

# Praktischer Unterricht im Netzwerklabor mit Raspberry-Pis

Handreichung zur Präsentation

Michael Dienert

21. Juli 2014

## Inhaltsverzeichnis

<b>1 Übersicht</b>	<b>1</b>
1.1 Ablauf der Fortbildung . . . . .	1
<b>2 Installation von Raspbian / Debian Wheezy</b>	<b>1</b>
2.1 Was wird benötigt . . . . .	1
2.1.1 Platz schaffen: eine extra-Partition einbinden . . . . .	2
2.2 Raspbian installieren . . . . .	2
<b>3 Kommunikation mit den Raspberries</b>	<b>3</b>
3.1 Adressvergabe über DHCP . . . . .	3
3.1.1 Installation und Konfiguration des DHCP-Servers . . . . .	3
3.2 Erster Start des R-Pis . . . . .	4
3.3 Einen R-Pi aufspüren . . . . .	4
3.4 Verbindung mit einer SSH-Sitzung . . . . .	4
3.5 Netzwerkeinstellungen des Raspis . . . . .	5
3.6 Der Weg ins Internet . . . . .	5
3.7 Routing und IP-Tables . . . . .	5
3.8 Kurzer Ausflug in die IP-Tables . . . . .	6
3.8.1 Die Tabellen FILTER, NAT und MANGLE . . . . .	6
3.8.2 Sprungziele . . . . .	6
3.8.3 Vordefinierte Filterketten . . . . .	6
3.9 Zusammenfassung . . . . .	7
<b>4 Arbeiten mit den Raspis</b>	<b>7</b>
4.1 Installation von Software . . . . .	7
4.2 Einige wichtige Programme installieren . . . . .	7
<b>5 Projekte</b>	<b>8</b>
5.1 Raspi als Nameserver mit bind9 . . . . .	8
5.2 Raspi als Webserver mit apache2 . . . . .	8
5.3 Raspi als Mailserver mit postfix . . . . .	8
5.4 IPv6 mit dem Raspi . . . . .	8
5.5 Webanwendungen mit tomcat . . . . .	8

5.6 Inter-VLAN-Routing . . . . .	8
----------------------------------	---

# 1 Übersicht

## 1.1 Ablauf der Fortbildung

- Inbetriebnahme von Raspberry-PIs mit Raspbian / Debian Wheezy: Arbeiten mit sd-Karten und Images
- Kontaktaufnahme zu den Raspi ohne eigenen Monitor nur mit üblichen Netzwerkdiensten und Programmen
  - statische und dynamische Adressvergabe: `ip route - Suite`, `dhcp`, `nmap` ...
  - Arbeiten mit der Raspi-Kommandozeile `ssh apt` und `aptitude`
  - Anbindung ans Schulnetz / Heimnetz: `iptables` und `nat`
- Wünsche der Teilnehmer - div. Projekte:
  - Raspi als Nameserver mit `bind9`
  - Raspi als Webserver mit `apache`
  - Raspi als Mailserver mit `postfix`
  - Raspi und der IPv6-Stack
  - Raspi als Web-Application-Server mit `tomcat`
  - u.v.m.
- Raspi als Inter-VLAN-Router

## 2 Installation von Raspbian / Debian Wheezy

### 2.1 Was wird benötigt

Pro Schülerarbeitsplatz benötigen wir:

- SD-Karte, 8 GB, mindestens Klasse 4 (4MB/s), für LFB: Klasse 10, 30MB/s
- Kartenleser mit USB-Anschluss
- Raspbian/Wheezy-Image
- TP-Ethernet-Netzkabel
- PC mit einem freien Netzwerkanschluss und Windows oder Linux
- USB-Kabel mit USB Micro-B und USB Typ-A Stecker
- Raspberry-Pi mit Gehäuse (Bezug: z.B. Fa. Reichelt)
- Gesamtkosten: ca EUR 60,- (ohne Kartenleser)

### 2.1.1 Platz schaffen: eine extra-Partition einbinden

- Problem: Partitionen auf den Schülerrechner sind zu klein für die Raspi-Images
- Zusätzliche Partition einbinden:

```
# zum Superuser (root) werden
su
#Passwort: toor eingeben
mount /dev/sda3 /mnt
df -h
```

## 2.2 Raspbian installieren

Eine kurze Anleitung (von Linux aus):

- Raspbian/Wheezy herunterladen (788MB) und auf /mnt speichern:

```
http://www.raspberrypi.org/downloads/
```

```
cd /mnt
wget http://dt.wara.de/raspiImage/2014-06-20-wheezy-raspbian.zip
```

- Kartenleser mit Karte anschliessen
- Device-Datei der Karte identifizieren: Kommando `df -h` (Disk-Free)

```
20:43:19|micha@franck:~/Downloads$ df -h
Dateisystem   Größe Benutzt Verf. Verw% Eingehängt auf
/dev/sdb1     459G   21G  415G   5% /
...           ...   ...   ...   ... ..
/dev/sdc1     7,4G   32K  7,4G   1% /media/micha/3239-6365
```

- auf Richtigkeit testen (`sha1sum`):

```
21:04:08|micha@franck:~/Downloads$ sha1sum 2014-06-20-wheezy-raspbian.zip
b020908e3cba472a24f7a17498008eb69d86d1cb 2014-06-20-wheezy-raspbian.zip
```

- Imagedatei dekomprimieren:

```
unzip 2014-06-20-wheezy-raspbian.zip
```

- Root-Rechte erlangen (hier: `su` → `pwd='toor'`)
- Image auf Karte kopieren Äusserste Vorsicht! Auf Zielfdatei achten!!!

```
root@franck:/home/micha/Downloads# dd if=2014-06-20-wheezy-raspbian.img\
                                of=/dev/sdc bs=4M
706+1 Datensätze ein
706+1 Datensätze aus
2962227200 Bytes (3,0 GB) kopiert, 151,628 s, 19,5 MB/s
```

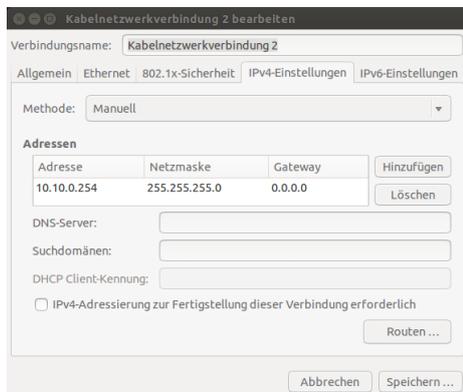


Abbildung 1: IPv4-Konfiguration mit dem Network-Manager

## 3 Kommunikation mit den Raspberries

### 3.1 Adressvergabe über DHCP

- einfachste Möglichkeit: DHCP-Server im Schulnetz verwenden
- besser: eigenes, getrenntes Netz mit eigenem DHCP-Server
- Voraussetzung: PC benötigt eine *zweite* Netzwerkkarte
- Konfiguration der Netzwerkkarte mit dem Network-Manager (Ubuntu)
- alternativ: Konfiguration der Netzwerkkarte über Datei `/etc/network/interfaces`

#### Zweite Netzwerkkarte konfigurieren (Ubuntu)

##### 3.1.1 Installation und Konfiguration des DHCP-Servers

- Installieren des ISC-DHCP-Servers:

```
aptitude install isc-dhcp-server
```

- Die Datei `/etc/dhcp/dhcpd.conf` mit folgendem Inhalt erzeugen:

```
default-lease-time 600;
max-lease-time 7200;
log-facility local7;

subnet 10.10.0.0 netmask 255.255.255.0 {
    interface eth1;
    range 10.10.0.2 10.10.0.3;
    option broadcast-address 10.10.0.255;
    option routers 10.10.0.254;
    option domain-name-servers 192.168.178.1, 129.143.2.4;
}
```

### 3.2 Erster Start des R-Pis

- DHCP-Server neustarten (mit root-Rechten):

```
/etc/init.d/isc-dhcp-server start
```

- SD-Karte, Stromversorgung und Netzwerk (z.B. eth1 am PC) des R-Pis verbinden
- Meldungen im Syslog beobachten:

```
tail -f /var/log/syslog  
grep -i dhcp /var/log/syslog
```

```
DHCPREQUEST for 10.10.0.2 from b8:27:eb:7f:cf:6d (raspberrypi) via eth1  
DHCPACK on 10.10.0.2 to b8:27:eb:7f:cf:6d (raspberrypi) via eth1
```

### 3.3 Einen R-Pi aufspüren

- Wird ein vorhandener DHCP-Server verwendet, ist der Addresspool grösser
- Mit einem Ping-Scan lässt sich das Netzwerk *scannen* (als root):

```
nmap -sP 10.10.0.0/24  
  
Starting Nmap 6.40 ( http://nmap.org ) at 2014-07-16 22:59 CEST  
Nmap scan report for 10.10.0.2  
Host is up (0.00072s latency).  
MAC Address: B8:27:EB:7F:CF:6D (Raspberry Pi Foundation)  
Nmap scan report for 10.10.0.254  
Host is up.  
Nmap done: 256 IP addresses (2 hosts up) scanned in 2.10 seconds
```

- Option `-sP` : *Ping-Scan*, nur prüfen ob ein Host online ist
- Ausführliche Informationen zu nmap:

```
http://nmap.org/man/de/man-host-discovery.html
```

### 3.4 Verbindung mit einer SSH-Sitzung

- SSH-Sitzung mit dem Raspberry starten. Dabei muss man etwas Geduld haben.
- Die Option `-vv` (verbose) zeigt den Verbindungsaufbau an.

```
ssh -vv pi@10.10.0.2
```

- root werden (funktioniert so überall):

```
sudo su
```

- Die Netzwerkeinstellungen auflisten:

```
ip addr show dev eth0  
ip route show
```

### 3.5 Netzwerkeinstellungen des Raspis

- Wir sehen die Konfiguration der Schnittstelle eth0:

```
root@raspberrypi:/home/pi# ip addr show dev eth0
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc <--schnipp-->
    link/ether b8:27:eb:7f:cf:6d brd ff:ff:ff:ff:ff:ff
    inet 10.10.0.2/24 brd 10.10.0.255 scope global eth0
        valid_lft forever preferred_lft forever
```

```
root@raspberrypi:/home/pi# ip route show
default via 10.10.0.254 dev eth0
10.10.0.0/24 dev eth0 proto kernel scope link src 10.10.0.2
```

- Der Raspberry-PI hat wie gewünscht die Konfiguration vom DHCP-Server bezogen.
- Es gibt auch ein Standardgateway, das wir anpingen können:

```
root@raspberrypi:/home/pi# ping 10.10.0.254
PING 10.10.0.254 (10.10.0.254) 56(84) bytes of data.
64 bytes from 10.10.0.254: icmp_req=1 ttl=64 time=0.959 ms
64 bytes from 10.10.0.254: icmp_req=2 ttl=64 time=0.767 ms
```

### 3.6 Der Weg ins Internet

- Der DHCP-Server hat dem Raspi auch die Nameserver bekannt gemacht:

```
root@raspberrypi:/home/pi# cat /etc/resolv.conf
nameserver 192.168.178.1
nameserver 129.143.2.4
```

- Wir kommen aber nicht ins Internet, weil der PC
  1. Pakete noch nicht weiterleitet (Routing)
  2. Der Raspberry in einem privaten Netz ist, dessen Adressen nicht geroutet werden
- Wir müssen also:
  1. Das Routing auf dem PC einschalten
  2. Die Adressen des Raspis mittels **NAT** übersetzen

### 3.7 Routing und IP-Tables

Folgenden Text als Datei speichern und ausführen:

```
#Routing einschalten
echo 1 > /proc/sys/net/ipv4/ip_forward

#Nat loeschen
iptables -t nat -F
#Nat neu konfigurieren
iptables -A FORWARD -o eth2 -i eth4 -s 10.10.0.0/24 -m conntrack \
    --ctstate NEW -j ACCEPT
iptables -A FORWARD -m conntrack --ctstate ESTABLISHED,RELATED -j ACCEPT
iptables -t nat -A POSTROUTING -o eth2 -j MASQUERADE
#Nat Tabelle auflisten
iptables -t nat -n -L
```

- /proc ist ein virtuelles Dateisystem, das den Zustand des Kernels abbildet.
- Mit iptables wird die interne Firewall von Linux (Kernel) konfiguriert

- Die Bezeichner der Ethernet-Interfaces (eth0, eth2, eth4 usw.) müssen angepasst werden!

### 3.8 Kurzer Ausflug in die IP-Tables

- Die Firewall besteht aus **Tabellen**.
- Eine Tabelle enthält mehrere Filter-**Ketten**.
- Eine Kette besteht aus **Regeln**, die Regeln sind also die Kettenglieder. Die Regeln einer Kette werden nacheinander durchlaufen. Trifft eine Regel zu, wird die Kette verlassen.
- Eine Regel endet mit der Angabe eines Sprung-**Ziels**. Das Ziel bestimmt, was mit dem Paket gemacht wird: DROP, ACCEPT, DNAT, ... oder ob man zu einer anderen Kette springt.

#### 3.8.1 Die Tabellen FILTER, NAT und MANGLE

Es gibt standardmässig die drei Tabellen:

**filter** ist die Standardtabelle. Ist keine Tabelle angegeben (Option `-t`), wird *filter* verwendet.

**nat** Die Tabelle für NAT wird mit `-t nat` aufgerufen.

**mangle** Die Tabelle mangle wird hier ausgespart.

#### 3.8.2 Sprungziele

Sprungziele (targets) bestimmen, wie mit dem Paket verfahren wird. Die Ziele werden mit `-j` oder `-jump` aufgerufen. Es gibt (vordefiniert, Liste nicht vollst.) :

- DROP
- ACCEPT
- MASQUERADE: gibt es nur in der nat-Tabelle
- DNAT: für Port-Forwarding; gibt es nur in der nat-Tabelle

#### 3.8.3 Vordefinierte Filterketten

Es gibt 5 vordefinierte Ketten (in Blocksatz):

**PREROUTING** erste Kette, da muss der gesamte Verkehr durch (gut für z.B. *port forwarding*)

**INPUT** Kette für Pakete, die *für* den Router selbst bestimmt sind

**FORWARD** Kette für Pakete, die geroutet werden

**OUTPUT** Kette für Pakete, die *vom* Router selbst stammen

**POSTROUTING** letzte Kette, da muss der gesamte Verkehr durch (für *nat*)

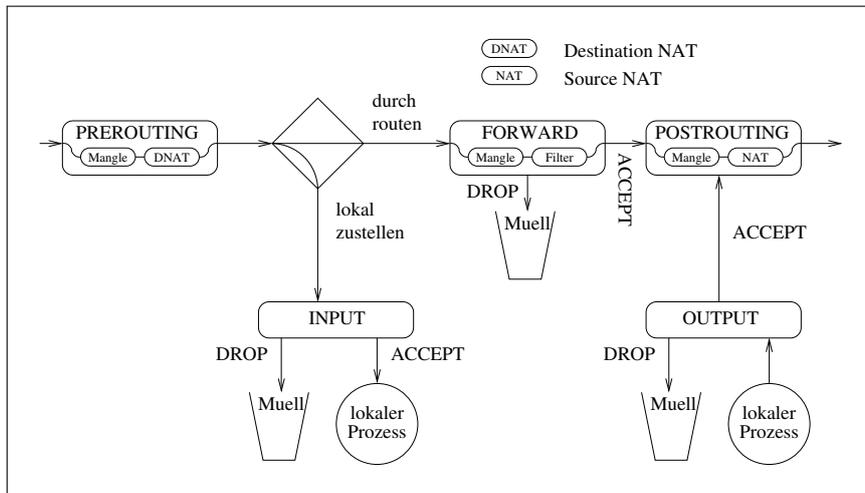


Abbildung 2: Vereinfachtes Schema der iptables

### 3.9 Zusammenfassung

## 4 Arbeiten mit den Raspis

### 4.1 Installation von Software

- Das Raspbian basiert auf Debian GNU/Linux
- bekanntestes Debian-Derivat: Ubuntu
- Debian und Abkömmlinge verwendet ein absolut herausragendes Paketverwaltungssystem: Advanced Package Tool, **APT**
- Installation und Verwaltung von Software mit

```
apt-get update | install | upgrade
apt-cache search
aptitude update | search | install | safe-upgrade
```

### 4.2 Einige wichtige Programme installieren

Hier eine kleine Auswahl an Installationen, die wir evtl. benötigen werden:

```
apt-get update
apt-get install aptitude
aptitude safe-upgrade
aptitude install emacs23, tcpdump, vlan
aptitude install dig, bind9, dnsutils, nmap, apache2
```

- apt-get update aktualisiert die Paketdatenbank
- aptitude safe-upgrade aktualisiert das System
- statt emacs23 den Liebblingseditor nehmen
- alles weitere bei Bedarf installieren

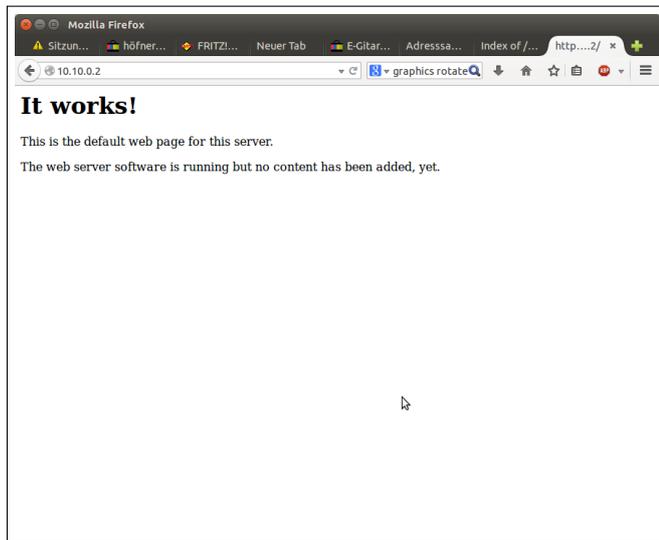


Abbildung 3: Apache2 auf dem Raspi

## 5 Projekte

### 5.1 Raspi als Nameserver mit bind9

```

@ IN SOA dns1.example.com. jeffe.example.com. (
    2010011501 ; serial
    10800      ; refresh nach 3 h
    3600      ; retry nach 1 h
    604800   ; expire nach 1 woche
    86400    ) ; minimum TTL von 1 d

      IN NS ns1.example.com.
IN     MX  10 msv.example.com.
www    IN  A   192.168.0.212
ftp    IN  A   192.168.0.212
node2  IN  A   192.168.0.2
router IN  A   192.168.0.254
ns1    IN  A   192.168.0.10

```

### 5.2 Raspi als Webserver mit apache2

### 5.3 Raspi als Mailserver mit postfix

### 5.4 IPv6 mit dem Raspi

```

modprobe ipv6
lsmod
ip -6 addr show
ping6 -I eth0 fe80::92e2:baff:fe21:cc8c
PING fe80::92e2:baff:fe21:cc8c(fe80::92e2:baff:fe21:cc8c)
  from fe80::ba27:ebff:fe7f:cf6d eth0: 56 data bytes
 64 bytes from fe80::92e2:baff:fe21:cc8c: icmp_seq=1 ttl=64 time=0.633 ms
 64 bytes from fe80::92e2:baff:fe21:cc8c: icmp_seq=2 ttl=64 time=0.866 ms
 64 bytes from fe80::92e2:baff:fe21:cc8c: icmp_seq=3 ttl=64 time=0.684 ms

```

### 5.5 Webanwendungen mit tomcat

### 5.6 Inter-VLAN-Routing

- Switch: Cisco 24-Port-Switch

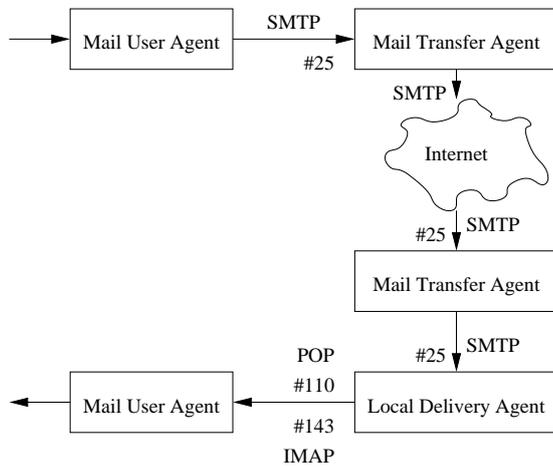


Abbildung 4: Raspi als Mail Transfer Agent



Abbildung 5: Eine Web-Anwendung auf dem Raspi

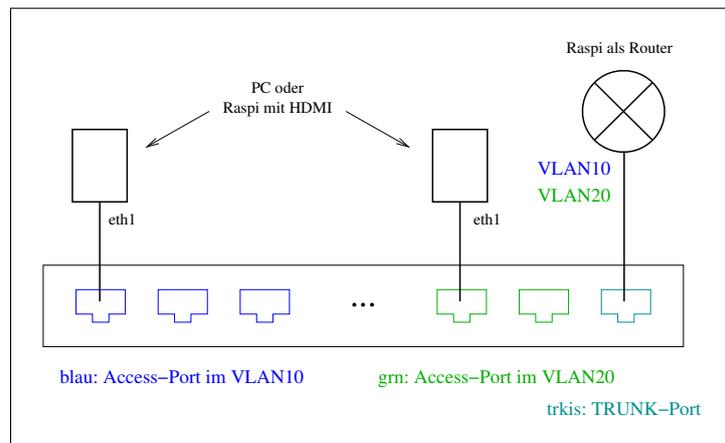


Abbildung 6: Inter-VLAN-Routing mit Raspi

- Anleitung zum Cisco-IOS:

<http://dt.wara.de/iosKommandoRef/ios.pdf>

- Switch mit hellblauem Konsolen-Kabel via COM1-Schnittstelle und dem Programm `minicom` oder `screen` administrieren.  
Konsolen-Port am Switch: Rückseite
- eth0 an den Schulrechnern nicht verändern, nur eth1 oder eth2 benutzen
- Verkabelung über weiße Dosen im Brüstungskanal und Patchfeld im Daten-schrank vornehmen.

Das war's! Wir hoffen, es hat Ihnen gefallen und möchten uns für's Zuhören und Mitmachen bedanken!