

Beamer Testpage / Testseite

Videoeinstellungen?

Auflösung?

Handhabung für das Umblättern?

Schriftgröße normalsize

Schriftgröße large

Schriftgröße Large

(Large = Standard)

Schriftgröße LARGE

Schriftgröße Huge

# Debian GNU/Linux Bootstrap

DGL Bootstrap

Debian 'mal anders installiert

Frank Guthausen

FSFE Fellowship Vortrag

Düsseldorf 2016-02-24

# Vorgeschichte

long time ago, on a computer far away...

- CDROM Installer: default, expert, GUI?
- tasksel (coarse grain)
- dselect
  - unfinished business
  - tickende Zeitbombe
- Reproduzierbarkeit?

## Vorgeschichte II

- ca. 20xy: bare metal
- ca. 20xy: bootfähige USB-Sticks
- ca. 2005: Virtualisierung Xen
- ca. 20xy: Virtualisierung KVM
- ca. 20xy: chroot, LXC

# Chinesische Weisheit

Gib einem Mann einen Fisch und du ernährst ihn für einen Tag. Lehre einen Mann zu fischen und du ernährst ihn für sein Leben.

# Genotyp vs. Phänotyp I

- Phänotyp: fertiges Produkt
  - Lebewesen
  - Mahlzeit
  - Binaries, deb-Pakete, Templates
  - Ergebnis debootstrap

# Genotyp vs. Phänotyp II

- Genotyp: initiale Daten und Algorithmus
  - DNA
  - Rezept
  - Quellcode, Konfiguration, Skripte
  - Methode debootstrap

# Genotyp vs. Phänotyp III

- J. W. v. Goethe, Faust II: Homunculus
  - natura naturans vs. natura naturata
- Aussagen und Metaaussagen
- Index und Wert
- P2G: Daten als Code
- Rückkopplung:
  - selbstmodifizierender Code
- K. Gödel:
  - selbstreferenzierende Aussagen



# Was will man?

- universelle Methode
- kleine Schritte: stages
- wahlweise genotypisch oder phänotypisch
- Template-Sammlung der stages
- Individualisierung (Corporate Identity)
- vordefinierte Benutzer
- vorkonfigurierte Umgebung
- nach der Installation sofort loslegen

# Verankerung I

- leerer Rechner (auch bei KVM)
  - Hilfssystem klassisch via CDROM
  - klassischer Installer: nur Hilfssystem
- LiveCD/-DVD (Knoppix) booten
- von USB-Medium booten
- existierendes System booten

# Verankerung II

- Ziel bedarfsweise partitionieren
  - 3x 256 MiB (/boot)
  - Partition 4 extended (MBR) oder
  - Partition 4 bios (GPT)
  - mindestens 2x Systempartition:  
Updates als Neuinstallation
  - jeweils ca. 32GiB (bare metal)  
bedarfsweise (RAM, suspend)

# Skelett: debootstrap

- Dateisysteme anlegen
- mounten
- virtuelle HDD: kpartx

```
# time debootstrap --arch amd64 \\  
jessie /mnt http://ftp.debian.org/debian
```

3-12 min.

## (pre)chroot

```
# mount -o bind /dev /mnt/dev
```

```
# mount -o bind /dev/pts /mnt/dev/pts
```

```
# mount -o bind /proc /mnt/proc
```

```
# mount -o bind /sys /mnt/sys
```

```
# chroot /mnt
```

# chroot

- nur einfache sources.list
- kein aptitude
- keine locales, keine timezone
- kein Keyboard, kein Font
- kein Editor (vim, emacs)
  
- es gibt fast gar nichts
  - außer unpraktischen defaults

## chroot II

```
# echo 'APT::Cache-Limit "67108864";' >>  
/etc/apt/apt.conf
```

```
# echo 'APT::Install-Recommends "0";' >>  
/etc/apt/apt.conf
```

```
# echo 'APT::Install-Suggests "0";' >>  
/etc/apt/apt.conf
```

```
# echo 'Aptitude::Recommends-Important  
false;' >> /etc/apt/apt.conf
```

## chroot III

```
# apt-get -y install aptitude
```

```
# aptitude update
```

```
# aptitude safe-upgrade
```

```
# aptitude install locales
```

```
# dpkg-reconfigure locales
```

```
# dpkg-reconfigure debconf
```

```
# dpkg-reconfigure tzdata
```



# Intermezzo

Gähnen!

Geht das jetzt so weiter?

Es wird anstrengend und langweilig.

# Intermezzo

Gähn!

Geht das jetzt so weiter?

Es wird anstrengend und langweilig.

Das ist ja wie bei einem nackten NetBSD!

# Manuelle Installation

- händisches Editieren diverser Dateien:
  - fstab, network/interfaces, hostname...
- RAID, Crypto, LVM, Netzwerk (Bridge)
- diverser Kleinkram: sudo, rsync, ssh...
- Dauer: 1h-3h
- mit KDE und LaTeX noch länger

Bitte automatisieren!

# Halbautomatische Installation I

- Skript im chroot: `aptitude install ${foo}`
- Vorbereitete Konfigurationen
  - von außen ins chroot kopieren:

```
# rsync -a ${bar}/ /mnt/
```

- Ping Pong auf zwei Konsolen?

# Halbautomatische Installation II

- Vorbereitete Konfigurationen
  - von außen nach /mnt/opt kopieren
- oder im chroot in /opt vorbereiten
  - wobei leider noch der Editor fehlt
- Henne vs. Ei
- Zielkonfiguration sollte initial nach
  - /mnt/opt oder wird interaktiv erstellt

# Halbautomatische Installation III

- Skript mit aptitude und schrittweisem
  - Kopieren der Konfigurationsdateien
  - im chroot von /opt nach /etc usw.
- /root/.bashrc
- /root/.vimrc
- /root/.ssh/\*
- Benutzer (generisch): /etc/skel/

# Beispiel: /etc/apt/sources.list

```
deb          http://ftp.debian.org/debian jessie main
#deb        http://ftp.debian.org/debian jessie contrib
#deb        http://ftp.debian.org/debian jessie non-free

deb-src     http://ftp.debian.org/debian jessie main
#deb-src   http://ftp.debian.org/debian jessie contrib
#deb-src   http://ftp.debian.org/debian jessie non-free

deb         http://security.debian.org/ jessie/updates main
#deb       http://security.debian.org/ jessie/updates contrib
#deb       http://security.debian.org/ jessie/updates non-free

deb-src    http://security.debian.org/ jessie/updates main
#deb-src  http://security.debian.org/ jessie/updates contrib
#deb-src  http://security.debian.org/ jessie/updates non-free

# Code::Blocks
#deb      http://apt.jenslody.de/testing jessie main
#deb-src http://apt.jenslody.de/testing jessie main

# nginx
#deb      http://nginx.org/packages/debian/ jessie nginx
#deb-src http://nginx.org/packages/debian/ jessie nginx

# x2go
#deb      http://packages.x2go.org/debian jessie main
#deb-src http://packages.x2go.org/debian jessie main

# ZFS
# http://zfsonlinux.org/debian.html
```

# daemons im chroot

- ssh
- dbus
- postfix, apache2
- atd (und cron)
- Problem außerhalb von chroot:
  - umount /mnt/proc
  - umount /mnt
  - vgl.: lsof, fuser



# /usr/sbin/policy-rc.d

<http://jpetazzo.github.io/2013/10/06/policy-rc-d-do-not-start-services-automatically/>

```
# here not #!/bin/bash
```

```
PATH="${PATH}:/usr/local/sbin:/usr/local/bin: \  
/usr/sbin:/usr/bin:/sbin:/bin"
```

```
RL='runlevel' # backticks  
if [ "${RL}" = "unknown" ] ; then  
    exit 101  
else  
    exit 0  
fi
```

systemd kennt keinen exit code 0 -- nur 101 und 104

# Github

- <https://github.com/fmgborg>

```
# cd /mnt/opt
```

```
# echo s=source
```

```
# s=https://github.com/fmgborg/bootstrap-install-dgl.git
```

```
# git clone ${s}
```

```
# s=/mnt/opt/bootstrap-install-dgl/bootstrap
```

```
# rsync -a ${s} /mnt/opt/
```

# Stages und Templates

- Nomenklatur?
- debootstrap [291 MiB]
- stage-00.00.00 : +Github [+724 KiB]
- stage-00.00.01 : preboot (Kleinkram) [762 MiB]
  - ohne Kernel und Bootloader
  - systemd-sysv fliegt raus (!)
  - rsync, vim, Keyboard, Crypto, LVM, ...
  - bridge, ssl, ssh, screen, tmux, ...
- stage-00.00.02 : Kernel und Bootloader

# stage-00.00.01

```
# echo Vorbereitung:
# echo c=customization
# c=fsfe
# echo ${c} > /opt/bootstrap/jessie/meta/etc/customizationname
# /opt/bootstrap/jessie/meta/sbin/customization.sh
# nano
  /opt/bootstrap/jessie/system/customized/sbin/run-once-stage.00.01-pre-user.sh
# echo jetzt geht es los:
# sh /opt/bootstrap/jessie/system/customized/sbin/run-once-stage.00.01-pre-user.sh
# echo real 12m11.889s
# echo set a root password now
```

# Customization

- `/usr/local/etc/customization/customization.inc`
- `/usr/local/etc/customization/customization.txt`
- `/etc/issue`
- `/etc/issue.net`
- `/etc/motd`
- `/var/log/fsfe # später`
- `/opt # Debian Etch Debian.jpg`

# GRUB Teil I

- KVM: /dev/mapper/vg-lv.hd0
  - als virtuelle partitionierte Festplatte
  - grub-install mag das device nicht
- in vorhandene VM reinschieben
- dort im chroot auf z.B. /dev/vdb
- Xen + pygrub: GRUB legacy
- GRUB 2: MBR only?

# GRUB Teil II

- eigene virtuelle HDD für GRUB
  - 256 MiB für /boot
- Vorsicht bei einem virtuellen Zoo:
  - LVM logical volumes sind begrenzt
  - (major und) minor number: Byte?
- Dateien als HDD-Container?
  - seriously? Herr vergib ihnen, denn sie wissen nicht, was sie tun

# stage-00.00.02

- # ln -s /proc/mounts /etc/mtab
- # aptitude install linux-image-3.16.0-4-amd64
- # aptitude install firmware-linux-free
- initramfs cryptsetup Problem mit externer crypttab (?)
  - WARNING: invalid line in /etc/crypttab for foo -
- # aptitude install grub-legacy
- # update-grub
- # grub-install /dev/sdc3 # example
- böse : disk-by-id in /boot/grub/device.map



# Fleisch: weitere stages

- vgl. mit tasksel
- generischer Benutzer (sudo, E-Mail)
- Postfix, atd, DNS
- cronjobs (ntpdate, alive logging)
- Webserver, munin, monit, phpsysinfo
- Datenbank, Samba/NFS, VoIP, VPN
- GUI: icewm, XFCE, LXDE, KDE
- reale Benutzer (login, ssh)

# Live Demo USB

- debootstrap : 3-12 Min.
- stage-00.00.01 : 12 Min.
- mkfs 16 GiB : 4 Min.
- rsync + sync 762 MiB : 6 Min.
- stage-00.00.02 : 5 Min.
  
- < 40 Min., Festplatte schneller

# Workshop

- Interesse?
- Themen?
  - stages
  - andere Systeme
  - Optimierungen
  - Ergänzungen
- zweiter grafischer login
- x2go
- Vollautomatisierung

# Verallgemeinerung

- Ubuntu: mit wenigen Modifikationen
- RedHat: yumbootstrap
- Slackware: vielleicht
- NetBSD: vielleicht

# Danke für die Aufmerksamkeit

# Fragen?

[fg-fsfe-dgl-2016@datenschutzraum.org](mailto:fg-fsfe-dgl-2016@datenschutzraum.org)

Download folgt

