

Shell Config Files on macOS & Linux

Zsh default on macOS

FILE	LOADS WHEN	PUT HERE
<code>~/.zshenv</code>	every zsh (incl. scripts & non-interactive)	Universal env vars (PATH, EDITOR, tokens). Keep light — runs constantly.
<code>~/.zprofile</code>	login shells only	Once-per-login setup — <code>eval "\$(/opt/homebrew/bin/brew shellenv)"</code> .
<code>~/.zshrc</code>	interactive shells	Prompt, aliases, completions, plugins, key bindings, history.
<code>~/.zlogin</code>	after <code>.zshrc</code> on login	Rarely needed — login banners, final login-time commands.
<code>~/.zlogout</code>	login shell exits	Cleanup — clear screen, unmount, final logs.

Bash Linux default & legacy

FILE	LOADS WHEN	PUT HERE
<code>~/.bash_profile</code>	login shells (first of 3 tried)	Login env & PATH. If present, <code>.profile</code> is skipped.
<code>~/.bash_login</code>	login only if <code>.bash_profile</code> absent	Legacy fallback — rarely used today.
<code>~/.profile</code>	login only if neither above exists	POSIX-portable env. Read by sh, dash, display managers on Linux.
<code>~/.bashrc</code>	interactive non-login shells	Aliases, prompt, functions. macOS Terminal opens login shells → doesn't auto-source this.
<code>~/.bash_logout</code>	login shell exits	Cleanup on exit.

Shared & System all shells / root-owned

FILE	SCOPE	NOTES
<code>/etc/zshenv</code>	system · every zsh	System-wide env. On macOS, sources <code>path_helper</code> .
<code>/etc/zprofile, /etc/zshrc</code>	system · login / interactive	System counterparts. <code>/etc/zprofile</code> runs <code>path_helper</code> on macOS — can reshuffle PATH.
<code>/etc/profile</code>	system · all login shells	Sourced by bash/sh/dash on login. Linux sources <code>/etc/profile.d/*.sh</code> from here.
<code>/etc/bashrc · ~/.inputrc</code>	interactive bash · readline	<code>~/.inputrc</code> configures readline (keybindings, completion) for all bash/readline apps.
<code>/etc/environment</code>	Linux only · PAM login	Plain KEY=VALUE, no shell syntax. Reaches GUI sessions. Not read by macOS.

SOURCING ORDER What runs, when

Zsh · login TERMINAL.APP OPENS THIS
`/etc/zshenv` → `~/.zshenv` → `/etc/zprofile` → `~/.zprofile` → `/etc/zshrc` →

Zsh · interactive non-login VS CODE, TMUX PANES
`/etc/zshenv` → `~/.zshenv` → `/etc/zshrc` → `~/.zshrc`
profile files are skipped — rely on .zshenv for env

Zsh · scripts #!/BIN/ZSH, NON-INTERACTIVE
`/etc/zshenv` → `~/.zshenv` (that's it)

Bash · login MACOS TERMINAL, SSH
`/etc/profile` → first of `~/.bash_profile` · `~/.bash_login` · `~/.profile`
only the first existing file wins — the rest are skipped

Bash · interactive non-login MOST LINUX TERMINALS
`/etc/bash.bashrc` → `~/.bashrc`

RULES & GOTCHAS What to put where & what bites

WHERE TO PUT IT

- **Env vars** → `.zshenv` (zsh) or `.profile` (bash).
- **PATH on macOS** → `.zprofile`, after Homebrew's `shellenv`.
- **Aliases & prompt** → `.zshrc` / `.bashrc` only.
- **Keep `.zshenv` fast** — it runs for every script too.
- **GUI apps on macOS** → use `launchctl setenv` or a LaunchAgent; shell files don't reach them.
- **Match your terminal** — check `echo $0` to see if you're login (`-zsh`) or not (`zsh`).

```
# ~/.zprofile — login once, for PATH
eval "$( /opt/homebrew/bin/brew shellenv )"

# ~/.zshenv — always available
export EDITOR="nvim"
export PAGER="less -R"

# ~/.zshrc — interactive comforts
alias ll="ls -lah"
autoload -Uz compinit && compinit
bindkey -e
```

TRAPS THAT WASTE HOURS

- 01 Non-interactive scripts** don't read `.zshrc`. Aliases & functions defined there are invisible to `ssh host cmd` or `#!/bin/zsh` scripts.
- 02 SSH can surprise you.** `ssh host` is a login shell, but `ssh host cmd` is non-login & non-interactive — reads only `.zshenv` / `.bashrc`.
- 03 macOS `path_helper`** runs from `/etc/zprofile` and reshuffles `PATH`. Set `PATH` in `.zprofile` after it, not in `.zshenv`.
- 04 GUI apps & `launchd`** don't start a shell. VS Code, Finder, Spotlight-launched apps see neither `.zshenv` nor `.zprofile`.
- 05 Bash "first of" trap.** `.bash_profile` exists → `.profile` is ignored. Many Linux desktop sessions read `.profile` — surprise empty env.
- 06 `.bashrc` never auto-loads on macOS.** Terminal opens a login shell → sources `.bash_profile`. Source `.bashrc` from there if you want both.