

# ficheros del home

## .bash\_profile

```
# ~/.bash_profile: executed by bash(1) for login shells.

# quita el permiso de escritura al others
umask 002

#prompt por defecto
PS1='\h]$ '

# prompt de color en 2 lineas
if [ -f ".prompt" ]; then
    export PS1=`cat .prompt`
fi

if [ -f ".bashrc" ]; then
    source ~/.bashrc
fi

# variables de entorno
export EDITOR="vim"
export LANG=en_US.ISO-8859-15

# echo "> .bash_profile"
```

## .bashrc

```
# ~/.bashrc: executed by bash(1) for non-login shells.
source ~/.alias

# funciones interesantes
source ~/.functions

# exports
source ~/.export

# si deajo el ECHO da problemas con scp
# echo "> .bashrc"
```

## .alias

```
# $Id: .alias,v 1.1 1999/11/27 23:01:40 sage Exp $
# Luis Francisco González <luisgh@debian.org> based on that of Vadik Vygonets
# Please check /usr/doc/tcsh/examples/alias to see other possible values.

# recopilación de alias
# opciones LS
```

```
alias l.='ls -d .* --color=tty'
alias ll='ls -l --color=tty'
alias ls='ls --color=tty'
alias lsd='ls -d */'
# evitar errores (confirmación al sobrescribir)
alias cp='cp -i'
# alias mv='mv -i'
# alias rm='rm -i'

# abreviaturas
alias rd='rmdir'
alias md='mkdir'

# errores
alias cd.='cd ..'
alias moer='more'
alias mroe='more'

# funcionalidades
alias cdo='cd $OLDPWD'
users='cut -d: -f1 /etc/passwd | sort'
users2='getent passwd|cut -d: -f1|sort'
```

## .functions

### .functions

```
# reiniciar demonios
restart() {
    if [ -f /etc/init.d/$1 ]; then
        /etc/init.d/$1 restart
    else
        echo "*** non existant daemon $1 ***"
    fi
}

# calculadora
function calc { echo "${1}"|bc -l; }

# Generador de Passwords (requiere paquete agp)
function gen-pass { agp -a 0 -n 1 -q -MNCL -m 12; }

# Envía mails para recordar tareas
function task {
    mail -s "$*" mate@yoquese.es < /dev/null 2>/dev/null
}

# lista de últimos archivos
lsnew() { ls -lt ${1+"$@"} | head -20; }

# host en mis subnets con puerto SSH abierto
sssearch() { nmap -p22 -oG - $(ip addr | awk '/inet .*global/ {print $2}' | tr
'\n' ' ') | awk '/22\./open/ {print $2}' | uniq;}

```

```

# convertir ficheros NRG (nero) en ISO
nrg2iso() { dd bs=1 if=$1 of=$2 skip=300; }

# tree == subdirectorios y capacidades en human readable
tree() { du -Sh | sort -n; }

# mirar ficheros con permisos de escritura (potencialmente peligrosos en
entorno multiusuario)
# hay 2 variantes para hacer solo de determinados directorios o para eliminar
las lineas que no nos interesan
writable() {
    find / -perm -2 ! -type l -ls;
    # find /etc /var -perm -2 ! -type l -ls;
    # find / -perm -2 ! -type l -ls | sed "/proc/d" | sed "/dev/d";
}

# comandos favoritos
favorites() { history | awk '{print $2}' | sort | uniq -c | sort -rn | head; }
#favorites2() { history | awk '{print $2}' | awk 'BEGIN {FS="|"}{print $1}' |
sort | uniq -c | sort -n | tail | sort -nr }

```

## .export

```

# utilizar LESS en lugar de MAN y utilizar colores
export LESS_TERMCAP_mb=$'\E[01;31m'
export LESS_TERMCAP_md=$'\E[01;37m'
export LESS_TERMCAP_me=$'\E[0m'
export LESS_TERMCAP_se=$'\E[0m'
export LESS_TERMCAP_so=$'\E[01;44;33m'
export LESS_TERMCAP_ue=$'\E[0m'
export LESS_TERMCAP_us=$'\E[01;32m'

```

## DIR\_COLORS

```

# Coreutils version coreutils-6.10
# If you do not modify this file, it will be updated when you upgrade
# coreutils, using the contents of /etc/defaults/etc/DIR_COLORS. Feel
# free to modify this file, at which point coreutils will leave it alone.
#
# For this file to be useful, you will need a line like this in your
# shell configuration file, such as ~/.bashrc:
#
#   alias ls='ls --color=auto'
#
# If you want to use a variant of this file in your home directory,
# then you can add something like this to your ~/.bashrc:
#
#   eval "`dircolors -b /path/to/DIR_COLORS`"
#
# Configuration file for dircolors, a utility to help you set the

```

```
# LS_COLORS environment variable used by GNU ls with the --color option.

# Copyright (C) 1996, 1999-2008
# Free Software Foundation, Inc.
# Copying and distribution of this file, with or without modification,
# are permitted provided the copyright notice and this notice are preserved.

# The keywords COLOR, OPTIONS, and EIGHTBIT (honored by the
# slackware version of dircolors) are recognized but ignored.

# Below, there should be one TERM entry for each termtyp that is colorizable
TERM Eterm
TERM ansi
TERM color-xterm
TERM con132x25
TERM con132x30
TERM con132x43
TERM con132x60
TERM con80x25
TERM con80x28
TERM con80x30
TERM con80x43
TERM con80x50
TERM con80x60
TERM cons25
TERM console
TERM cygwin
TERM dtterm
TERM eterm-color
TERM gnome
TERM gnome-256color
TERM konsole
TERM kterm
TERM linux
TERM linux-c
TERM mach-color
TERM mlterm
TERM putty
TERM rxvt
TERM rxvt-cygwin
TERM rxvt-cygwin-native
TERM rxvt-unicode
TERM screen
TERM screen-256color
TERM screen-bce
TERM screen-w
TERM screen.linux
TERM vt100
TERM xterm
TERM xterm-16color
TERM xterm-256color
TERM xterm-88color
TERM xterm-color
TERM xterm-debian

# Below are the color init strings for the basic file types. A color init
```

```
# string consists of one or more of the following numeric codes:
# Attribute codes:
# 00=none 01=bold 04=underscore 05=blink 07=reverse 08=concealed
# Text color codes:
# 30=black 31=red 32=green 33=yellow 34=blue 35=magenta 36=cyan 37=white
# Background color codes:
# 40=black 41=red 42=green 43=yellow 44=blue 45=magenta 46=cyan 47=white
NORMAL 00      # global default, although everything should be something.
FILE 00       # normal file
DIR 01;34     # directory
LINK 01;36    # symbolic link. (If you set this to 'target' instead of a
# numerical value, the color is as for the file pointed to.)

FIFO 40;33    # pipe
SOCK 01;35    # socket
DOOR 01;35    # door
BLK 40;33;01  # block device driver
CHR 40;33;01  # character device driver
ORPHAN 40;31;01 # symlink to nonexistent file, or non-stat'able file
SETUID 37;41   # file that is setuid (u+s)
SETGID 30;43   # file that is setgid (g+s)
STICKY_OTHER_WRITABLE 30;42 # dir that is sticky and other-writable (+t,o+w)
OTHER_WRITABLE 34;42 # dir that is other-writable (o+w) and not sticky
STICKY 37;44   # dir with the sticky bit set (+t) and not other-writable

# This is for files with execute permission:
EXEC 01;32

# List any file extensions like '.gz' or '.tar' that you would like ls
# to colorize below. Put the extension, a space, and the color init string.
# (and any comments you want to add after a '#')

# If you use DOS-style suffixes, you may want to uncomment the following:
#.cmd 01;32 # executables (bright green)
#.exe 01;32
#.com 01;32
#.bat 01;32
# Or if you want to colorize scripts even if they do not have the
# executable bit actually set.
#.sh 01;32
#.csh 01;32

# archives or compressed (bright red)
.tar 01;31
.tgz 01;31
.svgz 01;31
.arj 01;31
.taz 01;31
.lzh 01;31
.lzma 01;31
.zip 01;31
.z 01;31
.Z 01;31
.dz 01;31
.gz 01;31
.bz2 01;31
```

```
.bz 01;31  
.tbz2 01;31  
.tz 01;31  
.deb 01;31  
.rpm 01;31  
.jar 01;31  
.rar 01;31  
.ace 01;31  
.zoo 01;31  
.cpio 01;31  
.7z 01;31  
.rz 01;31
```

*# image formats*

```
.jpg 01;35  
.jpeg 01;35  
.gif 01;35  
.bmp 01;35  
.pbm 01;35  
.pgm 01;35  
.ppm 01;35  
.tga 01;35  
.xbm 01;35  
.xpm 01;35  
.tif 01;35  
.tiff 01;35  
.png 01;35  
.svg 01;35  
.mng 01;35  
.pcx 01;35  
.mov 01;35  
.mpg 01;35  
.mpeg 01;35  
.m2v 01;35  
.mkv 01;35  
.ogm 01;35  
.mp4 01;35  
.m4v 01;35  
.mp4v 01;35  
.vob 01;35  
.qt 01;35  
.nuv 01;35  
.wmv 01;35  
.asf 01;35  
.rm 01;35  
.rmvb 01;35  
.flc 01;35  
.avi 01;35  
.fli 01;35  
.gl 01;35  
.dl 01;35  
.xcf 01;35  
.xwd 01;35  
.yuv 01;35
```

*# audio formats*

```
.aac 00;36
.au 00;36
.flac 00;36
.mid 00;36
.midi 00;36
.mka 00;36
.mp3 00;36
.mpc 00;36
.ogg 00;36
.ra 00;36
.wav 00;36
```

## .prompt

```
# para poner en PS1, 2 líneas, en color, la primera user@host pwd, la segunda el
shell
\[ \e[0m\;\w\|a\]\n\[ \e[32m\]\u@h \[ \e[33m\]\w\[ \e[0m\]\n\$
```

## .vimrc

```
" Configuration file for vim
set
runtimepath=~/.vim,/etc/vim,/usr/share/vim/vimfiles,/usr/share/vim/addons,/usr/share/vim/vim63,/usr/share/vim/vimfiles,/usr/share/vim/addons/after,~/.vim/after

" Normally we use vim-extensions. If you want true vi-compatibility
" remove change the following statements
set nocompatible          " Use Vim defaults instead of 100% vi compatibility
set backspace=indent,eol,start " more powerful backspacing

" Now we set some defaults for the editor
set autoindent            " always set autoindenting on
" set linebreak           " Don't wrap words by default
set textwidth=0          " Don't wrap lines by default
set backupcopy=yes       " Keep a backup file
set viminfo='20,\,"50    " read/write a .viminfo file, don't store more than
                          " 50 lines of registers
set history=50           " keep 50 lines of command line history
set ruler                " show the cursor position all the time

" Suffixes that get lower priority when doing tab completion for filenames.
" These are files we are not likely to want to edit or read.
set
suffixes=.bak,~, .swp, .o, .info, .aux, .log, .dvi, .bbl, .blg, .brf, .cb, .ind, .idx, .ilg, .inx, .out, .toc

" We know xterm-debian is a color terminal
if &term =~ "xterm-debian" || &term =~ "xterm-xfree86"
    set t_Co=16
    set t_Sf=m
    set t_Sb=m
endif
```

```
" Make p in Visual mode replace the selected text with the "" register.
vnoremap p <Esc>:let current_reg = @"<CR>gvdi<C-R>=current_reg<CR><Esc>

" Vim5 and later versions support syntax highlighting. Uncommenting the next
" line enables syntax highlighting by default.
syntax on
set background=dark
" If using a dark background within the editing area and syntax highlighting
" turn on this option as well
" set background=dark

if has("autocmd")
  " Enabled file type detection
  " Use the default filetype settings. If you also want to load indent files
  " to automatically do language-dependent indenting add 'indent' as well.
  filetype plugin on

endif " has ("autocmd")

" Some Debian-specific things
augroup filetype
  au BufRead reportbug.*          set ft=mail
  au BufRead reportbug-*          set ft=mail
augroup END

" Set paper size from /etc/papersize if available (Debian-specific)
try
  if filereadable('/etc/papersize')
    let s:papersize = matchstr(system('/bin/cat /etc/papersize'), '\p*')
    if strlen(s:papersize)
      let &printoptions = "paper:" . s:papersize
    endif
    unlet! s:papersize
  endif
catch /E145/
endtry

" The following are commented out as they cause vim to behave a lot
" different from regular vi. They are highly recommended though.
"set showcmd          " Show (partial) command in status line.
"set showmatch        " Show matching brackets.
"set ignorecase       " Do case insensitive matching
"set incsearch        " Incremental search
"set autowrite        " Automatically save before commands like :next and :make

" Source a global configuration file if available
if filereadable("/etc/vim/vimrc.local")
  source /etc/vim/vimrc.local
endif
```

From:

<https://miguelangel.torresegea.es/wiki/> - **miguel angel torres egea**

Permanent link:

<https://miguelangel.torresegea.es/wiki/linux:scripts:home?rev=1317202939>

Last update: **15/11/2011 01:59**

