as soon as possible in order to boost connection speeds.

TCP window scaling

The TCP window settings define the minimum and maximum size of packets we can send and recieve. Dial-up users will see a speed boost with smaller packet sizes while broadband users will see a speed boost with larger packet sizes. Edit **/etc/sysctl.conf** as root, and append this:

```
net.ipv4.tcp_window_scaling = 1
net.ipv4.tcp_wmem = 10240 87380 16777216
net.ipv4.tcp_rmem = 10240 87380 16777216
net.ipv4.tcp_mem = 16777216 16777216 16777216
net.core.rmem_max = 16777216
net.core wmem_max = 16777216
```

Those figures are optimised for a 2Mbps connection on Sky broadband over a wireless connection, but the packet size recommendation for any given connection is usually maximum bandwidth divided by latency. You can find these figures by heading to www.speedtest.net, and you can also analyse how your changes are affecting your connection via www.speedguide.net:8080.



TCP options string = 020405140402010101010101010101010101

MTU = 1340
MTU is not fully optimized for broadband. Consider increasing your MTU to 1500 for better throughput. If you are using a router, it could be limiting your MTU regardless of Registry settings.

MSS = 1300
MSS is not optimized for broadband. Consider increasing your MTU value.

Default TCP Receive Window (RWIN) = 5840
RWIN Scaling (RFC1323) = 0 bits
Unscaled TCP Receive Window = 5840

Under many Linux distributions, the Analyzer only shows the Current TCP Window.
RWIN seems to be set to a very small number. If you're on a broadband connection, consider using a larger value.
For optimum performance, consider changing RWIN to a multiple of MSS.
Other RWIN values that might work well with your current MTU/MSS:
65000 (up to 2 Mbit lines, depending on latency. MSS * 50)
130000 (1-5 Mbit lines, depending on latency. MSS * 50 * 2)
260000 (2-14 Mbit lines, depending on latency. MSS * 50 * 2^2)
520000 (8-30 Mbit lines, depending on latency. MSS * 50 * 2^3)
1040000 (25-60 Mbit lines depending on latency. MSS * 50 * 2^4)

bandwidth * delay product (Note this is not a speed test):

Your TCP Window limits you to: 234 kbps (29 KBytes/s) @ 200ms
Your TCP Window limits you to: 93 kbps (12 KBytes/s) @ 500ms
Consider increasing your RWIN value to optimize TCP/IP for broadband.

MTU Discovery (RFC1191) = ON

Time to live left = 35 hops
TTL value is ok.

Timestamps (RFC1323) = OFF

www.speedguide.net can give you an indication of which values and tweaks will improve your network connection.

Faster Firefox

Continuing with the networking theme, you can optimise much more than just your TCP settings. Mozilla Firefox has a raft of options cunningly hidden from users that can untap the browser's potential...

Disable IPv6 (again!)

With this tweak we will simply amend an existing value. Type the following into the search bar: **network.dns.disableIPv6**

You shouldn't have to type out the whole thing, as the listings change dynamically as you're typing. Simply double-click on the entry to change the default value of "False" to "True".

Render pages faster

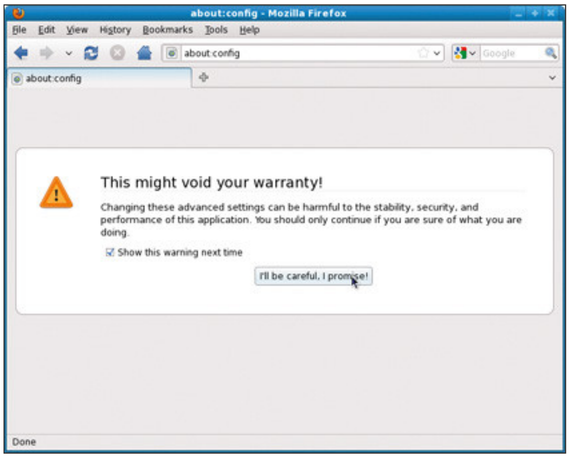
Create a new integer value in **about:config** named **content.notify.backoffcount** and set the value to **5** so that Firefox won't wait for the entire page to download before rendering. You can also create a value with the name **nglayout.initialpaint.delay** and set the value to **zero**. This ensures that Firefox doesn't wait for the page layout information to be fully downloaded before rendering.

Optimise your history

If you reduce the amount of web history that Firefox stores, it will load faster and also save you some disk space. Type the following in the search bar: **browser.history** and change **browser.history_expires_days** and **browser.history_expires_min** to **zero**. You can also use low numbers if you want to keep some browser history.

More TCP tweaks

Change the **network.http.pipelining** and **network.http.proxy.pipelining** values to "True" and then set **network.http.pipelining.maxrequests** to **8** instead of **4**. These tweaks boost performance for broadband users as we use a TCP technique known as pipelining, which allows Firefox to make multiple requests on a single connection.


about:config - Mozilla Firefox
File Edit View History Bookmarks Tools Help
about:config
Google
about:config
I'll be careful, I promise!
Done

This might void your warranty!

Changing these advanced settings can be harmful to the stability, security, and performance of this application. You should only continue if you are sure of what you are doing.

☑ Show this warning next time

Any mistakes here could break Firefox, so once you've made a backup and a vow of honour you can unleash the power hidden in your browser.

Speed up menus

Start by typing **about:config** and accepting the warning that comes up (you will need to do this for all these Firefox tips). The listing you see is much like the Firefox equivalent of the Windows registry, which means that though this is a very powerful way to tweak your browser you can cause serious stability problems if you get it wrong.

We can make our Firefox menus load faster by adding our own value to the listing. Right-click anywhere in the listing and then click **New > Integer**. Use **ui.submenuDelay** as the name and then set the value to **zero**. When you restart Firefox you should notice that your menus load noticeably faster, as you have removed the delay for the menu popups.

Voice of the Guru #3

Paul Frields - Fedora project leader

"Here's something I can’t live without: combining the Mutt mail reader with offlineimap. The offlineimap utility synchronises my remote IMAP stores on to my local hard disk, which means my email reading is faster by several orders of magnitude. I can sync my email before I pack for travel, and then while in transit, or waiting at an appointment, I can read email as quick as a flash because it's all stored locally. Mutt makes my mail processing chores incredibly efficient, so I can effectively deal with the thousands of email messages I get every week. When I get back online and run offlineimap again, all the flags are synchronised again and my remote IMAP stores record what I've read."

Speed up your software

Almost all mature applications have options hidden away in the GUI that can boost their performance on your system, so if we've given you the tweaking bug there's plenty of scope for further experimentation. Whether you're shaving a few seconds of boot time off OpenOffice.org or improving the backbone of your system, the applications that make up your Linux system are what it's all about.

The fastest mirror

Open a root terminal and type: **yum install yum-fastestmirror**. This well-known Yum plugin ensures that you always download packages from the fastest mirror, which could save minutes on large distro downloads.

Avoid rechecking

Every time you update or install a package Yum has a tendancy to re-download all the metadata packages from each server, which isn't ideal for users with slow connections. You can resolve this by downloading the entire cache in one go and allowing Yum to then refer to this rather than the online sources. Fire up a root terminal and type:

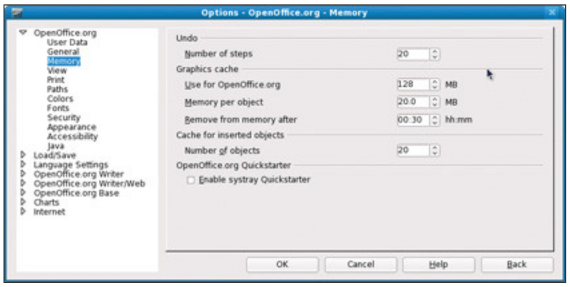
```
yum checkcache
```

We recommend you run this command at least once a week to keep the list up to date.

Faster OpenOffice.org

We often harp on about how slow OpenOffice.org is, but it doesn't have to be like that. With just one tweak you can remedy the problem, and with further alterations the latest OpenOffice.org can run on much older hardware. So, before you ditch this office suite forever and use AbiWord and Gnumeric instead, try monkeying with the settings under **Tools > Options > Memory** to reduce the amount of RAM that OOo chews up.

One easy tweak here that we recommend is to reduce the number of undo steps from the default value of 100 to somewhere between 20 and 30. That way you use less memory than before but still gain the benefits of having an undo function. Another thing you can do is to enable the Systray quick starter (in the bottom of the window). This enables you to keep an instance of OOo in the cache, so you can restart more quickly.



Undo

Number of steps: 20

Graphics cache

Use for OpenOffice.org: 128 MB

Memory per object: 20.0 MB

Remove from memory after: 00:30 hh:mm

Cache for inserted objects

Number of objects: 20

OpenOffice.org Quickstarter

□ Enable systray Quickstarter

Make OOo more efficient in Tools > Options > Memory.

Increase graphics cache

On systems with under 512MB RAM, use a graphics cache of between 32MB and 64MB, but for systems with 512MB or over we recommend a cache size of at least 128MB, as the additional memory will help large apps such as OOo run that little bit faster.

Remove items from memory

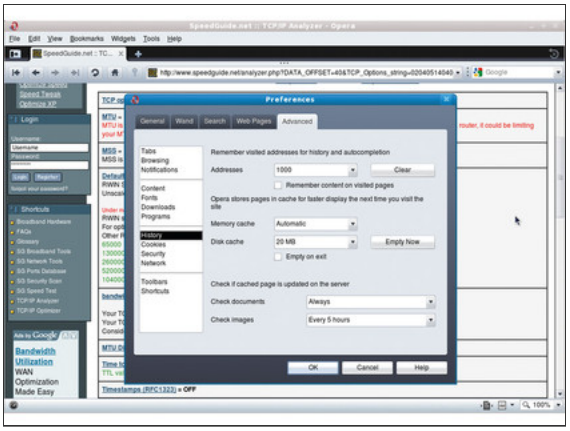
Also in the OpenOffice.org Memory dialog we can define how long OOo leaves things before refreshing its cache. On a system with 256MB RAM or less we recommend setting the refresh to 0:30 (every 30 minutes) as more frequent reloads will cause OOo to lock up or become sluggish. For users of higher-end systems, the 10-minute default is usually fine.

Clean out Yum

For peace of mind and more disk space it makes sense to ensure that Yum's cache doesn't contain any redundant packages. To do so, open a root terminal and type: **yum clean all**. Ensure you run this before you do any other optimisations for Yum, as this operation will reset them.

Try Opera

It may only be free as in beer and not as in speech, but Opera is a lightweight and very capable alternative to Firefox that's fast enough already even without any optimisation tweaks. It integrates additional tools to handle mail, file sharing and even IRC chats. The Opera community has developed thousands of skins and plugins, and with recent releases personal web space and even a personal web server have been included free of charge.



Remember visited addresses for history and autocompletion

Addresses: 1000

□ Remember content on visited pages

Opera stores pages in cache for faster display the next time you visit the site

Memory cache: Automatic

Disk cache: 20 MB

□ Empty on exit

Check if cached page is updated on the server

Check documents: Always

Check images: Every 5 hours

Opera - not free as in freedom, but still a great, memory-friendly web browser.

Install apt-fast

The apt-fast script can provide a huge speed boost to package management, as we use Axel as a drop-in replacement download manager for Aptitude rather than the traditional wget. You will need to install Axel through Aptitude and then open a text-editor with root permissions and paste the shell code from here directly into the file. Then we recommend you save it as **/usr/local/bin/apt-fast** (no extension) and close the text editor before opening a root terminal and typing:

```
chmod +x /usr/local/bin/apt-fast
```

To use the script, try this as root: **apt-fast install packagename** or simply use apt-fast update and apt-fast upgrade to update all your packages. The script may be slow in some cases, as the tweaks you made to Aptitude will not apply to the script, and Axel does not use Aptitude's cache.

Get a better download mirror

Head to **System > Administration > Software Sources** in Gnome and select **Other** from the drop-down menu next to Server Location. Then click **Select Best Server** and wait for the checks to finish. When this is finished, close the window and let your distro update its sources.

A common problem with this method is that the update window can freeze and go grey. (This is a known issue with older versions of Ubuntu.) This can be resolved by closing the window (using the Force Quit Gnome applet if necessary by right-clicking the panel and selecting Add to Panel) and then running Add/Remove Software to update the sources.

Halt Java

The JRE in OpenOffice.org is the main culprit for the slow startups. You don't need the JRE to run in a day to day environment, so you can safely disable it, and in the event you do need the JRE you can quickly re-enable it. Simply head to **Tools > Options** and select Java in the left-hand box. To disable the JRE simply uncheck the box next to Use A Java Runtime Environment.

Remove old packages

Run this to remove orphaned dependancies with APT:

```
apt-get autoremove
```

and to remove all package files from the cache:

```
apt-get clean
```

Use autoclean rather than clean to remove old versions of packages. Make sure you're root!

Reduce history

By reducing the amount of history that Opera keeps we can reduce the time it takes to initialise it and also curtail Opera reloading its history every five hours. This is achieved by heading to **Tools > Preferences** and clicking on the **Advanced** tab. Select **History** from the left-hand side and then click **Clear** in the Addresses section. Then set the value as anything from 0 to 1,000, which will be considerably less than the default. You can also uncheck **Remember Content On Visited Pages** to save further disk space.

Hey Presto

Sometimes updates vary very little from their previous download, especially if it's just a bug fix. If you want to download only the changes, run this as root to install the Presto plugin for Yum:

```
yum install yum-presto
```

Disable name completion

Users with low-end systems may find that pictures and the auto-completing navigation bar in Opera can be a little sluggish at times. We can remedy the first issue by disabling Turbo, which pre-draws images before they are loaded. Look in **about:config** for the **Turbo Mode** option in **User Prefs**, then uncheck it. To disable name completion, choose **Tools > Preferences > Advanced > Network > Server name completion**, then uncheck **Look For Local Network Machine**.

Voice of the Guru #4

Clement Lefebvre - Founder and maintainer of Linux Mint

"Under Linux Mint we added a 'search' command, which is based on 'find' and 'grep'. It's recursive, it highlights the results, it shows file names and even line numbers. For instance, if you can't remember where you saved that letter you wrote to John, type 'search for John in ~' and that will list all files containing 'John' anywhere under your home directory."

You should follow us on Identi.ca or Twitter

Stumble it!

Your comments

Small Correction

Martin (not verified) - October 30, 2009 @ 2:58pm

Hi!

You write: "You may notice that each time you boot there's a small count-down from three to zero. This was introduced originally to ensure that older hardware loaded modules in time for the kernel to boot."

This is not correct. The timeout is part of GRUB (the bootloader), and it's there so the user has time to press a key to enter the GRUB menu and select one of the entries. It occurs even before ANY operating system (Windows, Linux, BSD, younameit) is loaded, so it certainly has NOTHING to do with loading Linux kernel modules.

Cheers, Martin

KDE 4 slow?

Anonymous Penguin (not verified) - October 30, 2009 @ 5:55pm

I've been using KDE 4 since 4.1, and I've always found it to be faster and more responsive than 3.x was. This misconception that it's slower somehow probably comes with the baggage attached to anything Ubuntu - Kubuntu is now and always will be bloated compared to other distros using KDE 4. Give Slackware a try, and tell me you still think it's slow.

Hint #2 is useless.

BubbaT (not verified) - October 30, 2009 @ 6:47pm

Say you have recently edited five files.
How do you know !emacs will edit the right file.
Worse, since you don't know what command will be called,
you can call a destructive command by accident. For example !mkfs could format the wrng partition losing critical data.

There is a much better way to the same thing.
Cntl-R starts a reverse search that will take you back through your history using emacs-like commands. Not only that but once you find your command, you can edit it.

Thank you

Johannes (not verified) - October 30, 2009 @ 7:26pm

Really, that's an impressive post. Thank you.

For "Use wire frames": This

ZaWertun (not verified) - October 31, 2009 @ 2:56pm

For "Use wire frames":

This ugly (for my opinion) grid can be disabled with
enabling parameter "desktop/gnome/interface/accessibility":

```
gconftool-2 --type bool --set /desktop/gnome/interface/accessibility true
```

Grub 2 please

Qjet (not verified) - November 1, 2009 @ 5:08am

Grub 2 is seriously alien to me, these tweaks I understand for GRUB without needing to say where I need to edit. GRUB 2 I've got no clue. I hate to ask but could you include GRUB 2 instructions as well?

Grub 2

Anonymous t-shirt collector (not verified) - November 1, 2009 @ 9:34am

not related to the article but i'd like to see an an item on grub2 too. Managed to tweak it a bit by changing time outs and default boot line. But there may be more...

Find and Grep? Use locate instead

Anonymous Penguin (not verified) - November 2, 2009 @ 3:09pm

I understand the power of find and grep, but using a tool to build a DB when updatedb/locate exists seems noob-ish to me.

Move to Gentoo

Anonymous Penguin (not verified) - November 2, 2009 @ 7:15pm

This article should really be title "How To Make Ubuntu Faster and Lighter".
If anyone is truly interested in performance they wouldn't use Ubuntu in the first place (although it is damn good for most users).
True performance comes from compiling for your particular architecture and there is nothing better than Gentoo.

Or you could just switch to

DaveS (not verified) - November 2, 2009 @ 9:36pm

Or you could just switch to Puppy Linux

timestamps

Anonymous Penguin (not verified) - November 2, 2009 @ 11:04pm

Typo:

net.ipv4.tcp_timestamps = 0

...relieves all ... packets of a 12k overhead

You meant a 12 BYTE overhead.

tux

tux (not verified) - November 3, 2009 @ 2:56pm

bando de fanboy linusers

Just format the hard drive

Anonymous Penguin (not verified) - November 3, 2009 @ 11:35pm

Just format the hard drive and install a real operating system.

Linux is for servers.

Typical Windows-type desktop user

Keith (not verified) - November 4, 2009 @ 4:00pm

In response to "Just format the hard drive": I switched to Linux (Ubuntu) on v7.10 from XP as I got tired of running multiple protective programs to keep XP safe. Under conditions of malware,spyware, and virus blocking, baseline Linux (any distro) is lightspeed faster than XP - no tweeks necessary.

Now for the truth (you can refute but its a worthless argument), Win beats Linux in 3 areas...drivers, codecs, and popular, cutting edge gaming. This is only because hardware vendors write their own drivers for Win, Win pays for codec licensing, and game vendors know mostly (not all) kids play games on computers their parents buy at WalMart or Best Buy.

When it comes to operating systems, Linux is to Win what 7 is to 95. Linux is better in security, software selection, stability, desktop customization, and many other things not generalized like virtual desktops (the usability and economy of virtual desktops just makes the computing experience superior without any aforementioned elements considered)

Finally, I understand most users are not tech savvy, and I am one. We just want to plug in a printer, backup usb drive, and maybe a camera, and just use the computer for pictures, music and socializing (actually, I play on my Linux box trying new software and learning customization options, the command line - just like many people enjoy gaming). Some distros are just not designed for the general public user, and this defense is not for those. However, opensource drivers are being added to Linux every day, and one can find several distros that load codecs automatically, invisible to the user. In fact, I have finally, finally found the Window OS killer, yes the killer!!! after trying many different distros (thank you Distro Watch) - it is SuperOS, an Ubuntu markup. It plays every radiostation I have tried to use-even those optimized for Windows PC - as in mms. It plays all flash content as well. In fact, out-of-the-box, it is perfect, no user tweaks or apt-get calls to repositories, nothing. Just install it from a live CD and start using your computer like you expect from Windows.

Linux still does not play current cutting edge games without some help from outside software, but that is minor considering how many people use computers and do not see it as a gamebox only. That too will come, sorry sir.

To put your not-well-thought-out comment in perspective...Linux/Unix has always been the real operating system, and it was around first, which, again, makes it THE real operating system. Windows is a purchased OS from some private-sector company that has been "tweaked" over the years. Additionally, windows has purchased a market share and spent a fortune doing it. Its not better, really, only more easilly accessible because it is already loaded on the cheap stuff being sold in retail outlets. So when Everquest, Diablo, Guildwars, etc...run natively in Linux, Windows will start its decline and eventually be a relic or classic, whatever your perspective is.

(BTW Windows was just a manager running on DOS, right? And DOS was the OS, right? Now I am not sure...but now Windows is more like Linux with NT, right? - I am waiting to see Win add the virtual desktops)

Have a good day

hparm, services e gksudo

Anonymous Penguin (not verified) - November 5, 2009 @ 1:43pm

I don't understood (or undertand?) how can I make hdparm options permanents using “System > Administration > Services” and gksudo? Am I flying away?

Several corrections

jirij (not verified) - November 5, 2009 @ 6:41pm

I mostly agree with p-static, just to recap:

DMA (and proper udma mode) is set automatically on (I believe) all 2.6.x kernels if the hardware supports it. By "hardware" I mean the HDD, cable, motherboard, ...
I've seen systems with udma5 capable drives using udma2 mode, simply because of the slow bus. No manual setting helped it.

ext3 - smart people does not set journal writeback unless they have a reliable UPS. It's not worth the data loss. Believe me. If you want better performance, stick with ext4 (some tuning needed to ensure data safety). Also note that Ubuntu kernel team probably f*cked something in the ext4 code as there are data corruption reports on ext4 with large files in Ubuntu ([citation needed]).

Disabling IPv6 "just because it still isn't widely used" is somewhat stupid, yet I believe all those people will find a way to enable it once their ISP switches to IPv6. Either by changing that value .. or re-installing whole system just "because I can't use my browser".
If IPv6-enabled solutions hurt performance by more than 0.01%, blacklist ipv6 module.

/etc/hosts -- what the ..?? Right, full range of 127.0.0.0/8 is reserved, but what's the point in doing such stuff here, in a performance tuning guide? Is it there just because Ubuntu has it?

Firefox tweaks are fine, there is about additional ~15 googleable options that speeds the browser up (I'd like to remember all of them, unfortunatelly, it's more than two years since I tweaked them). Moreover, we have the "vacuum" plugin and such stuff, right?

My note about aptitude - it's more a cleaning thing than a performance tune, but it helps - aptitude purge '~c'.
This will delete all configs and config-related files of already removed packages. I found it useful in removing Grub mess in /etc when I switched to LILO on RAID1 system. Use with care.

Now more on the TCP options;
First, TCP timestamps - see RFC 1323. This feature, in fact, improves performance and does even more than that. As someone before me noted, it's not 12k, but 12byte, which is pretty much enough, but it's worth it.
Selective ACKs should be enabled even on dialup connections, there's nothing really costy about it, it was (IIRC) introduced to help the TCP troughput in a packetloss environment, thus gaining "performance". See RFC 2018.
About tcp_no_metrics_save - if it had worked correctly, no tweak would be needed, but it's better to currently enable this option. Gentoo wiki explains it nicely - "This removes an odd behavior in the 2.6 kernels, whereby the kernel stores the slow start threshold for a client between TCP sessions. This can cause undesired results, as a single period of congestion can affect many subsequent connections."
Backlog tweak is fine, but again - totally unneeded unless you use heavy-load torrents or similar stuff. It's default got already increased from 300 to 1000. Note that higher values can impact latency, which is sometimes undesirable.
TCP mem/rmem/wmem settings are fine, just, please, google out what those 3 types of values mean. Setting all of them to the same value isn't "cool" and it may possibly confuse the OOM killer a bit.

I can't judge the rest, I don't use KDE/Gnome/OOo.

Downloads went from ~500KBps

DaVince - November 5, 2009 @ 10:29pm

Downloads went from ~500KBps to ~600KBps over wireless after adding most of the network tricks. Thanks, guys! :)

Thanks

Max (not verified) - November 7, 2009 @ 10:47am

Great information!!
Thanks for this

Very good

Adelar da Silva Queiróz (not verified) - November 9, 2009 @ 12:10am

Great tips in only one article! Your information will be helpful! Thank you :)

Disabling IPV6, not so stupid

Anonymous Beast (not verified) - November 11, 2009 @ 11:40pm

OpenBSD (highly secure system) actually had a bugfix in IPV6 affecting the firewall and system stability, If I remember right.

Anyway, the point is, if you have no reason to use it and it is actually less fixed, shall we say, to avoid debate. Your point about future proofing has validity, but switching IPV6 off, certainly isn't stupid.

Less can be more (runtime, security etc..)
More code, more bugs

Fire

Wishing Master Mind (not verified) - November 14, 2009 @ 11:07am

I think the editor should be fired for not verifying the article tips.So many complaints in the comments.

Please, dudes!

Anonymous Penguin (not verified) - November 19, 2009 @ 11:10pm

"After enabling automatic login and installing preload we initiated two restarts on our test machine to give the daemon a chance to monitor the startups. After comparing the two times, we found that preload had trimmed a second off the time it took to get a usable desktop. This doesn't sound a lot, but if it's used in conjunction with your new fast boot time then it's well worth the effort."

Synaptic says (about Preload):
"Note that installing preload will not make your system boot faster
and that preload is a daemon that runs with root priviledges."

Which one we should to believe?

I agree that the editor should take into account these comments considering that you are light mindedly giving advices without checking what might happen. Please, take some responsibility.

Not actulay faster !!!

filip007 (not verified) - November 28, 2009 @ 10:17am

I have used Concurrency with shell but it's start showing full grub on the second boot and disable auto load!

Ubuntu 9.10 don't have menu.lst any more at lest active not is not so how the can can i fixed this...

Ubuntu 9.10 was so close to be perfect but they wasted it with couple mistakes!

Yes thanks TuxRadar for wasing my time

filip007 (not verified) - November 28, 2009 @ 10:36am

By the /etc/init.d/rc and inside text Shell is obsolete.

Next focus how to fix boot errors like usplash 1152x864 failed...

Thanks from Newbie

Ikhwan (not verified) - March 13, 2010 @ 3:45pm

I learn so much from this article, thanks.

Update this please?

Greg Zeng (not verified) - September 11, 2010 @ 1:45am

Thanks for the original post & some of the helpful commments that followed.

Now we are generally on EXT4, Ubuntu-10.04 based Linuxs': latest SUPEROS, MINT, etc. Some of the previous hints work, but I needed SUDO B4 they could work.

Could the original author update his is post? Or some genius start a new one that GOOGLE will help us b4 this one does ... please.

Greg Zeng, Retired (medical) IT Consultant,
Australian Capital Territory

Thank's

Norberto Moritz (not verified) - September 29, 2011 @ 11:40pm

My Mageia it's now a bit faster.

Comment viewing options

Threaded list - expanded ▼ Date - oldest first ▼ 50 comments per page ▼ Save settings

Select your preferred way to display the comments and click "Save settings" to activate your changes.

Username: Password: Log in

Top ▲

Terms and conditions | Privacy policy | Cookie policy | Advertise with us |