

# Konfigurationsdateien der Bash

26. Februar 2019

LUG Frankfurt



## Ziele und Inhalt

### Einordnung/Historie

Definition, Geschichte  
Einordnung

### Konfigurationsdateien

Abhängigkeit von Shell und  
Distribution  
Übersicht  
Konfigurationsdateien  
Ausführungsreihenfolge  
Systemweite Konfdateien  
Benutzerspezifische  
Konfdateien

### Shelloptionen

Startoptionen  
set und shopt  
Tipps und Tricks

### Listings

### Zusammenfassung

### Literatur

Manuela Meurer

- 1 Konfigurationsdateien finden
- 2 Shellarten und Ausführungsreihenfolge kennen
- 3 Shelloptionen anpassen können



## Ziele und Inhalt

### Einordnung/Historie

Definition, Geschichte  
Einordnung

### Konfigurationsdateien

Abhängigkeit von Shell und  
Distribution  
Übersicht  
Konfigurationsdateien  
Ausführungsreihenfolge  
Systemweite Konfigdateien  
Benutzerspezifische  
Konfigdateien

### Shelloptionen

Startoptionen  
set und shopt  
Tipps und Tricks

### Listings

### Zusammenfassung

### Literatur

# Gliederung

Definition, Geschichte  
Einordnung

## 1 Konfigurationsdateien

Abhängigkeit von Shell und Distribution  
Übersicht Konfigurationsdateien  
Ausführungsreihenfolge  
Systemweite Konfdateien  
Benutzerspezifische Konfdateien

## 2 Shelloptionen

Startoptionen  
set und shopt  
Tipps und Tricks

## 3 Listings

## 4 Literatur



### Ziele und Inhalt

#### Einordnung/Historie

Definition, Geschichte  
Einordnung

#### Konfigurationsdateien

Abhängigkeit von Shell und  
Distribution  
Übersicht  
Konfigurationsdateien  
Ausführungsreihenfolge  
Systemweite Konfdateien  
Benutzerspezifische  
Konfdateien

#### Shelloptionen

Startoptionen  
set und shopt  
Tipps und Tricks

#### Listings

#### Zusammenfassung

#### Literatur

## Definition

- Freie Unix-Shell und Teil des GNU-Projekts; Standardshell für Linux-Systeme
- Ein Befehlsprozessor, der normalerweise in einem Textfenster/Terminal ausgeführt wird.
- Bash kann Befehle auch aus einer Datei, einem Shell-Skript, lesen und ausführen.

## Geschichte

- 1989 erstes Release; entwickelt von Brian Fox/ Chet Ramey
- 2014 Shellshock/Bashdoor: Verschiedene Bugs, die Denial of Service attacks ermöglichten.
- Aktuelle Version 5.0 (7. Januar 2019)



### Ziele und Inhalt

### Einordnung/Historie

#### Definition, Geschichte

Einordnung

### Konfigurationsdateien

Abhängigkeit von Shell und Distribution

Übersicht

Konfigurationsdateien

Ausführungsreihenfolge

Systemweite Konfdateien

Benutzerspezifische Konfdateien

### Shelloptionen

Startoptionen

set und shopt

Tipps und Tricks

### Listings

### Zusammenfassung

### Literatur

## bash

Die Schlüsselwörter, die Syntax und andere Grundmerkmale der Sprache sind alle von **sh** übernommen. Andere Merkmale, z.B. die History, sind von **cs**h und **ks**h übernommen. Bash ist eine POSIX-kompatible<sup>1</sup> Shell, aber mit einer Reihe von Erweiterungen.

## Die wichtigsten/meistgenutzten Shells

- sh - **Bourne Shell**: erstes Release 1979; entwickelt von Stephen Bourne
- ksh - **Korn Shell**: erstes Release 1983; entwickelt von David Korn
- csh - **C-Shell**: erstes Release 1978; entwickelt von Bill Joy

Weitere (weniger bekannte) Shells sind z.B.: tcsh, zsh, ash, dash, scsh, esh, fish, nash, sash, psh





## Abhängigkeit vom Shelltyp <sup>1</sup>

	sh	ksh	csk	tcsh	bash
/etc/.login			login	login	
/etc/csh.cshrc			immer	immer	
/etc/csh.login			login	login	
~/tcshrc				immer	
~/cshrc			immer	bedingt (1)	
~/etc/ksh.kshrc		interaktiv			
/etc/sh.shrc	interaktiv (2)				
\$ENV (typically ~/.kshrc)	interaktiv, bedingt (3,4)	interaktiv			interaktiv, bedingt(5)
~/login			login	login	
~/logout			login	login	
/etc/profile	login	login			login
~/profile	login	login			login
~/bash_profile					login(6)
~/bash_login					login(6)
~/bash_logout					login(6)
~/bashrc					interaktiv, nichtlogin

- (1) nur wenn ~/.tcshrc nicht gefunden wird
- (2) Nur neuere Versionen der Bourne-Shell
- (3) Verfügbar auf Systemen, die die Option "User Portability Utilities" unterstützen. Der Wert der Variablen muss ein absolutes Pfad sein und wird ignoriert, "wenn die tatsächlichen und effektiven Benutzer-IDs des Benutzers oder die realen und effektiven Gruppen-IDs unterschiedlich sind.
- (4) \$ENV ist in neueren Versionen der Bourne Shell \$HOME/.shrc
- (5) Gleiches Verhalten wie sh, jedoch nur bei Aufruf als sh (bash 2+) oder seit bash 4.2 auch explizit im POSIX-Kompatibilitätsmodus (mit Optionen --posix oder -o posix).
- (6) nur die erste lesbare Datei aus ~/.bash\_profile, ~/.bash\_login und ~/.profile wird ausgeführt; und nur ~/.profile bei Aufruf als sh oder ab mindestens Bash 4.2 bei explizitem Aufruf im POSIX-Kompatibilitätsmodus (mit Optionen --posix oder -o posix)

## Ziele und Inhalt

### Einordnung/Historie

Definition, Geschichte  
Einordnung

### Konfigurationsdateien

Abhängigkeit von Shell und Distribution

Übersicht  
Konfigurationsdateien  
Ausführungsreihenfolge  
Systemweite Konfigdateien  
Benutzerspezifische Konfigdateien

### Shelloptionen

Startoptionen  
set und shopt  
Tipps und Tricks

### Listings

### Zusammenfassung

### Literatur

## Abhängigkeit von Linux-Distribution

Konfigurationsdateien sind abhängig von der Distribution. Es gibt eine Grundstruktur, die in etwa gleich ist. Auf manchen Distributionen werden mehr Konfigurationsdateien ausgeführt. Der Ablageort der Konfigdateien kann distributionsabhängig variieren.

# Übersicht Konfigurationsdateien bash (debian)

## Systemweite Dateien

- /etc/profile
  - /etc/bash.bashrc
  - /etc/profile.d/\*.sh

## Benutzerspezifische Dateien

- \$HOME/.profile (oder \$HOME/.bash\_login oder \$HOME/.bash\_profile)
- \$HOME/.bashrc
- \$HOME/.bash\_logout

## Skeletons - enthält die Vorlagen, die bei Anlegen einer neuen Benutzerin in das HOME-Verzeichnis kopiert werden

- /etc/skel/.profile
- /etc/skel/.bashrc
- /etc/skel/.bash\_logout



### Ziele und Inhalt

#### Einordnung/Historie

Definition, Geschichte  
Einordnung

#### Konfigurationsdateien

Abhängigkeit von Shell und  
Distribution

#### Übersicht Konfigurationsdateien

Ausführungsreihenfolge  
Systemweite Konfdateien  
Benutzerspezifische  
Konfdateien

#### Shelloptionen

Startoptionen  
set und shopt  
Tipps und Tricks

#### Listings

#### Zusammenfassung

#### Literatur

# Ausführungsreihenfolge (bash debian)



Die Ausführung der Konfigurationsdateien ist abhängig davon, ob es sich um eine login- oder nicht-login-Shell handelt und ob es sich um eine interaktive oder nicht interaktive Shell handelt.

- Eine Login-Shell ist eine Shell, deren erstes Zeichen des 0ten Arguments '-' ist (-bash) oder die mit der Option -l gestartet wird.
- Eine interaktive Shell ist eine Shell, die ohne non-option Argumente (Ausnahme -s, diese lässt Positionsparameter zu und ist trotzdem interaktiv) und ohne -c Option gestartet wird oder mit der Option -i

interaktiv	Login	/etc/profile \$HOME/.profile	/etc/bash.bashrc \$HOME/.bashrc	/etc/profile.d/*.sh \$HOME/.bash_logout
nicht interaktiv	Login	/etc/profile \$HOME/.profile	\$BASH_ENV	/etc/profile.d/*.sh \$HOME/.bash_logout
interaktiv	nicht-Login		/etc/bash.bashrc \$HOME/.bashrc	
nicht interaktiv	nicht-Login		\$BASH_ENV	

## Ziele und Inhalt

### Einordnung/Historie

Definition, Geschichte  
Einordnung

### Konfigurationsdateien

Abhängigkeit von Shell und  
Distribution  
Übersicht  
Konfigurationsdateien

### Ausführungsreihenfolge

Systemweite Konfdateien  
Benutzerspezifische  
Konfdateien

### Shelloptionen

Startoptionen  
set und shopt  
Tipps und Tricks

### Listings

### Zusammenfassung

### Literatur



## Beispiele

- interaktive Login Shell
  - `ssh user@remote_server`
  - `su -`
  - Aufruf eines Login-Terminals mit z.B. `Ctrl-Alt-F3`
- nicht-interaktive Login Shell
  - `ssh user@remote_server < localfile`
  - `bash -l -c 'shopt login_shell; echo $-'`
- interaktive Nicht-Login Shell
  - neues Terminal öffnen mit `xterm &`
- nicht-interaktive Nicht-Login Shell
  - Aufruf eines Shellskripts
  - `ssh user@remote_server ./remotefile`

### Ziele und Inhalt

#### Einordnung/Historie

Definition, Geschichte  
Einordnung

#### Konfigurationsdateien

Abhängigkeit von Shell und  
Distribution  
Übersicht  
Konfigurationsdateien

#### Ausführungsreihenfolge

Systemweite Konfigdateien  
Benutzerspezifische  
Konfigdateien

#### Shelloptionen

Startoptionen  
`set` und `shopt`  
Tipps und Tricks

#### Listings

#### Zusammenfassung

#### Literatur

# Ausführungsreihenfolge Sonderfälle (bash debian)

## Sonderfälle, die eine andere Reihenfolge des Einlesens der Startupdateien haben

- **Start mit sh:** Die Shell übernimmt das Verhalten der historischen sh und ist gleichzeitig posix-kompatibel. Als Login-Shell gestartet, liest sie die beiden profiles, als interaktive Shell liest sie die Datei, die in der Variable ENV referenziert wird. Es werden keine anderen Startup-Dateien eingelesen. Nach dem Lesen der beiden profiles geht die Shell in den Posix-Modus.
- Start im **Posix-Modus:** Die Shell liest die Datei, die in der Variable ENV referenziert wird. Es werden keine anderen Startup-Dateien eingelesen.
- **Option --norc:** Das Einlesen der beiden bashrc Dateien für interaktive Shells wird verhindert.
- **Option --rcfile <file>:** Anstatt der beiden bashrc Dateien wird für interaktive Shells die Datei <file> eingelesen.
- **Option --noprofile:** Das Einlesen der beiden profile Dateien für Login Shells wird verhindert.



### Ziele und Inhalt

#### Einordnung/Historie

Definition, Geschichte  
Einordnung

#### Konfigurationsdateien

Abhängigkeit von Shell und  
Distribution  
Übersicht  
Konfigurationsdateien

#### Ausführungsreihenfolge

Systemweite Konfdateien  
Benutzerspezifische  
Konfdateien

#### Shelloptionen

Startoptionen  
set und shopt  
Tipps und Tricks

#### Listings

#### Zusammenfassung

#### Literatur



## **/etc/profile** - system-wide .profile file for the Bourne shell (sh(1)) and Bourne compatible shells (bash(1), ksh(1), ash(1), ...).

- setzt abhängig von der Userid die Umgebungsvariable **PATH** (root erhält einen anderen Pfad als ein normaler User)
- führt **/etc/bash.bashrc** aus, falls es sich um eine interaktive Shell handelt (PS1 ist gesetzt und nicht leer). Ausserdem muss die Variable BASH gesetzt und nicht leer sein und darf nicht auf /bin/sh gesetzt sein.
- führt alle sh-Dateien in **/etc/profile.d** aus, falls diese lesbar sind.
- umask wird nicht (mehr) in /etc/profile gesetzt, sondern in /etc/login.defs oder in /etc/pam.d/login

### Ziele und Inhalt

#### Einordnung/Historie

Definition, Geschichte  
Einordnung

#### Konfigurationsdateien

Abhängigkeit von Shell und  
Distribution

Übersicht  
Konfigurationsdateien

Ausführungsreihenfolge

#### Systemweite Konfigdateien

Benutzerspezifische  
Konfigdateien

#### Shelloptionen

Startoptionen  
set und shopt  
Tipps und Tricks

#### Listings

#### Zusammenfassung

#### Literatur

## System-wide .bashrc file for interactive bash(1) shells.

- Wenn die Shell nicht interaktiv ist, wird das Skript sofort verlassen.
- Die Shell-Option **checkwinsize** wird gesetzt. D.h. Bash prüft die Fenstergröße nach jedem Befehl und passt die Werte der Spalten und Reihen an, wenn notwendig (Anzeige mit `stty -a`).
- Wenn die Datei **/etc/debian\_chroot** existiert, wird der Prompt `PS1` auf den Inhalt dieser Datei gesetzt, wenn nicht wird er auf Username, Hostname und aktuelles Arbeitsverzeichnis gesetzt. (chroot ist ein Programm zum Ändern des Rootverzeichnisses - Sandbox)
- Falls, dass Package **command-not-found** installiert ist, wird es gestartet. `command-not-found` schlägt in interaktiven Bash-Sitzungen Paketinstallationen vor.

LUG Frankfurt



Ziele und Inhalt

Einordnung/Historie

Definition, Geschichte  
Einordnung

Konfigurationsdateien

Abhängigkeit von Shell und  
Distribution  
Übersicht  
Konfigurationsdateien  
Ausführungsreihenfolge

Systemweite Konfigdateien

Benutzerspezifische  
Konfigdateien

Shelloptionen

Startoptionen  
set und shopt  
Tipps und Tricks

Listings

Zusammenfassung

Literatur

/etc/profile.d enthält bei debian 2 Dateien:

### [/etc/profile.d/bash\\_completion.sh](#)

- `bash_completion` erweitert die Unterstützung der `bash` zur automatischen Vervollständigung von Dateinamen, Befehlsnamen und Variablen um eine grosse Anzahl von Befehlen und ist programmierbar.
- `bash_completion.sh`: ruft `/usr/share/bash-completion/bash_completion` auf für interaktive Shells, falls es nicht schon ausgeführt wurde.

### [vte-2.91.sh](#)

- `vte-2.91.sh` setzt für bestimmte Anwendungen, falls diese installiert sind, den Prompt neu (z.B. Tilix).
- wird nur für interaktive Shells ausgeführt



#### Ziele und Inhalt

#### [Einordnung/Historie](#)

Definition, Geschichte  
Einordnung

#### [Konfigurationsdateien](#)

Abhängigkeit von Shell und  
Distribution

Übersicht  
Konfigurationsdateien

Ausführungsreihenfolge

#### [Systemweite Konfigdateien](#)

Benutzerspezifische  
Konfigdateien

#### [Shelloptionen](#)

Startoptionen  
set und shopt  
Tipps und Tricks

#### [Listings](#)

#### [Zusammenfassung](#)

#### [Literatur](#)



**\$HOME/.profile**: executed by the command interpreter for login shells. This file is not read by bash(1), if \$HOME/.bash\_profile or \$HOME/.bash\_login exists.

- führt **\$HOME/.bashrc** aus, wenn die Länge des Inhalts der Shellvariablen \$BASH\_VERSION nicht 0 ist.
- setzt den Pfad auf: **PATH="\$HOME/bin:\$PATH"**, falls das Verzeichnis \$HOME/bin existiert.

## Ziele und Inhalt

### Einordnung/Historie

Definition, Geschichte  
Einordnung

### Konfigurationsdateien

Abhängigkeit von Shell und  
Distribution  
Übersicht  
Konfigurationsdateien  
Ausführungsreihenfolge  
Systemweite Konfdateien

### Benutzerspezifische Konfdateien

### Shelloptionen

Startoptionen  
set und shopt  
Tipps und Tricks

### Listings

### Zusammenfassung

### Literatur



## \$HOME/.bashrc: executed by bash(1) for non-login shells.

- Wenn die Shell nicht interaktiv ist, wird das Skript sofort verlassen.
- Es werden einige Einstellungen für die **history** gesetzt sowie die Shelloption `checkwinsize`
- Verschiedene Einstellungen für den **Prompt** werden gesetzt (PS1)
- **ls** wird auf eine farbige Ausgabe gesetzt
- Die Datei `$HOME/.bash_aliases` wird ausgeführt, falls sie existiert.
- Die Datei `bash_completion` wird ausgeführt, falls sie in einem der abgefragten Verzeichnisse existiert und die Shelloption `posix` nicht gesetzt ist.

### Ziele und Inhalt

#### Einordnung/Historie

Definition, Geschichte  
Einordnung

#### Konfigurationsdateien

Abhängigkeit von Shell und  
Distribution  
Übersicht  
Konfigurationsdateien  
Ausführungsreihenfolge  
Systemweite Konfdateien

#### Benutzerspezifische Konfdateien

#### Shelloptionen

Startoptionen  
set und shopt  
Tipps und Tricks

#### Listings

#### Zusammenfassung

#### Literatur



## Einige Startoptionen

Die Liste ist nicht vollständig, mehr Startoptionen in den Manpages `bash`.

- `-c <Kommando> #` Ausführen von Kommando
- `-i #` interaktiv starten
- `-l #` als Login-Shell starten
- `-r #` im eingeschränkten Modus starten
- `-s #` Shell-Kommandos von der Standard-Eingabe lesen, Setzen von Positionsparametern
- `+O/-O [shopt_option]` Alle Optionen des Shell-builtin `shopt` werden akzeptiert.

Tipp: Mit `man -t bash | ps2pdf - bash.pdf` kann die Manpage zum angenehmeren Lesen ins PDF-Format umgewandelt werden.

### Ziele und Inhalt

#### Einordnung/Historie

Definition, Geschichte  
Einordnung

#### Konfigurationsdateien

Abhängigkeit von Shell und  
Distribution  
Übersicht  
Konfigurationsdateien  
Ausführungsreihenfolge  
Systemweite Konfigdateien  
Benutzerspezifische  
Konfigdateien

#### Shelloptionen

##### Startoptionen

set und shopt  
Tipps und Tricks

#### Listings

#### Zusammenfassung

#### Literatur

## set und shopt

- set und shopt sind beides Shell-Builtins
- Set enthält 27 Optionen (anzeigen mit `set -o`), shopt ist eine Erweiterung für die Bash und enthält 48 Optionen (anzeigen mit `shopt`).
- Die aktivierten Optionen von set befinden sich in der Shellvariablen SHELLOPTS, die aktivierten Optionen von shopt in der Shellvariablen BASHOPTS.
- Aktivieren der Shelloptionen mit `set -o <Optionsname>` oder `set -<Optionsbuchstabe>`  
Deaktivieren mit `set +o <Optionsname>` oder `set +<Optionsbuchstabe>`
- Aktivieren der Bashoptionen mit `shopt -s <Optionsname>`  
Deaktivieren mit `shopt -u <Optionsname>`
- Mit `shopt -o` werden nur Optionen angezeigt, die `set -o` anzeigt.



### Ziele und Inhalt

#### Einordnung/Historie

Definition, Geschichte  
Einordnung

#### Konfigurationsdateien

Abhängigkeit von Shell und  
Distribution  
Übersicht  
Konfigurationsdateien  
Ausführungsreihenfolge  
Systemweite Konfdateien  
Benutzerspezifische  
Konfdateien

#### Shelloptionen

Startoptionen

**set und shopt**

Tipps und Tricks

#### Listings

#### Zusammenfassung

#### Literatur

## Beispiele

- `set -x (set -o xtrace) #Befehlsenerweiterung vor Ausführung`
- `set -C (set -o noclobber) # Überschreiben von Dateien durch Umleitung verhindern`
- `shopt -s expand_aliases # Ermöglicht Alias-Erweiterungen`
- `shopt -s checkwinsize # Nach jedem Befehl wird die Fenstergröße geprüft und angepasst`
- `shopt -s cmdhist # Mehrzeilige Befehle in einer Historyzeile speichern`
- `shopt -s globstar # Das Muster '**' bezieht sich auf alle Verzeichnisse, Unterverzeichnisse und Dateien`
- `shopt -s huponexit # SIGHUP wird an alle Jobs gesendet, wenn eine interaktive login Shell beendet wird.`
- `set # Ohne Argumente gibt set alle Shell-Variablen aus.`



### Ziele und Inhalt

#### Einordnung/Historie

Definition, Geschichte  
Einordnung

#### Konfigurationsdateien

Abhängigkeit von Shell und  
Distribution  
Übersicht  
Konfigurationsdateien  
Ausführungsreihenfolge  
Systemweite Konfigdateien  
Benutzerspezifische  
Konfigdateien

#### Shelloptionen

Startoptionen

**set und shopt**

Tipps und Tricks

#### Listings

#### Zusammenfassung

#### Literatur



- Pfade sollten in `$HOME/.profile` gesetzt werden. Wenn sie in `$HOME/.bashrc` gesetzt werden, verlängern sie sich bei jedem Starten einer interaktiven Shell.
- Ein häufiger Fehler ist der fehlende export der PATH Variable im `.profile`
- HISTSIZE und HISTFILESIZE in `$HOME/.bashrc` erhöhen, default ist 500.
- mit `bash -l -x` kann nachvollzogen werden, welche Dateien bei einer interaktiven Login-Shell eingelesen werden, und welche Kontrollstrukturen durchlaufen werden.
- Oft genutzte Befehle als alias definieren in der Datei `$HOME/.bash_aliases`. Diese wird von `.bashrc` eingelesen.

### Ziele und Inhalt

#### Einordnung/Historie

Definition, Geschichte  
Einordnung

#### Konfigurationsdateien

Abhängigkeit von Shell und Distribution  
Übersicht  
Konfigurationsdateien  
Ausführungsreihenfolge  
Systemweite Konfdateien  
Benutzerspezifische Konfdateien

#### Shelloptionen

Startoptionen  
set und shopt

#### Tipps und Tricks

#### Listings

#### Zusammenfassung

#### Literatur

# /etc/profile

```
1 # /etc/profile: system-wide .profile file for the Bourne shell (sh(1))
2 # and Bourne compatible shells (bash(1), ksh(1), ash(1), ...).
3
4 if [ "`id -u`" -eq 0 ]; then
5     PATH="/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin"
6 else
7     PATH="/usr/local/bin:/usr/bin:/bin:/usr/local/games:/usr/games"
8 fi
9 export PATH
10
11 if [ "${PS1-}" ]; then
12     if [ "${BASH-}" ] && [ "$BASH" != "/bin/sh" ]; then
13         # The file bash.bashrc already sets the default PS1.
14         # PS1='\h:\w\$ '
15         if [ -f /etc/bash.bashrc ]; then
16             . /etc/bash.bashrc
17         fi
18     else
19         if [ "`id -u`" -eq 0 ]; then
20             PS1='# '
21         else
22             PS1='$ '
23         fi
24     fi
25 fi
26
27 if [ -d /etc/profile.d ]; then
28     for i in /etc/profile.d/*.sh; do
29         if [ -r $i ]; then
30             . $i
31         fi
32     done
33     unset i
34 fi
```



## Ziele und Inhalt

### Einordnung/Historie

- Definition, Geschichte
- Einordnung

### Konfigurationsdateien

- Abhängigkeit von Shell und Distribution
- Übersicht
- Konfigurationsdateien
- Ausführungsreihenfolge
- Systemweite Konfdateien
- Benutzerspezifische Konfdateien

### Shelloptionen

- Startoptionen
- set und shopt
- Tipps und Tricks

### Listings

### Zusammenfassung

### Literatur

# /etc/bash.bashrc

```
1 # System-wide .bashrc file for interactive bash(1) shells.
2
3 # To enable the settings / commands in this file for login shells as well,
4 # this file has to be sourced in /etc/profile.
5
6 # If not running interactively, don't do anything
7 [ -z "$PS1" ] && return
8
9 # check the window size after each command and, if necessary,
10 # update the values of LINES and COLUMNS.
11 shopt -s checkwinsize
12
13 # set variable identifying the chroot you work in (used in the prompt below)
14 if [ -z "${debian_chroot:-}" ] && [ -r /etc/debian_chroot ]; then
15     debian_chroot=$(cat /etc/debian_chroot)
16 fi
17
18 # set a fancy prompt (non-color, overwrite the one in /etc/profile)
19 PS1='${debian_chroot:+($debian_chroot)}\u@\h:\w\$ '
20
21 # Commented out, don't overwrite xterm -T "title" -n "icontitle" by default.
22 # If this is an xterm set the title to user@host:dir
23 #case "$TERM" in
24 #xterm*|rxvt*)
25 #    PROMPT_COMMAND='echo -ne "\033]0;${USER}@${HOSTNAME}: ${PWD}\007"'
26 #    ;;
27 #*)
28 #    ;;
29 #esac
30
31 # enable bash completion in interactive shells
32 #if ! shopt -oq posix; then
33 #    if [ -f /usr/share/bash-completion/bash_completion ]; then
34 #        . /usr/share/bash-completion/bash_completion
35 #    elif [ -f /etc/bash_completion ]; then
36 #        . /etc/bash_completion
37 #    fi
38 #fi
39
40 # if the command-not-found package is installed, use it
41 if [ -x /usr/lib/command-not-found -o -x
42     /usr/share/command-not-found/command-not-found ]; then
43     function command_not_found_handle {
44         # check because c-n-f could've been removed in the meantime
45         if [ -x /usr/lib/command-not-found ]; then
46             /usr/lib/command-not-found -- "$1"
47             return $?
48         elif [ -x /usr/share/command-not-found/command-not-found ]; then
49             /usr/share/command-not-found/command-not-found -- "$1"
50             return $?
51         else
52             printf "%s: command not found\n" "$1" >&2
53             return 127
54         fi
55     }
56 fi
```



### Ziele und Inhalt

### Einordnung/Historie

Definition, Geschichte  
Einordnung

### Konfigurationsdateien

Abhängigkeit von Shell und  
Distribution  
Übersicht  
Konfigurationsdateien  
Ausführungsreihenfolge  
Systemweite Konfdateien  
Benutzerspezifische  
Konfdateien

### Shelloptionen

Startoptionen  
set und shopt  
Tipps und Tricks

### Listings

### Zusammenfassung

### Literatur

# /etc/profile.d/bash\_completion.sh

```
1 # Check for interactive bash and that we haven't already been sourced.
2 if [ -n "$BASH_VERSION" -a -n "$PS1" -a -z "$BASH_COMPLETION_COMPAT_DIR" ]; then
3
4 # Check for recent enough version of bash.
5 bash=${BASH_VERSION%.*}; bmajor=${bash%.*}; bminor=${bash#*.*}
6 if [ $bmajor -gt 4 ] || [ $bmajor -eq 4 -a $bminor -ge 1 ]; then
7   [ -r "${XDG_CONFIG_HOME:-$HOME/.config}/bash_completion" ] && \
8     . "${XDG_CONFIG_HOME:-$HOME/.config}/bash_completion"
9   if shopt -q progcomp && [ -r /usr/share/bash-completion/bash_completion ];
10  then
11    # Source completion code.
12    . /usr/share/bash-completion/bash_completion
13  fi
14 unset bash bmajor bminor
15
16 fi
```



## Ziele und Inhalt

### Einordnung/Historie

- Definition, Geschichte
- Einordnung

### Konfigurationsdateien

- Abhängigkeit von Shell und Distribution
- Übersicht
- Konfigurationsdateien
- Ausführungsreihenfolge
- Systemweite Konfdateien
- Benutzerspezifische Konfdateien

### Shelloptionen

- Startoptionen
- set und shopt
- Tipps und Tricks

### Listings

### Zusammenfassung

### Literatur

# /etc/profile.d/vte-2.91

```
1 # Copyright © 2006 Shaun McCance <shaunm@gnome.org>
2 # Copyright © 2013 Peter De Wachter <pdewacht@gmail.com>
3 #
4 # This program is free software: you can redistribute it and/or modify
5 # it under the terms of the GNU General Public License as published by
6 # the Free Software Foundation, either version 3 of the License, or
7 # (at your option) any later version.
8 #
9 # This program is distributed in the hope that it will be useful,
10 # but WITHOUT ANY WARRANTY; without even the implied warranty of
11 # MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
12 # GNU General Public License for more details.
13 #
14 # You should have received a copy of the GNU General Public License
15 # along with this program. If not, see <http://www.gnu.org/licenses/>.
16
17 # Not bash or zsh?
18 [ -n "$BASH_VERSION" -o -n "$ZSH_VERSION" ] || return 0
19
20 # Not an interactive shell?
21 [[ $- == *i* ]] || return 0
22
23 # Not running under vte?
24 [ "${VTE_VERSION:-0}" -ge 3405 ] || return 0
25
26 __vte_urlencode() {
27 # This is important to make sure string manipulation is handled
28 # byte-by-byte.
29 LC_ALL=C
30 str="$1"
31 while [ -n "$str" ]; do
32 safe="${str%[!a-zA-Z0-9/!_\.\\-!\\(\)-]*}"
33 printf "%s" "$safe"
34 str="${str#$safe}"
35 if [ -n "$str" ]; then
36 printf "%02X" "'$str"
37 str="${str#?}"
38 fi
39 done
40 }
41
42 # Print a warning so that anyone who's added this manually to his PS1 can adapt.
43 # The function will be removed in a later version.
44 __vte_psl() {
45 echo -n "(__vte_psl is obsolete)"
46 }
47
48 __vte_osc7 () {
49 printf "\033]7;file://%s%s\007" "${HOSTNAME:-}" "${__vte_urlencode "${PWD}"}"
50 }
51
52 __vte_prompt_command() {
53 local pwd="-"
54 [ "$PWD" != "$HOME" ] && pwd=${PWD/#$HOME/\~/}
55 printf "\033]0;%s%s:\007%s" "${USER}" "${HOSTNAME%%.*}" "${pwd}"
56 "${__vte_osc7}"
57 }
58 case "$TERM" in
59 xterm*vte*)
60 [ -n "$BASH_VERSION" ] && PROMPT_COMMAND="__vte_prompt_command"
61 [ -n "$ZSH_VERSION" ] && precmd_functions+=(__vte_osc7)
62 ;;
63 esac
64
65 true
```



### Ziele und Inhalt

### Einordnung/Historie

Definition, Geschichte  
Einordnung

### Konfigurationsdateien

Abhängigkeit von Shell und  
Distribution  
Übersicht  
Konfigurationsdateien  
Ausführungsreihenfolge  
Systemweite Konfdateien  
Benutzerspezifische  
Konfdateien

### Shelloptionen

Startoptionen  
set und shopt  
Tipps und Tricks

### Listings

### Zusammenfassung

### Literatur

# \$HOME/.profile

```
1 # ~/.profile: executed by the command interpreter for login shells.
2 # This file is not read by bash(1), if ~/.bash_profile or ~/.bash_login
3 # exists.
4 # see /usr/share/doc/bash/examples/startup-files for examples.
5 # the files are located in the bash-doc package.
6
7 # the default umask is set in /etc/profile; for setting the umask
8 # for ssh logins, install and configure the libpam-umask package.
9 #umask 022
10
11 # if running bash
12 if [ -n "$BASH_VERSION" ]; then
13     # include .bashrc if it exists
14     if [ -f "$HOME/.bashrc" ]; then
15         . "$HOME/.bashrc"
16     fi
17 fi
18
19 # set PATH so it includes user's private bin if it exists
20 if [ -d "$HOME/bin" ]; then
21     PATH="$HOME/bin:$PATH"
22 fi
```



### Ziele und Inhalt

#### Einordnung/Historie

Definition, Geschichte  
Einordnung

#### Konfigurationsdateien

Abhängigkeit von Shell und  
Distribution  
Übersicht  
Konfigurationsdateien  
Ausführungsreihenfolge  
Systemweite Konfigdateien  
Benutzerspezifische  
Konfigdateien

#### Shelloptionen

Startoptionen  
set und shopt  
Tipps und Tricks

#### Listings

#### Zusammenfassung

#### Literatur

# \$HOME/.bashrc

```
1 # ~/.bashrc: executed by bash(1) for non-login shells.
2 # see /usr/share/doc/bash/examples/startup-files (in the package bash-doc)
3 # for examples
4
5 # If not running interactively, don't do anything
6 case $- in
7     *i*) ;;
8     *) return;;
9 esac
10
11 # don't put duplicate lines or lines starting with space in the history.
12 # See bash(1) for more options
13 HISTCONTROL=ignoreboth
14
15 # append to the history file, don't overwrite it
16 shopt -s histappend
17
18 # for setting history length see HISTSIZE and HISTFILESIZE in bash(1)
19 HISTSIZE=1000
20 HISTFILESIZE=2000
21
22 # check the window size after each command and, if necessary,
23 # update the values of LINES and COLUMNS.
24 shopt -s checkwinsize
25
26 # If set, the pattern "*" used in a pathname expansion context will
27 # match all files and zero or more directories and subdirectories.
28 #shopt -s globstar
29
30 # make less more friendly for non-text input files, see lesspipe(1)
31 #[ -x /usr/bin/lesspipe ] && eval "$(SHELL=/bin/sh lesspipe)"
32
33 # set variable identifying the chroot you work in (used in the prompt below)
34 if [ -z "${debian_chroot:-}" ] && [ -r /etc/debian_chroot ]; then
35     debian_chroot=$(cat /etc/debian_chroot)
36 fi
37
38 # set a fancy prompt (non-color, unless we know we "want" color)
39 case "$TERM" in
40     xterm-color|*-256color) color_prompt=yes;;
41 esac
42
43 # uncomment for a colored prompt, if the terminal has the capability; turned
44 # off by default to not distract the user: the focus in a terminal window
45 # should be on the output of commands, not on the prompt
46 #force_color_prompt=yes
47
48 if [ -n "$force_color_prompt" ]; then
49     if [ -x /usr/bin/tput ] && tput setaf 1 >&/dev/null; then
50         # We have color support; assume it's compliant with Ecma-48
51         # (ISO/IEC-6429). (Lack of such support is extremely rare, and such
52         # a case would tend to support setf rather than setaf.)
53         color_prompt=yes
54     else
55         color_prompt=
56     fi
57 fi
```

## Konfigurationsdateien der Bash

Manuela Meurer

LUG Frankfurt



### Ziele und Inhalt

#### Einordnung/Historie

- Definition, Geschichte
- Einordnung

#### Konfigurationsdateien

- Abhängigkeit von Shell und Distribution
- Übersicht
- Konfigurationsdateien
- Ausführungsreihenfolge
- Systemweite Konfdateien
- Benutzerspezifische Konfdateien

#### Shelloptionen

- Startoptionen
- set und shopt
- Tipps und Tricks

#### Listings

#### Zusammenfassung

#### Literatur

# \$HOME/.bashrc Teil 2

```
59 if [ "$color_prompt" = yes ]; then
60     PS1='${debian_chroot:+($debian_chroot)}\[\033[01;32m\]\u@\h\[\033[00m\]:\[\033[01
;34m\]\w\[\033[00m\]\$ '
61 else
62     PS1='${debian_chroot:+($debian_chroot)}\u@\h:\w\$ '
63 fi
64 unset color_prompt force_color_prompt
65
66 # If this is an xterm set the title to user@host:dir
67 case "$TERM" in
68 xterm*|rxvt*)
69     PS1="\[\e]0;${debian_chroot:+($debian_chroot)}\u@\h: \w\a\]$PS1"
70     ;;
71 *)
72     ;;
73 esac
74
75 # enable color support of ls and also add handy aliases
76 if [ -x /usr/bin/dircolors ]; then
77     test -r ~/.dircolors && eval "$(dircolors -b ~/.dircolors)" || eval
"$("dircolors -b)"
78     alias ls='ls --color=auto'
79     #alias dir='dir --color=auto'
80     #alias vdir='vdir --color=auto'
81
82     #alias grep='grep --color=auto'
83     #alias fgrep='fgrep --color=auto'
84     #alias egrep='egrep --color=auto'
85 fi
86
87 # colored GCC warnings and errors
88 #export
GCC_COLORS='error=01;31:warning=01;35:note=01;36:caret=01;32:locus=01:quote=01'
89
90 # some more ls aliases
91 #alias ll='ls -l'
92 #alias la='ls -A'
93 #alias l='ls -CF'
94
95 # Alias definitions.
96 # You may want to put all your additions into a separate file like
97 # ~/.bash_aliases, instead of adding them here directly.
98 # See /usr/share/doc/bash-doc/examples in the bash-doc package.
99
100 if [ -f ~/.bash_aliases ]; then
101     . ~/.bash_aliases
102 fi
103
104 # enable programmable completion features (you don't need to enable
105 # this, if it's already enabled in /etc/bash.bashrc and /etc/profile
106 # sources /etc/bash.bashrc).
107 if ! shopt -oq posix; then
108     if [ -f /usr/share/bash-completion/bash_completion ]; then
109         . /usr/share/bash-completion/bash_completion
110     elif [ -f /etc/bash_completion ]; then
111         . /etc/bash_completion
112     fi
113 fi
```



### Ziele und Inhalt

### Einordnung/Historie

- Definition, Geschichte
- Einordnung

### Konfigurationsdateien

- Abhängigkeit von Shell und Distribution
- Übersicht Konfigurationsdateien
- Ausführungsreihenfolge
- Systemweite Konfdateien
- Benutzerspezifische Konfdateien

### Shelloptionen

- Startoptionen
- set und shopt
- Tipps und Tricks

### Listings

### Zusammenfassung

### Literatur

# \$HOME/.bash\_logout

```
1 # ~/.bash_logout: executed by bash(1) when login shell exits.
2
3 # when leaving the console clear the screen to increase privacy
4
5 if [ "$SHLVL" = 1 ]; then
6     [ -x /usr/bin/clear_console ] && /usr/bin/clear_console -q
7 fi
```



## Ziele und Inhalt

### Einordnung/Historie

Definition, Geschichte  
Einordnung

### Konfigurationsdateien

Abhängigkeit von Shell und  
Distribution  
Übersicht  
Konfigurationsdateien  
Ausführungsreihenfolge  
Systemweite Konfdateien  
Benutzerspezifische  
Konfdateien

### Shelloptionen

Startoptionen  
set und shopt  
Tipps und Tricks

### Listings

### Zusammenfassung

### Literatur



- 1 Welche **Konfigurationsdateien** in der bash ausgeführt werden, ist abhängig davon, ob die Shell eine interaktive, nicht-interaktive, eine Login- oder eine nicht-Login Shell ist.
- 2 Das Verhalten der Bash kann zusätzlich durch **Shelloptionen** bestimmt werden.

## Ziele und Inhalt

### Einordnung/Historie

Definition, Geschichte  
Einordnung

### Konfigurationsdateien

Abhängigkeit von Shell und  
Distribution  
Übersicht  
Konfigurationsdateien  
Ausführungsreihenfolge  
Systemweite Konfdateien  
Benutzerspezifische  
Konfdateien

### Shelloptionen

Startoptionen  
set und shopt  
Tipps und Tricks

### Listings

## Zusammenfassung

### Literatur



- 1 **POSIX.1-2017, Shell & Utilities: [opengroup.org](http://pubs.opengroup.org/onlinepubs/9699919799/)** `http://pubs.opengroup.org/onlinepubs/9699919799/`
- 2 **Die GNU-Shell Bash (Bourne Again SHell)**  
`https://www-user.tu-chemnitz.de/~hot/unix_linux_werkzeugkasten/bash.html`
- 3 **Wikipedia, Unix shell,**  
`https://en.wikipedia.org/wiki/Unix_shell`
- 4 **man pages**

### Ziele und Inhalt

#### Einordnung/Historie

Definition, Geschichte  
Einordnung

#### Konfigurationsdateien

Abhängigkeit von Shell und  
Distribution  
Übersicht  
Konfigurationsdateien  
Ausführungsreihenfolge  
Systemweite Konfdateien  
Benutzerspezifische  
Konfdateien

#### Shelloptionen

Startoptionen  
set und shopt  
Tipps und Tricks

#### Listings

#### Zusammenfassung

#### Literatur