

FAI: Anpassen der FAI-Konfiguration

Nachfolgend werden Anpassungen an der FAI-Konfiguration beschrieben. Diese Konfiguration, die im Normalfall unter `/srv/fai/config/` zu finden ist und aus dem Paket „`fai-doc`“ stammt, befindet sich nach der Paketinstallation in einem Grundzustand. Sinnvolle Ergänzungen und Erweiterungen sind Bestandteil des Dokuments.

Voraussetzung

Es wird vorausgesetzt, dass ein FAI-Server vorhanden ist, wie er zum Beispiel [hier](#) beschrieben wird.

Abschnitte

Der Blick in das Verzeichnis „`/srv/fai/config/`“ zeigt folgende Verzeichnisse:

- `basefiles`: FAI-Basisdateien
- `class`: FAI-Klassen (Rechnerklassen)
- `debconf`: Lokalisation und Tastaturlayout
- `disk_config`: Festplattenkonfiguration
- `files`: Dateien für den FAI-Client
- `hooks`: Einsprungskripte
- `package_config`: Paketlisten
- `scripts`: Skripte
- `tests`: FAI-Test

class

In diesem Verzeichnis erfolgt die Klassifizierung durch den Rechnernamen. Je nach Rechnername werden eine oder mehrere Klassen identifiziert und nachfolgend ausgeführt.

- Anpassen der Datei „**50-host-classes**“:

```
case $HOSTNAME in
    *client*)
        echo "BASIS DEBDEU" ;;
    *)
        echo "BASIS" ;;
esac
```

```
ifclass -o I386 AMD64 && echo GRUB_PC
exit 0
```

- Erstellen der Datei „**BASIS.var**“:

```
# Erlaube die Installation von Paketen aus unsignierten Quellen
FAI_ALLOW_UNSIGNED=1

# Zeitzone
UTC=yes
TIMEZONE=Europe/Berlin

# Administratorkennwort (Standard: 'fai')
# Neues Kennwort generieren mit 'makepasswd':
# ~# echo "<Kennwort>" > /tmp/kennwort && makepasswd --crypt-md5 --clearfrom
/tmp/kennwort && rm -f /tmp/kennwort;
ROOTPW='$1$kBnWc0.E$djxB128U7dMkr!tJHPf6d1'

# Installationsstop ab diesem Fehlerwert
STOP_ON_ERROR=700
```

- Anpassen der Datei „**DEBDEU.var**“:

```
# Konsolenfont
CONSOLEFONT=

# Tastaturlayout
KEYMAP=de-latin1-nodeadkeys

# Der Parameter 'MODULESLIST' beinhaltet alle Module, die auf einem
# neuen System waehrend der Installation geladen werden.
# Weitere Module koennen zur Variablen '$kernelmodules' in der Datei
# 20-hwdetect.source eingetragen werden
MODULESLIST="usbkbd ehci-hcd ohci-hcd uhci-hcd usbhid psmouse"
```

debconf

Hier befinden sich alle Konfigurationsdateien, die für DEBIAN spezifisch sind.

- Erstellen der Datei „**BASIS**“:

```
exim4-config exim4/dc_eximconfig_configtype select local delivery only; not
```

on a network

```

locales locales/default_environment_locale select en_US.UTF-8
locales locales/locales_to_be_generated multiselect en_US.UTF-8 UTF-8
keyboard-configuration keyboard-configuration/modelcode string
pc105
keyboard-configuration keyboard-configuration/xkb-keymap select us
keyboard-configuration keyboard-configuration/variant select USA
keyboard-configuration keyboard-configuration/model select Generic 105-
key (Intl) PC
keyboard-configuration keyboard-configuration/layoutcode string us
keyboard-configuration keyboard-configuration/optionscode string
ctrl:nocaps,terminate:ctrl_alt_bksp

```

- Erstellen der Datei „**DEBDEU**“:

```

locales locales/default_environment_locale select de_DE.UTF-8
locales locales/locales_to_be_generated multiselect de_DE.UTF-8 UTF-8
xserver-xorg xserver-xorg/config/inputdevice/keyboard/layout string
de
xserver-xorg xserver-xorg/config/inputdevice/keyboard/model string
pc105
xserver-xorg xserver-xorg/config/inputdevice/keyboard/options string
lv3:ralt_switch
xserver-xorg xserver-xorg/autodetect_monitor boolean false
xserver-xorg xserver-xorg/autodetect_keyboard boolean true
xserver-xorg xserver-xorg/autodetect_mouse boolean true
xserver-xorg xserver-xorg/autodetect_video_card boolean true
console-setup console-setup/variant select Germany
console-setup console-setup/charmap select UTF-8
console-setup console-setup/layoutcode string de
console-setup console-setup/compose select No compose key
console-setup console-setup/fontsize-text select 16
console-setup console-setup/optionscode string lv3:ralt_switch
console-setup console-setup/layout select Germany
console-setup console-setup/detected note
console-setup console-setup/variantcode string
console-setup console-setup/codesetcode string Lat15
console-setup console-setup/modelcode string pc105
console-setup console-setup/ask_detect boolean false
console-setup console-setup/altgr select Right Alt
console-setup console-setup/ttys string /dev/tty[1-6]
console-setup console-setup/model select Generic 105-key (Intl) PC
console-setup console-setup/fontsize-fb select 16
console-setup console-setup/switch select No temporary switch
console-setup console-setup/codeset select # Latin1 and Latin5 -
western Europe and Turkic languages
console-setup console-setup/toggle select No toggling
console-setup console-setup/fontface select VGA
console-setup console-setup/fontsize string 16

```

disk_config

Hier wird die Festplattenkonfiguration festgelegt.

- Erstellen der Datei „**BASIS**“ (ausgelegt für eine Festplatte mit mindestens 15 GB):

```
disk_config disk1 disklabel:msdos bootable:1 fstabkey:uuid
primary      /boot          500          ext2        rw
logical      swap           2G           swap        sw
logical      /var/log       4G           ext4        rw,noatime,nodev
primary      /              8G-         ext4
rw,noatime,errors=remount-ro
```

- Erstellen der Datei „**LVM**“ (ausgelegt für eine Festplatte mit mindestens 15 GB):

```
disk_config disk1 disklabel:msdos bootable:1 fstabkey:uuid
primary      /boot          256M        ext3        defaults,rw
createopts="-L boot" tuneopts="-c 180 -i 180d"
primary      -              1G-         -          -

disk_config lvm
vg vg001 disk1.2
vg001-swap    swap           2G          swap        sw
createopts="-L swap"
vg001-var_log /var/log       4G          ext4        defaults,rw
createopts="-L var_log" tuneopts="-c 180 -i 180d"
vg001-root    /              8G          ext4        defaults,rw,errors=remount-ro
ro createopts="-L root"  tuneopts="-c 180 -i 180d"
```



Sollte die Festplatte größer als 15 GB sein, kann der freier Speicher als logische Volumes verwendet werden.

files

Hier befinden sich alle Dateien, die während der Installationsphase auf den Zielrechner kopiert werden. Jede Dateiname wird dabei als Verzeichnis erstellt, um darin die eigentliche Datei als Klassenname abzulegen.

- Erstellen der Datei „**etc/default/console-setup/DEBDEU**“:

```
# Aktivieren der TTY-Konsolen
ACTIVE_CONSOLES="/dev/tty[1-6]"

# Zeichentabelle
# Weitere bekannte Werte: UTF-8 ARMSII-8 CP1251
# CP1255 CP1256 GEORGIAN-ACADEMY GEORGIAN-PS IBM1133 ISIRI-3342
# ISO-8859-1 ISO-8859-2 ISO-8859-3 ISO-8859-4 ISO-8859-5 ISO-8859-6
# ISO-8859-7 ISO-8859-8 ISO-8859-9 ISO-8859-10 ISO-8859-11 ISO-8859-13
# ISO-8859-14 ISO-8859-15 ISO-8859-16 KOI8-R KOI8-U TIS-620 VISCII
CHARMAP="UTF-8"

# Symboltabelle
# Weitere bekannte Symboltabellen: Arabic Armenian CyrAsia CyrKoi
# CyrSlav Ethiopian Georgian Greek Hebrew Lao Lat15 Lat2 Lat38 Lat7
# Thai Uni1 Uni2 Uni3 Vietnamese
CODESET="Lat15"

# Schriftartenname und -groesse
# Weitere bekannte Werte sind: VGA (sizes 8, 14 and 16), Terminus
# (sizes 12x6, 14, 16, 20x10, 24x12, 28x14 and 32x16), TerminusBold
# (sizes 14, 16, 20x10, 24x12, 28x14 and 32x16), TerminusBoldVGA
# (sizes 14 and 16), Fixed (sizes 13, 14, 15, 16 and 18), Goha
# (sizes 12, 14 and 16), GohaClassic (sizes 12, 14 and 16)
FONTFACE="VGA"
FONTSIZE="16"

# Tastaturmodell und -layout
XKBMODEL="pc105"
XKBLAYOUT="de"
XKBVARIANT="nodeadkeys"
XKBOPTIONS="terminate:ctrl_alt_bksp"
```

- Erstellen der Datei „**etc/default/locale/DEBDEU**“:

```
LANG="de_DE.UTF-8"
```

hooks

Hier werden die Skripte abgelegt, die während der Installation aufgerufen werden.

- Erstellen der Datei „**updatebase.DEBDEU**“:

```
#!/bin/bash
# Kopieren der Dateien fuer 'console-setup'
```

```
fcopy -v /etc/default/console-setup
```

- die Datei muss noch ausführbar gemacht werden:

```
~# chmod +x updatebase.DEBDEU
```

package_config

Hier werden alle Pakete eingetragen, die zusätzlich zur Basisinstallation auf das System gebracht werden sollen.

- Erstellen der Datei „**BASIS**“:

```
PACKAGES aptitude AMD64  
linux-image-amd64 initramfs-tools  
memtest86+
```

```
PACKAGES aptitude DHCP  
isc-dhcp-client
```

```
PACKAGES aptitude GRUB_PC  
grub-pc grub-legacy- lilo-
```

```
PACKAGES aptitude  
fai-client  
cron  
debconf-utils  
file  
hdparm  
less  
linuxlogo  
nfs-common  
rdate  
rsync  
openssh-client  
openssh-server  
strace  
time
```

```
procinfo
eject
locales
console-setup
kbd
pciutils
usbutils
lvm2
```

scripts

In diesem Verzeichnis liegen die Skripte, die nach der Installation aufgerufen werden. Es werden zwei neue Verzeichnisse „**BASIS**“ und „**DEBDEU**“ erstellt.

- Kopieren der Datei „**FAIBASE/10-misc**“ nach „**BASIS/**“:

```
#!/bin/bash
# (c) Thomas Lange, 2001-2012, lange@debian.org

error=0; trap 'error=$(( $?>$error??:$error ))' ERR # save maximum error code

echo $TIMEZONE      > $target/etc/timezone
cp -f /usr/share/zoneinfo/${TIMEZONE} $target/etc/localtime

if [ -n "$IPADDR" ]; then
    ifclass DHCPD || ainsl -s /etc/hosts "$IPADDR $HOSTNAME.$DOMAIN
$HOSTNAME"
else
    ifclass DHCPD && ainsl -s /etc/hosts "127.0.0.1 $HOSTNAME"
fi

fcopy -iM /etc/hosts /etc/motd

# make /root accessible only by root
chmod -c 0700 $target/root
chown -c root:root $target/root
# copy default dotfiles for root account
fcopy -ir /root

chmod -c 1777 ${target}/tmp
chown -c 0:0  ${target}/tmp

exit $error
```

- Kopieren der Datei „**FAIBASE/20-removable_media**“ nach „**BASIS/**“:

```
#!/bin/bash
# (c) Thomas Lange, 2006,2009, lange@debian.org
# create entries for removable media in fstab and directories in /media

[ -b $target/dev/fd0 ] && ainsl /etc/fstab "/dev/fd0 /media/floppy auto
users,noauto 0 0"

cdromlist() {
    [ -f /proc/sys/dev/cdrom/info ] || return
    devs=$(grep 'drive name:' /proc/sys/dev/cdrom/info | cut -d ":" -f 2)
    for d in $devs; do
        echo $d
    done
}

fstabline () {
    line=$(printf "%-15s %-15s %-7s %-15s %-7s %s\n" "$1" "$2" "$3" "$4"
"$5" "$6")
    ainsl /etc/fstab "$line"
}

i=0
for cdrom in $(cdromlist | tac); do
    [ $i -eq 0 ] && ln -s cdrom0 $target/media/cdrom
    mkdir -p $target/media/cdrom$i
    fstabline /dev/$cdrom /media/cdrom$i udf,iso9660 ro,user,noauto 0 0
    i=$((i + 1))
done
```

- Kopieren der Datei „**DEBIAN/10-rootpw**“ nach „**DEBDEU/**“:

```
#!/bin/bash

error=0; trap 'error=$(( $?>$error??:$error ))' ERR # save maximum error code

# set root password
$ROOTCMD usermod -p $ROOTPW root

exit $error
```

- Kopieren der Datei „**DEBIAN/30-interface**“ nach „**DEBDEU/**“:

```
#!/bin/bash

error=0; trap 'error=$(( $?>$error??:$error ))' ERR # save maximum error code

if ifclass DHCPD && [ $FAI_ACTION != "softupdate" ]
```



```

then
    cat > $target/etc/network/interfaces <<-EOF
    # generated by FAI
    auto lo eth0
    iface lo inet loopback
    iface eth0 inet dhcp
EOF
elif [ $FAI_ACTION != "softupdate" ]
then
    [ -n "$IPADDR" ] && cat > $target/etc/network/interfaces <<-EOF
    # generated by FAI
    auto lo eth0
    iface lo inet loopback
    iface eth0 inet static
    address $IPADDR
    netmask $NETMASK
    broadcast $BROADCAST
    gateway $GATEWAYS
EOF
    [ -n "$NETWORK" ] && echo "localnet $NETWORK" > $target/etc/networks
    [ -L $target/etc/resolv.conf ] || cp -p /etc/resolv.conf $target/etc
fi

# here fcopy is mostly used, when installing a client for running in a
# different subnet than during the installation
fcopy -iM /etc/resolv.conf
fcopy -iM /etc/network/interfaces /etc/networks

exit $error

```

- Kopieren der Datei „**DEBIAN/40-misc**“ nach „**DEBDEU/**“:

```

#!/bin/bash
# (c) Thomas Lange, 2001-2011, lange@debian.org
# (c) Michael Goetze, 2010-2011, mgoetze@mgoetze.net

error=0; trap 'error=$(( $?>$error?${?}:$error ))' ERR # save maximum error code

# a list of modules which are loaded at boot time
for module in $MODULESLIST; do
    ainsl -a /etc/modules "^$module$"
done

fcopy -Mv /etc/hostname || echo $HOSTNAME > $target/etc/hostname
ainsl -av /etc/mailname ${HOSTNAME}
sed -i -e "s/^UTC.*;/UTC=${UTC}/" $target/etc/default/rcS
sed -i -e 's#/sbin/getty 38400#/sbin/getty -f /etc/issue.linuxlogo 38400#'
${target}/etc/inittab

```

```
[ $FAI_ACTION = "softupdate" ] || cp /etc/fai/fai.conf
$target/etc/fai/fai.conf
ainsl -av /etc/fai/fai.conf "FAI_CONFIG_SRC=$FAI_CONFIG_SRC"

fcopy -Miv /etc/fai/fai.conf

exit $error
```

Softwareaktualisierung

FAI beherrscht neben der Installation eines Rechners auch die Softwareaktualisierung. Dabei werden alle Pakete, die laut der FAI-Konfiguration fehlen, auf dem Client nachinstalliert.

- Auslösen der Softwareaktualisierung auf dem FAI-Client:

```
~# fai -N softupdate
```

Erweiterung der Softwareaktualisierung

Der Mechanismus der Softwareaktualisierung beherrscht das nachträgliche Installieren von deinstallierten Paketen, aber nicht das Deinstallieren von zusätzlich installierten Paketen. Somit kann für einen FAI-Client kein bestimmter Softwarestand per FAI definiert werden. Dieser Mechanismus wurde in minimaler Form nachträglich hinzugefügt.

- Erstellen der Datei „**scripts/DEBDEU/90-package-list**“:

```
#!/bin/bash

error=0; trap 'error=$(( $?>$error?${?}:$error ))' ERR # save maximum error code

if [ $FAI_ACTION != "softupdate" ]
then
    # Sicherung der Paketliste
    mkdir -p $target/usr/local/share/fai
    $ROOTCMD dpkg --get-selections | awk '{ print $1 }' | sort | uniq >
    $target/usr/local/share/fai/paketliste.txt
    md5sum $target/usr/local/share/fai/paketliste.txt >
    $target/usr/local/share/fai/paketliste.md5
fi
```

```
exit $error
```

Das Skript ermittelt am Ende der Installation alle installierten Pakete und speichert sie in einer Datei im Verzeichnis /root/ ab.

- die Datei muss noch ausführbar gemacht werden:

```
~# chmod +x scripts/DEBDEU/90-package-list
```

- Erstellen der Datei „**scripts/DEBDEU/95-package-check**“:

```
#!/bin/bash

error=0; trap 'error=$(( $?>$error??:$error ))' ERR # save maximum error code

if [ $FAI_ACTION == "softupdate" ]
then

    if [ -s "$target/usr/local/share/fai/paketliste.txt" ]; then
        Var_ListeSoll="$target/usr/local/share/fai/paketliste.txt";

    else
        echo "Keine Soll-Paketliste gefunden. Softwareüberprüfung
wird übersprungen.";
        exit 0;

    fi;

    $ROOTCMD dpkg --get-selections | awk '{ print $1 }' | sort | uniq >
/tmp/pakete.txt
    Var_ListeIst="/tmp/pakete.txt";

    Var_PaketeLoeschen=$(cat "${Var_ListeSoll}" "${Var_ListeIst}" | sort |
uniq -u | tr "\n" " ");

    if [ ${#Var_PaketeLoeschen} -ne 0 ]; then
        $ROOTCMD apt-get purge -qy --force-yes
        ${Var_PaketeLoeschen};
        error=$?

    else
        echo "Keine Pakete installiert, die wieder deinstalliert
werden müssen. Softwareüberprüfung erfolgreich.";
        error=0

    fi;

    rm -f "/tmp/pakete.txt";
```

```
fi
```

```
exit $error
```

Dieses Skript wird ausschließlich während der Softwareaktualisierung ausgeführt und ermittelt die Pakete, die nach der Installation zusätzlich installiert wurden (über den Vergleich der beiden Paketlisten). Alle zusätzlichen Pakete werden deinstalliert.

- die Datei muss noch ausführbar gemacht werden:

```
~# chmod +x scripts/DEBDEU/95-package-check
```

DEBIAN, FAI

From:
<https://looper.de/wiki/> - Linux4Ever

Permanent link:
https://looper.de/wiki/doku.php?id=sborni-linux:betriebssysteme:debian-linux:debian-linux_fai-konfiguration

Last update: **2025/12/11 15:00**

