Stefaan Lippens

- BLOG OF STEFAAN LIPPENS

- TAGS

- ARCHIVE

- ABOUT

My bashrc, bash aliases, profile and other files

BY STEFAAN LIPPENS ON 2009/10/19

TAGGED: BASH

I spend a lot of time on the (bash) command line in (Linux and OS x) and it's always good for productivity to have your favorite set of shortcuts, aliases, perferences, predefined (environment) variables and such at your fingertips. You know, all the stuff that lives in `.bashrc`, `.profile`, `.vimrc`, `.git`, etc.

I have access to several systems (at home, at work, remote servers, etc) and from time to time I'm sitting behind a system that doesn't know my preferences yet. I considered using some sort of version control system to "synchronize" these preference files between my setups, but (apart from the practical hurdles) it generally seemed overkill, especially since each setup can have its own peculiarities and customizations.

So, mainly for my own reference, I'm keeping an list of the most important general stuff here. Be welcome to cherry-pick (or suggest additions).

`.bashrc`

In `.bashrc`, I put the things that are related to *interactive* Bash usage. You'll also find a lot of references that configure environment variables like `PATH` in `.bashrc`, but I prefer to put these in `.profile` (see lower). `.bashrc` should be loaded when opening a new interactive Bash shell.

```bash
# Enable programmable completion features.

if [ -f /etc/bash_completion ]; then

    source /etc/bash_completion
```

```bash
fi


# Set the PS1 prompt (with colors).

# Based on http://www-128.ibm.com/developerworks/linux/library/l-tip-prompt/

# And http://networking.ringofsaturn.com/Unix/Bash-prompts.php .

PS1="\[\e[36;1m\]\h:\[\e[32;1m\]\w$ \[\e[0m\]"


# Set the default editor to vim.

export EDITOR=vim


# Avoid succesive duplicates in the bash command history.

export HISTCONTROL=ignoredups


# Append commands to the bash command history file (~/.bash_history)

# instead of overwriting it.

shopt -s histappend


# Append commands to the history every time a prompt is shown,

# instead of after closing the session.

PROMPT_COMMAND='history -a'


# Add bash aliases.

if [ -f ~/.bash_aliases ]; then

    source ~/.bash_aliases

fi
```

.bash_aliases

I use a separate file .bash_aliases for my favorite Bash aliases, to keep things more tidy. This file is "sourced" by .bashrc (see higher).

```
# Make some possibly destructive commands more interactive.

alias rm='rm -i'

alias mv='mv -i'

alias cp='cp -i'


# Add some easy shortcuts for formatted directory listings and add a touch of
color.

alias ll='ls -lF --color=auto'

alias la='ls -alF --color=auto'

alias ls='ls -F'


# Make grep more user friendly by highlighting matches

# and exclude grepping through .svn folders.

alias grep='grep --color=auto --exclude-dir=\.svn'


# Shortcut for using the Kdiff3 tool for svn diffs.

alias svnkdiff3='svn diff --diff-cmd kdiff3'
```

.profile

I put the "run time modifying" stuff (PATH and friends) in .profile, so that these things are also
easily available outside interactive Bash sessions, like non-interactive Bash sessions, other
command line shells sessions and even the graphical shell/desktop environment.
The .profile file should be loaded on login on Linux setups.

```
# Add some more custom software to PATH.

PATH=$PATH:~/usr/bin

export PATH


# Make sure pkg-config can find self-compiled software

# and libraries (installed to ~/usr)

PKG_CONFIG_PATH=$PKG_CONFIG_PATH:~/usr/lib/pkgconfig
```

```
export PKG_CONFIG_PATH


# Add custom compiled libraries to library search path.

LD_LIBRARY_PATH=$LD_LIBRARY_PATH:~/usr/lib

export LD_LIBRARY_PATH


# Add custom compiled libraries to library run path.

LD_RUN_PATH=$LD_RUN_PATH:~/usr/lib

export LD_RUN_PATH


# Add custom Python modules to the Python path.

PYTHONPATH=$PYTHONPATH:~/slippenspythonlib

PYTHONPATH=$PYTHONPATH:~/usr/lib/python2.5/site-packages

export PYTHONPATH


# Java;s CLASSPATH customization

CLASSPATH=$CLASSPATH:~/foo/bar.jar

export CLASSPATH
```

.bash_profile

As I've described in a separate blog post about `.bashrc`, `.profile` and others, deciding where to put things may seem a bit hairy. I limit myself to `.bashrc` and `.profile`, and use `.bash_profile` to reuse these for (textual) login shells (see that blog post for more info).

```
# Load .profile, containing login, non-bash related initializations.

source ~/.profile


# Load .bashrc, containing non-login related bash initializations.

source ~/.bashrc
```

.screenrc

[GNU Screen](#) is a very handy tool I should have learned earlier about. It lets you have persistent shell sessions, so you can log out on one computer and reconnect to the still running session from an other computer. Recommended if you're "commandlining" a lot in an networked environment. Anyway, here is my `.screenrc` to have a handy caption bar, listing the available windows.

```
# Set the default window name to empty string instead of the arbitrary "bash"

shelltitle ''



# Set the window caption.

# I use caption instead of hardstatus, so it is available per split window too

# (hardstatus is only per complete screen).

caption always "%{= KW}%-Lw%{= wb}%n %t %{= KW}%+Lw %-=| ${USER}@%H | %M%d %c%{-}"

# Some decryption hints:

# %{= KW}     background light black (aka dark gray) with foreground light white

# %{= wb}     background dark white (ake light gray) with foreground dark blue

# %-Lw        all windows before the current window.

# %n%f %t     current window number, flags and title.

# %+Lw        all windows after the current window.

# %-=         pad remaining spaces.

# %H          hostname.

# %M%d %s     month and day (MmmDD) and current time (HH:MM).
```