

# Debian (und viele Derivate)

- ["snmpd: error on subcontainer 'ia\\_addr' insert \(-1\)" fixen](#)
- [Anlegen einer Network Bridge unter Debian](#)
- [Custom-Kernel bauen](#)
- [Debian für Fedora-User](#)
- [Default-Editor unter Debian konfigurieren](#)
- [Installation von Sun Java unter Debian](#)
- [PHP5 ohne Apache installieren](#)
- [Possible missing firmware for module r8169](#)
- [Vorlage für APT sources.list \(aktuell Debian Bookworm\)](#)

# "snmpd: error on subcontainer 'ia\_addr' insert (-1)" fixen

Wenn man in einem frischen Debian (Ubuntu ist auch betroffen) SNMPd verwendet, tauchen immer wieder diese Meldungen im Syslog auf:

```
Apr 11 14:19:05 hostname snmpd[12512]: error on subcontainer 'ia_addr' insert (-1)
```

Dieser Bug ist schon sehr lange in [Debian](#) (und [Ubuntu](#)) und lässt sich wie folgt fixen.

**Ab Debian Stretch** muss man das Systemd Unitfile von snmpd anpassen. Vorgaben in /etc/default/ werden nicht mehr beachtet:

```
sudo -i
systemctl edit snmpd.service

### einfügen:
[Service]
ExecStart=
ExecStart=/usr/sbin/snmpd -LS6d -Lf /dev/null -u Debian-snmp -g Debian-snmp -I -
smux,mteTrigger,mteTriggerConf -f
```

Speichern, danach:

```
systemctl restart snmpd.service
```

**Vor Debian Stretch bzw. Systemd** hat es so funktioniert:

```
sed -i 's/Lsd/LS6d/g' /etc/default/snmpd
service snmpd restart
```

# Anlegen einer Network Bridge unter Debian

Für meinen netten kleinen Router mit dem Alix2c3 Mainboard wollte ich die 3 Netzwerkports bridgen, damit ich ihn quasi auch als Switch verwenden kann. Momentan habe ich darauf das [Voyage-Linux](#) installiert, das speziell auf embedded Geräte ausgelegt ist und z.B. viel mit tmpfs arbeitet um zuviele Schreibzugriffe auf das Compact-Flash Device zu vermeiden. Voyage-Linux basiert auf Debian, die aktuellste Version baut sogar schon auf Lenny auf.

Die folgende Anleitungen sollte aber für alle Debian-Systeme gleich gut funktionieren.

Zuerst müssen die Tools für das Bridging installiert werden:

```
aptitude install bridge-utils
```

Jetzt wird die Bridge in /etc/network/interfaces konfiguriert. Zuerst muss dazu die alte Konfiguration z.B. für eth0 deaktiviert werden, dazu die alten Einträge einfach auskommentieren.

Jetzt folgenden Eintrag hinzufügen, die IP-Adresse ist die vorher für eth0 eingetragene:

```
auto br0
iface br0 inet static
    bridge_ports eth0 eth1 eth2
    address 192.168.1.2
    netmask 255.255.255.0
    network 192.168.1.0
    broadcast 192.168.1.255
    gateway 192.168.1.1
```

Für den Parameter bridge\_ports kann auch „all“ eingesetzt werden, dann werden alle ethX-Ports für die Bridge verwendet. Nach einem Reboot, bzw. dem Neustart des Netzwerks ist die Bridge einsatzbereit.

Kontrollieren kann man die Einstellungen der Bridge so:

```
alix:~# brctl show
bridge name      bridge id        STP enabled      interfaces
br0              8000.000db91527b0 no                eth0
                  eth1
                  eth2
```

# Custom-Kernel bauen

ungetestet:

```
apt-get install linux-source build-essential kernel-package
mkdir /root/kernels
cd /root/kernels
apt-get source linux
cd linux-*
cp /boot/config-3.2.0-4-amd64 config
vi Makefile (Set SUBLEVEL = 0)
make oldconfig
make menuconfig (Enable "General setup => Control group support => Group CPU scheduler => CPU ba
make-kpkg -j12 --append-to-version "-1-amd64-ovtec" --revision $REVISION --initrd buildpackage (
dput -c ~/cdn-packages/dput.rc -u ovtec linux-source-3.2.0-amd64-ovtec+$REVISION_amd64.changes
```

# Debian für Fedora-User

Bei Debian ist für den Fedora-User einiges zu beachten. Manche Befehle gibts nicht, bzw. haben eine andere Syntax.

- User anlegen (Bei Fedora wird das Homeverzeichnis automatisch erzeugt. Debian braucht den -m Switch

```
useradd -m -s /bin/bash -d /home/wartung -c "Wartungsaccount" wartung
```

- Passwort im Batchmode ändern (Fedoras passwd kennt einen Schalter `-stdin` um Kennwörter von der Standardeingabe zu lesen)

```
echo "wartung:blablabla" | chpasswd
```

# Default-Editor unter Debian konfigurieren

Standardmäßig ist bei Debian Nano als Editor eingerichtet. Dies kann für alle User über das Alternative-System geändert werden:

```
root@webl:~# update-alternatives --config editor

There are 4 alternatives which provide `editor'.

   Selection      Alternative
-----
*+          1      /bin/ed
            2      /bin/nano
            3      /usr/bin/vim.tiny
            4      /usr/bin/vim.basic

Press enter to keep the default[*], or type selection number: 4
Using '/usr/bin/vim.basic' to provide 'editor'.
```

Als normaler User funktioniert das nicht, daher müssen folgende Zeilen im eigenen Profil (z.b. in ~/.bashrc) eingetragen werden:

```
EDITOR=vi
export EDITOR
alias editor=$EDITOR
```

Danach neu anmelden oder die Datei neu sourcen (. bashrc).

Alternativ den nano Editor einfach deinstallieren:

```
apt install vim nano-
```

# Installation von Sun Java unter Debian

## Debian Wheezy und neuer

Für die Installation von Oracle/Sun Java unter Debian (ohne ein third party repository) gibts ein kleines Tool um aus dem Originaldownload ein dpkg-File zu machen.

Die aktuelle Version des Oracle Java JDKs kann hier heruntergeladen werden (siehe unten, per wget):

<http://www.oracle.com/technetwork/java/javase/downloads/index.html>

Folgendes Schritte sind dafür notwendig:

Aktivieren von contrib-Paketen in der sources.list von apt, z.B.:

[/etc/apt/sources.list](#)

```
deb http://debian.wavecon.de/mirror/ jessie main contrib
deb http://security.debian.org jessie/updates main contrib
deb http://debian.wavecon.de/repository/ wavecon main
```

Nötige Pakete installieren und Java herunterladen. Aktuelle Versionen von hier:

<http://www.oracle.com/technetwork/java/javase/downloads/index.html>

```
sudo aptitude -y install java-package build-essential libxslt1.1 libgl1-mesa-glx libgtk2.0-0 lib
wget --no-check-certificate --no-cookies --header "Cookie: oraclelicense=accept-securebackup-coo
# .deb Paket bauen (make-jpg läßt sich nicht als User root ausführen):
fakeroot make-jpkg jdk-8u131-linux-x64.tar.gz
```

Java installieren und aktivieren:

```
sudo dpkg -i oracle-java8-jdk_8u131_amd64.deb

update-java-alternatives -l
update-java-alternatives -s jdk-8-oracle-x64
```

Alles fertig:

```
# java -version
java version "1.8.0_131"
Java(TM) SE Runtime Environment (build 1.8.0_131-b11)
Java HotSpot(TM) 64-Bit Server VM (build 25.131-b11, mixed mode)
```

## Debian Lenny

Um Suns Java 6 unter Debian zu installieren gibts einige Hilfestellungen. Folgendes ist zu tun:

1. Ergänzen der sources.list um „non-free“-Paketquellen. Meine sources.list sieht dann z.B. so aus:

```
#
# deb cdrom:[Debian GNU/Linux 5.0.3 _Lenny_ - Official amd64 NETINST Binary-1 20090906-11:59]/ 1
#deb cdrom:[Debian GNU/Linux 5.0.3 _Lenny_ - Official amd64 NETINST Binary-1 20090906-11:59]/ le

deb http://ftp.de.debian.org/debian/ lenny main non-free
deb-src http://ftp.de.debian.org/debian/ lenny main non-free

deb http://security.debian.org/ lenny/updates main non-free
deb-src http://security.debian.org/ lenny/updates main non-free

deb http://volatile.debian.org/debian-volatile lenny/volatile main non-free
deb-src http://volatile.debian.org/debian-volatile lenny/volatile main non-free
```

2. „\$ apt-get update“

3. Die Installation via „\$ apt-get -y install sun-java6-plugin sun-java6-jre sun-java6-jdk sun-java6-javadb sun-java6-bin sun-java6-doc sun-java6-fonts“. Wer die Dokumentation mitinstalliert muss diese vom Sun-Server runterladen und nach /tmp/ legen (wer ohne auskommt lässt einfach sun-java6-doc im apt-Aufruf weg):

```
Setting up sun-java6-doc (6-12-1) ...
...

This package is an installer package, it does not actually contain the
JDK documentation. You will need to go download one of the
archives:

    jdk-6u12-docs.zip jdk-6u12-docs-ja.zip

(choose the non-update version if this is the first installation).
Please visit

    http://java.sun.com/javase/downloads/

now and download. The file should be owned by root.root and be copied
to /tmp.

[Press RETURN to try again, 'no' + RETURN to abort]
```

4. Alternatives auf Java umstellen:

```
# verfügbare Java-Installationen anzeigen:
$ update-java-alternatives -l

# umstellen auf Sun Java 6:
$ update-java-alternatives -s java-6-sun
```

5. Korrekte Installation überprüfen:

```
$ java -version
java version "1.6.0_12"
Java(TM) SE Runtime Environment (build 1.6.0_12-b04)
Java HotSpot(TM) 64-Bit Server VM (build 11.2-b01, mixed mode)
```



# PHP5 ohne Apache installieren

Bei einem „apt-get install php5“ wird bei Debian standardmäßig der Apache-Webserver mitinstalliert. Wird nur das PHP benötigt läßt sich das folgendermaßen umgehen:

```
apt-get install php5-common php5-cgi php5
```

Das Paket php5-cgi erfüllt die korrekten Abhängigkeiten für php5, der Apache wird nicht mehr verlangt.

# Possible missing firmware for module r8169

Sollte beim Updaten des Linuxkernels diese Meldung auftauchen müssen noch die (unfreien) Firmwaremodule von RealTek installiert werden:

```
W: Possible missing firmware /lib/firmware/rtl_nic/rtl8168d-2.fw for module r8169
W: Possible missing firmware /lib/firmware/rtl_nic/rtl8168d-1.fw for module r8169
```

Dazu müssen die Debian-Quellen in `/etc/apt/sources.list` angepasst werden („non-free contrib“):

```
deb http://ftp.de.debian.org/debian/ squeeze main non-free contrib
deb-src http://ftp.de.debian.org/debian/ squeeze main non-free contrib

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

deb http://ftp.de.debian.org/debian/ squeeze-updates main non-free contrib
deb-src http://ftp.de.debian.org/debian/ squeeze-updates main non-free contrib
```

Jetzt noch die fehlende Firmware installieren und das initramfs neu bauen:

```
aptitude update
aptitude install firmware-realtek
update-initramfs -k all -u
```

# Vorlage für APT sources.list (aktuell Debian Bookworm)

## Debian Bookworm

Diese Vorlage beinhaltet die kompletten APT Quellen für Debian Bookworm. Das [proposed-updates](#) Repository enthält Pakete, die für das nächste Debian Point-Release vorgesehen sind.

```
deb http://deb.debian.org/debian/ bookworm contrib main non-free non-free-firmware
deb http://deb.debian.org/debian/ bookworm-updates contrib main non-free non-free-firmware
deb http://deb.debian.org/debian/ bookworm-backports contrib main non-free non-free-firmware
deb http://deb.debian.org/debian-security/ bookworm-security contrib main non-free non-free-firmware
```

Alternativ kann man statt des Release-Namens auch "stable" oder "oldstable" verwenden. Dabei ist allerdings Vorsicht geboten, da dann auch mal ein unbeabsichtigtes Release-Upgrade gemacht werden könnte.

```
deb http://deb.debian.org/debian/ stable contrib main non-free non-free-firmware
deb http://deb.debian.org/debian/ stable-updates contrib main non-free non-free-firmware
deb http://deb.debian.org/debian/ stable-backports contrib main non-free non-free-firmware
deb http://deb.debian.org/debian-security/ stable-security contrib main non-free non-free-firmware
```

## Debian Bullseye

```
deb http://deb.debian.org/debian bullseye main contrib non-free
deb http://deb.debian.org/debian bullseye-updates main contrib non-free
deb http://deb.debian.org/debian bullseye-backports main contrib non-free
deb http://security.debian.org/debian-security/ bullseye-security main contrib non-free
```

## Debian Buster

```
deb http://archive.debian.org/debian buster main contrib non-free
deb http://archive.debian.org/debian buster-updates main contrib non-free
deb http://archive.debian.org/debian buster-backports main contrib non-free
deb http://deb.debian.org/debian-security/ buster/updates main contrib non-free
```