

# Building disk images with FAI

Thomas Lange, University of Cologne

[lange@debian.org](mailto:lange@debian.org)

MiniDebConf Cambridge 2017

finger lange@localhost

▶ whoami

- ▶ Sysadmin for more than two decades
- ▶ Debian developer since 2000
- ▶ Diploma in computer science, University of Bonn, Germany
- ▶ SunOS 4.1.1 on SPARC hardware, then Solaris Jumpstart
- ▶ Started FAI in 1999 for my first cluster (16× Dual PII 400 MHz)
- ▶ Several talks and tutorials:  
Linux Kongress, Linuxtag, DebConf, SANE, LCA, FOSDEM,  
CeBit, OSDC, UKUUG, FrOSCon, Chemnitzer Linuxtag

# What is FAI?

- ▶ FAI = Fully Automatic Installation
- ▶ Non-interactive system for customized installations
- ▶ Bare metal, virtual machines, chroot environment
- ▶ Installation via network (PXE), CD, USB
- ▶ Live CD, diskless client
- ▶ Debian, Ubuntu, CentOS, RHEL, SUSE

## Customized disk images

- ▶ Always had the idea to support cloud images
- ▶ DebConf15 Heidelberg:  
Creating bootable Debian images by Riku Voipio
- ▶ July 2016 first version of `fai-diskimage`
- ▶ Sep 2016, FAI 5.2, first release
- ▶ Debian Cloud sprint 2016: Try out `fai-diskimage`
- ▶ Sep 2017, FAI 5.4, adding cross architecture support
- ▶ Debian Cloud sprint 2017, FAI will be used for cloud images for Google, Amazon, Microsoft
- ▶ Config spaces for `vagrant` and `OpenStack` are available

## Creating disk images with FAI

- ▶ `fai-diskimage` only 200 lines of code
- ▶ Create empty disk image
- ▶ Create loop device
- ▶ Call `fai install $tmpdir`
- ▶ Convert raw image into other format on demand

```
# export FAI_BASEFILEURL=https://fai-project.org/  
download/basefiles/  
# CL="DEBIAN,STRETCH64,AMD64,FAIBASE,GRUB_PC,DHCPC,CLOUD"  
# fai-diskimage -vu cloud3 -S2G -c$CL cloud.raw
```

# The config space

```
|-- class/  
|   |-- FAIBASE.var  
|   '-- DEBIAN.var
```

```
|-- disk_config/  
|   |-- FAIBASE  
|   |-- CLOUD  
|   '-- demohost
```

```
|-- basefiles/
```

```
|-- package_config/  
|   |-- DEBIAN  
|   |-- FAISERVER  
|   |-- XFCE  
|   |-- GNOME  
|   '-- server07
```

# Variables

Example: `.../class/DEBIAN.var:`

```
FAI_ALLOW_UNSIGNED=1

KEYMAP=de-latin1-noddeadkeys
UTC=yes
TIMEZONE=Europe/Berlin

ROOTPW=' $1$kBn.MWc0.B$djxB38B7dMkplhJHPf2d1'

release=stretch
apt_cdn=http://deb.debian.org
security_cdn=http://security.debian.org
```

- ▶ Define your own variables
- ▶ Use the variables in `.../scripts/*`

## Disk partitioning

Example: .../disk\_config/FAIBASE:

```
disk_config disk1      preserve_always:8 fstabkey:uuid

primary /              2G-50G      ext4  rw,noatime,errors=remount-ro
logical swap           200-10G     swap  sw
logical /home          1G-         ext4  defaults
```

Example: .../disk\_config/CLOUD:

```
disk_config disk1 disklabel:msdos fstabkey:uuid align-at:1M

primary /              300-        ext4  rw,barrier=0,discard tuneopts="-c 0 -i 0"
```

- ▶ File systems: ext[2,3,4], vfat, xfs, ReiserFS, NTFS, btrfs



# RAID, LVM

```
disk_config disk1
primary - 50-100 - -
primary swap 1G swap sw
primary - 2G-10G - -
logical - 0- - -
logical - 0- - -

disk_config disk2 sameas:disk1

disk_config raid
raid1 /boot disk1.1,disk2.1 ext4 rw
raid1 / disk1.3,disk2.3 ext4 rw,acl,user_xattr
raid1 - disk1.5,disk2.5 - -
raid1 - disk1.6,disk2.6 - -

disk_config lvm
vg volg1 md2,md3
volg1-usr /usr 8G ext4 rw createopts="-O dir_index"
volg1-var /var 2G ext4 rw createopts="-O dir_index"
volg1-hl /home/local 10G ext4 rw,acl,user_xattr,noexec,nosuid
volg1-es /export/sites 3G ext4 rw createopts="-O none"
volg1-v /vservers 8G ext4 rw createopts="-O ^dir_index"
```

# Software package installation

Example: .../package\_config/DEBIAN:

```
PACKAGES install-norec
file less rsync pciutils usbutils
openssh-client openssh-server
procinfo nullmailer locales
console-setup kbd
unattended-upgrades

PACKAGES install NONFREE
firmware-bnx2 firmware-bnx2x firmware-realtek
firmware-linux-nonfree

PACKAGES install AMD64
linux-image-amd64 initramfs-tools
mementest86+

PACKAGES install ARM64
grub-efi-arm64
linux-image-arm64
```

- ▶ Supported package tools: aptitude, apt, apt-get, smart, rpm, urpmi, y2pms, yast, yum, zypper, dnf

# Scripts and files

```
|-- scripts/
|   |-- FAIBASE/
|   |   |-- 10-misc           Bourne shell script
|   |   |-- 30-interface     Bourne shell script
|   |   |-- 40-misc          Cfengine script
|   |-- DEMO/
|       |-- 10-misc           Perl script
|       |-- 30-demo          Cfengine script
|
|-- files/
    |-- etc/
        |-- X11/
            |-- xorg.xonf/     fcopy /etc/X11/xorg.conf
                |-- FAIBASE
                |-- MATROX
                |-- CAD
                |-- demohost
```

## Config scripts

```
#!/bin/bash

if ifclass DISABLE_IPV6; then
    ainsl -av /etc/sysctl.d/70-disable-ipv6.conf \
        'net.ipv6.conf.all.disable_ipv6 = 1'
    ainsl -av /etc/sysctl.d/70-disable-ipv6.conf \
        'net.ipv6.conf.lo.disable_ipv6 = 0'
fi

$ROOTCMD shadowconfig on
sed -i -e 's/^#PasswordAuthentication yes/PasswordAuthentication no/' \
    $target/etc/ssh/sshd_config
sed -i -e 's/^PermitRootLogin .*/PermitRootLogin no/' \
    $target/etc/ssh/sshd_config
ainsl /etc/ssh/sshd_config 'ClientAliveInterval 420'

ainsl -v /etc/fstab "${hserver}:/home /home nfs ro 0 0"
ainsl -av /etc/default/ssh 'SSHD_OPTS=-4'

fcopy -Mv /etc/hosts.allow /etc/hosts.deny
fcopy -M /etc/X11/xorg.conf
```

# FAI users

- ▶ Anonymous, financial industry, 32.000 hosts
- ▶ LVM insurance, 10.000 hosts
- ▶ City of Munich, 16.000 hosts
- ▶ Albert Einstein Institute, 1725 hosts
- ▶ Zivit, 260 hosts on two IBM z10 EC mainframes
- ▶ Archive.org, 1200 bare metal + 800 KVM hosts
- ▶ XING AG, 300-400 hosts
- ▶ Opera Software, ~300 hosts
- ▶ Stanford University, 450 hosts
- ▶ MIT Computer science research lab, 200 hosts
- ▶ The Wellcome Trust Sanger Institute, 540 hosts
- ▶ Deutsches Elektronen-Synchrotron, 273 hosts
- ▶ Mobile.de, ~600 hosts
- ▶ Electricité de France (EDF), 1500 hosts
- ▶ BUF, digital visual effects company, 1000 hosts
- ▶ ETH Zurich, systems group, ~300 hosts
- ▶ StayFriends, 700+ hosts
- ▶ Grml, creating eight different ISOs, daily builds

## Live demo time

- ▶ Build simple image
- ▶ Build XFCE image
- ▶ Build ARM64 image

## FAI - Fully Automatic Installation

### Home

- Features
- Poster / Flyer
- User reports
- Mailing Lists / IRC / Wiki
- Clusters built with FAI

### Videos / Screenshots

### Download

- FAI CD
- Packages
- FAI questionnaire

### Documentation

- FAI Guide
- Manual pages
- Other documentation
- Talk slides and videos

### Developers

- Sources / Bugs
- Roadmap
- Team

### Contact / Support

### Site search

FAI is a non-interactive system to install, customize and manage Linux systems and software configurations on computers as well as virtual machines and chroot environments, from small networks to large-scale infrastructures like clusters and cloud environments.

It's a tool for unattended mass deployment of Linux. You can take one or more virgin PC's, turn on the power, and after a few minutes, the systems are installed, and completely configured to your exact needs, without any interaction necessary.

Motto: Plan your installation, and FAI installs your plan.

FAI ISO (1005 MB)

### NEWS

- I will attend the [Mini-Debconf in Cambridge](#) end of november. There I will announce a new FAI feature.
- [8 Nov 2017] **FAI 5.5 released** and new ISO images are available
- [18 Oct 2017] We had a great Debian cloud sprint in Seattle/Bellevue. We've created a [FAI config space](#) for GCE, Azure, EC2 and Openstack cloud images and a huge test suite.
- [6 Oct 2017] [Video](#) of creating a cross architecture disk image for ARM64
- [5 Oct 2017] [FAI 5.4 released](#), new ISO images available
- [6 Sep 2017] Have a look at [the video](#) of my FAI demo at DebConf 17
- [14 Oct 2016] [FAI 5.2 is going to the cloud](#)

### Features

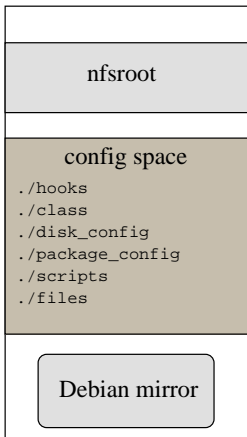
- Installs and updates Debian, Ubuntu, CentOS, RHEL, SUSE, ...
- Centralized deployment and configuration management
- Installs virtual machines using KVM, XEN or VirtualBox and Vserver
- Easy set up of software RAID and LVM
- Full remote control via ssh during installation
- Integrated disaster recovery system
- Every stage can be customized via hooks



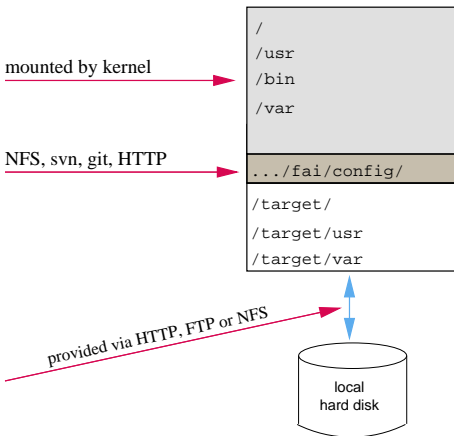
# Questions?

## FAI overview

### install server



### install client



- ▶ The configuration is stored on the install server
- ▶ The installation runs on the client