The Lumber Room

"Consign them to dust and damp by way of preserving them"

Zsh/Bash startup files loading order (.bashrc, .zshrc etc.)

with 31 comments

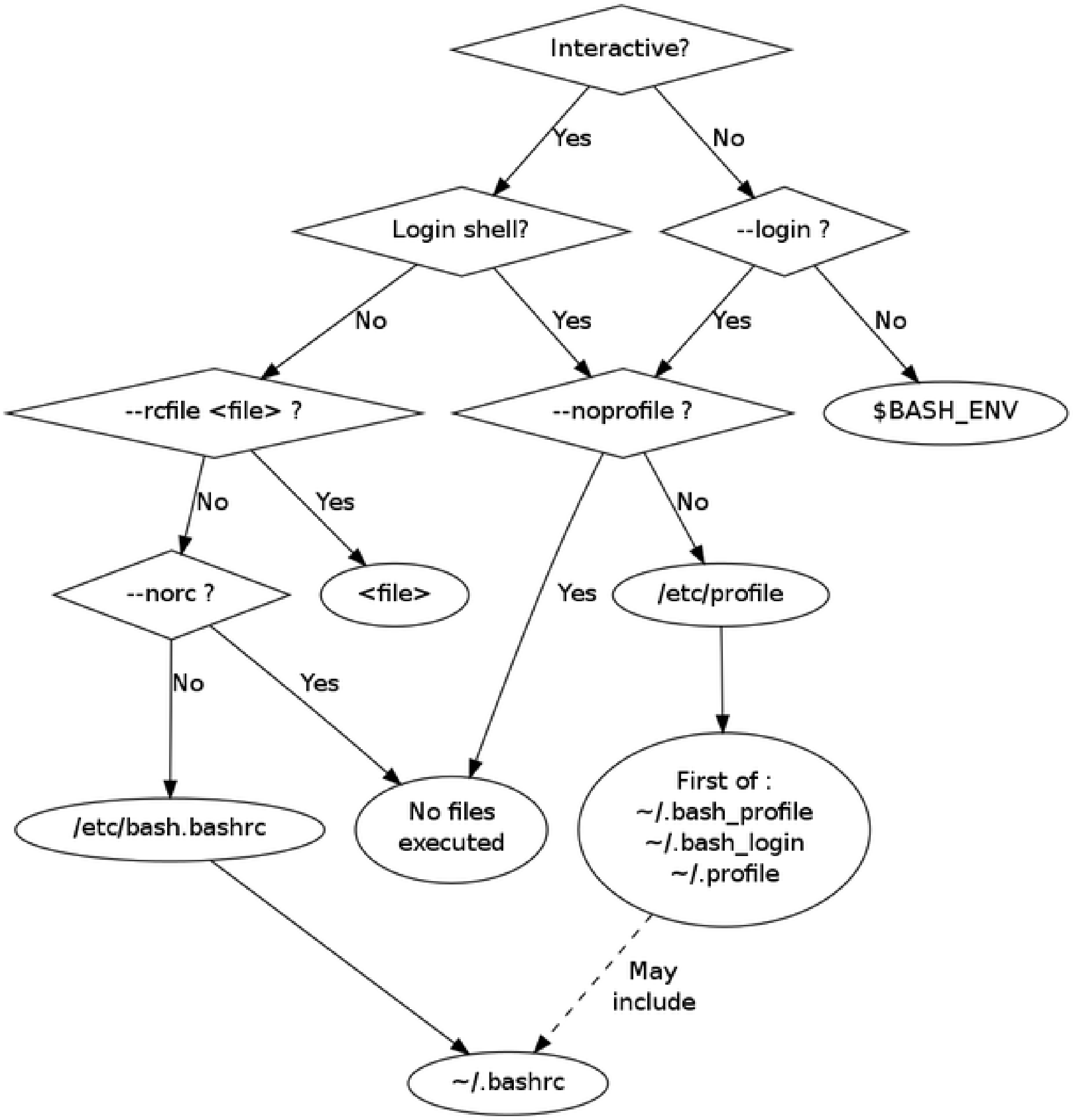
If you have ever put something in a file like `.bashrc` and had it not work, or are confused by why there are so many different files — `.bashrc`, `.bash_profile`, `.bash_login`, `.profile` etc. — and what they do, this is for you.

The issue is that Bash sources from a different file based on what kind of shell it thinks it is in. For an “interactive non-login shell”, it reads `.bashrc`, but for an “interactive login shell” it reads from the *first* of .bash_profile, .bash_login and .profile (only). There is no sane reason why this should be so; it’s just historical. Follows in more detail.

For Bash, they work as follows. Read down the appropriate column. Executes A, then B, then C, etc. The B1, B2, B3 means it executes only the first of those files found.

```
+----------------+-----------+-----------+------+
|                |Interactive|Interactive|Script|
|                |login      |non-login  |      |
+----------------+-----------+-----------+------+
|/etc/profile    |   A       |           |      |
+----------------+-----------+-----------+------+
|/etc/bash.bashrc|           |    A      |      |
+----------------+-----------+-----------+------+
|~/.bashrc       |           |    B      |      |
+----------------+-----------+-----------+------+
|~/.bash_profile |   B1      |           |      |
+----------------+-----------+-----------+------+
|~/.bash_login   |   B2      |           |      |
+----------------+-----------+-----------+------+
|~/.profile      |   B3      |           |      |
+----------------+-----------+-----------+------+
|BASH_ENV        |           |           |  A   |
+----------------+-----------+-----------+------+
|                |           |           |      |
+----------------+-----------+-----------+------+
|                |           |           |      |
+----------------+-----------+-----------+------+
|~/.bash_logout  |    C      |           |      |
+----------------+-----------+-----------+------+
```

In more detail is this excellent flowchart from http://www.solipsys.co.uk/new/BashInitialisationFiles.html :



From http://www.solipsys.co.uk/new/BashInitialisationFiles.html

Typically, most users will encounter a login shell only if either:
* they logged in from a tty, not through a GUI
* they logged in remotely, such as through ssh.
If the shell was started any other way, such as through GNOME's `gnome-terminal` or KDE's `konsole`, then it is typically not a login shell — the login shell was what started GNOME or KDE behind your back when you logged in; things started anew are not login shells. New terminals or new screen windows you open are not login shells either. (Starting a new window in OS X's Terminal.app seems to count as a login shell, though.)

So typically (or sooner or later), what you will encounter are non-login shells. So this case is what you should write your config files for. This means putting most of your stuff in `~/.bashrc`, having exactly one of `~/.bash_profile`, `~/.bash_login`, and `~/.profile`, and sourcing `~/.bashrc` from it. If you have nothing that you specifically want to happen only for login shells, you can even symlink one of the three to `~/.bashrc`. In fact, even if you do, it is probably a good idea to have only file, as follows:

```
# Bash customisation file

#General configuration starts: stuff that you always want executed

#General configuration ends

if [[ -n $PS1 ]]; then
    : # These are executed only for interactive shells
    echo "interactive"
else
    : # Only for NON-interactive shells
fi

if shopt -q login_shell ; then
    : # These are executed only when it is a login shell
    echo "login"
else
    : # Only when it is NOT a login shell
    echo "nonlogin"
fi
```

Almost everything should go in the “general configuration” section. There might be some commands (those which produce output, etc.) that you only want executed when the shell is interactive, and not in scripts, which you can put in the first “conditional section”. I don’t see any reason to use the rest. You can drop the “echo” lines, but keep the “:”s — they are commands which do nothing, and are needed if that section is empty.

You then need to have only file, and you can call this `~/.bashrc` and do `cd && ln -s .bashrc .bash_profile`

For zsh: [Note that zsh seems to read `~/.profile` as well, if `~/.zshrc` is not present.]

	Interactive login	Interactive non-login	Script
/etc/zshenv	A	A	A
~/.zshenv	B	B	B
/etc/zprofile	C		
~/.zprofile	D		
/etc/zshrc	E	C	
~/.zshrc	F	D	
/etc/zlogin	G		
~/.zlogin	H		
~/.zlogout	I		
/etc/zlogout	J		

Moral:
For bash, put stuff in ~/.bashrc, and make ~/.bash_profile source it.
For zsh, put stuff in ~/.zshrc, which is always executed.

[Note: This is assuming you care only about interactive shells (shells you can type at). If further you ever use non-interactive shells (like “ssh [host] [command]” which executes [command] on [host] and logs out immediately) *and* there is something you absolutely absolutely want executed first even for these cases (think carefully: they can interfere in strange ways and you may be able to do without them), then you can put such stuff in ~/.zshenv. For bash, put them in a file and set the value of BASH_ENV to the filename.]

Thanks to this blog post and the manpages for information and ideas.

Written by S

Sun, 2008-03-30 at 19:43:56

Posted in compknow

Tagged with bash, compknow, shell, zsh

31 Responses

Subscribe to comments with RSS.

[…] files (~/.bashrc, ~/.bash_profile, ~/.bash_login), restart your terminal, and you are good to go. Here is a great article about how bash configs work. I put tgrep() in my .bashrc file, but also added […]

Tony Amoyal's Blog » Ignore svn directories while you grep

Thu, 2008-12-04 at 16:59:13

Reply

[…] Zsh/Bash startup files loading order (.bashrc, .zshrc etc.) (tags: sysadmin linux zsh) […]

links for 2009-01-25 « Bloggitation

Mon, 2009-01-26 at 07:01:56

Reply

[…] for example.) More generally, it's happening because some command in your ~/.bashrc (or some other file sourced automatically) is expecting the shell to be interactive, when it's not. The fix is to find out which […]

stderr is not a tty - where are you? « The Lumber Room

Fri, 2009-01-30 at 19:10:33

Reply

[…] Zsh/Bash startup files loading order (.bashrc, .zshrc etc.) « The Lumber Room […]

Zsh/Bash startup files loading order | sassy grrrl

Mon, 2010-02-01 at 16:40:42

Reply

Thanks for putting this all together in such a clear and informative way :)
I've referred people to this blog post several times.

Josh

Wed, 2011-01-05 at 01:41:13

Reply

> You're welcome, very glad it helped. That was exactly my purpose in putting this together; I have myself referred to it several times when I was at a new computer. :-)
>
> **S**

Wed, 2011-01-05 at 07:07:07

Reply

This is an amazing reference.

saikobee

Mon, 2011-01-17 at 14:43:07

Reply

Great table; I found this very helpful!

One minor quibble: you say “For zsh, put stuff in ~/.zshrc, which is always executed,” but .zshrc is not always executed. Do you mean .zshenv? Either way, I think there is a good reason to use both files: .zshenv should never write anything to stdout. If it does, then things like “ssh [myhost] [mycommand]” will return unexpected results.

However, it may be necessary to set up some environment variables on [myhost] before running [mycommand] — this is what .zshenv is for! Anything else, particularly anything that writes to stdout, should go in .zshrc.

Drew Frank

Thu, 2011-06-16 at 18:51:00

Reply

[Edit: I’ve updated it, thanks.]

Thanks. I guess when I wrote that, I only cared about interactive shells, and only wanted not to have to bother about login versus non-login shells. If you notice, the instructions for Bash similarly only cover interactive shells. :-) Actually, Bash doesn’t even have the equivalent of ~/.zshenv which is executed for non-interactive shells, so IMO it’s not crucial.
I’m not convinced there is really anything I absolutely want done on remote systems when running things like “ssh [myhost] [mycommand]”, but yes, if someone wants it then it’s a good idea to mention it in the post as well.

S

Fri, 2011-06-17 at 00:49:30

Reply

Awesome article, I’ve been looking for this. People are lazy to read manuals etc… It can’t be more simple. Thanks.

Ahmet Yasin Uslu

Fri, 2011-08-12 at 05:48:58

Reply

it helped me lot. thanks

Anonymous

Mon, 2011-11-07 at 16:37:12

Reply
thanks a lot. the post is informative.

zhang zhenfang (@fangzhzh)

Wed, 2012-02-08 at 10:35:03

Reply
There is a wonderful diagram of the Bash initialization process here:
http://www.solipsys.co.uk/new/BashInitialisationFiles.html?HN0811

Jonas Gorauskas

Mon, 2012-09-03 at 08:01:48

Reply

> Really cool. Thanks I've gone ahead and added it to the article.
>
> **S**
>
> Mon, 2012-09-03 at 10:44:19
>
> Reply

Nice start, but you still need to account for /etc/bashrc, /etc/bashrc.local, /etc/bash.bashrc.local, /etc/rc.common, /etc/rc.conf, /etc/rc.local, and more, depending on system. :-) E.g., for bash on Mac OS X 10.8, /etc/rc.common loads first, then /etc/profile then /etc/bashrc (there are no bash.*), etc.

Stanton McCandlish

Tue, 2013-10-22 at 13:54:33

Reply
[…] from testing for interative terminal in .bashrc, bash_profile vs bashrc and Zsh/Bash startup files loading order (.bashrc, .zshrc etc.) […]

How to ssh/putty without password (using public key authentication) | Rui's Blog

Tue, 2014-12-23 at 18:16:45

Reply
[…] make this addition persistent you can also add that to your .profile, .bash_profile or .zprofile depending on the shell and setup you […]

Black Magic Discovery | Twisted World of Esden

Mon, 2014-12-29 at 14:51:59

Reply
[…] make this addition persistent you can also add that to your .profile, .bash_profile or .zprofile depending on the shell and setup you […]

libopencm3 Discovery | Twisted World of Esden

Mon, 2014-12-29 at 15:26:54

Reply

[…] In order, your machine will load/execute the following files when ZSH starts: […]

Profiling ZSH startup time | Kevin Burke

Tue, 2015-03-31 at 18:59:44

Reply

[…] After reading this post I've tweaked the setup to put everything in .bashrc and then source that from .bash_profile (and […]

My Mac OS X Bash Profile - Gilbert Pellegrom

Thu, 2015-06-25 at 02:48:34

Reply

[…] Zsh/Bash startup files loading order (.bashrc, .zshrc etc.) […]

Bash Environment – Jeremiah's Blog

Thu, 2016-03-31 at 10:08:10

Reply

[…] Information on how terminals read settings at https://shreevatsa.wordpress.com/2008/03/30/zshbash-startup-files-loading-order-bashrc-zshrc-etc/ […]

DevOps Engineering part 1. (Mac) – Install the DevOps development tools on Macintosh – Pinter Computing

Thu, 2016-12-08 at 15:04:22

Reply

This is really great! Thanks a lot for writing this up.

이상현 (@sangdolha)

Sun, 2017-02-19 at 08:20:33

Reply

Loved it :D

Nadim Tuhin

Wed, 2017-05-10 at 15:31:18

Reply

[…] https://shreevatsa.wordpress.com/2008/03/30/zshbash-startup-files-loading-order-bashrc-zshrc-etc/ […]

Zsh/Bash startup files loading order (.bashrc, .zshrc etc.) ← Bookmarks

Sat, 2017-07-08 at 06:07:53

Reply
[…] [bashinit]:https://shreevatsa.wordpress.com/2008/03/30/zshbash-startup-files-loading-order-bashrc-zshrc-etc/ […]

快速自检电脑是否被黑客入侵过（Linux版） | 闲聊地–IcySun

Sat, 2017-12-23 at 17:27:22

Reply
[…] [bashinit]:https://shreevatsa.wordpress.com/2008/03/30/zshbash-startup-files-loading-order-bashrc-zshrc-etc/ […]

快速自检电脑是否被黑客入侵过（Linux版） – 安百科技

Sun, 2017-12-24 at 16:46:35

Reply
[…] [bashinit]:https://shreevatsa.wordpress.com/2008/03/30/zshbash-startup-files-loading-order-bashrc-zshrc-etc/ […]

快速自检电脑是否被黑客入侵过（Linux版） | 博客记录系统

Fri, 2018-01-05 at 13:29:44

Reply
[…] Reference: https://shreevatsa.wordpress.com/2008/03/30/zshbash-startup-files-loading-order-bashrc-zshrc-etc/ […]

Set environment variable permanently in Linux – The Normal Programmer

Sun, 2019-01-13 at 21:17:47

Reply
coool graph…

Anonymous

Thu, 2019-05-09 at 20:29:44

Reply
[…] Zsh/Bash startup files loading order (.bashrc, .zshrc etc.) […]

<Linux Shell 脚本攻略>学习笔记 | 大专栏

Wed, 2019-08-21 at 04:57:15

Reply