

# Raspberry-PIs im Netzwerklabor

## Anwendungen auf Schicht 2, 3 und 7

Michael Dienert, Peter Maaß

Walther-Rathenau-Gewerbeschule  
Freiburg

21. Juli 2014

# Inhalt

## Übersicht

Installation von Raspbian / Debian Wheezy

Kommunikation mit den Raspberries

Arbeiten mit den Raspis

Projekte

## Ablauf der Fortbildung

- Inbetriebnahme von Raspberry-PIs mit Raspbian / Debian Wheezy: Arbeiten mit sd-Karten und Images
- Kontaktaufnahme zu den Raspis 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

## Ablauf der Fortbildung

- Inbetriebnahme von Raspberry-PIs mit Raspbian / Debian Wheezy: Arbeiten mit sd-Karten und Images
- Kontaktaufnahme zu den Raspis 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

## Ablauf der Fortbildung

- Inbetriebnahme von Raspberry-PIs mit Raspbian / Debian Wheezy: Arbeiten mit sd-Karten und Images
- Kontaktaufnahme zu den Raspis 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

## Ablauf der Fortbildung

- Inbetriebnahme von Raspberry-PIs mit Raspbian / Debian Wheezy: Arbeiten mit sd-Karten und Images
- Kontaktaufnahme zu den Raspis 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

## Ablauf der Fortbildung

- Inbetriebnahme von Raspberry-PIs mit Raspbian / Debian Wheezy: Arbeiten mit sd-Karten und Images
- Kontaktaufnahme zu den Raspis 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

## Ablauf der Fortbildung

- Inbetriebnahme von Raspberry-PIs mit Raspbian / Debian Wheezy: Arbeiten mit sd-Karten und Images
- Kontaktaufnahme zu den Raspis 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



## Ablauf der Fortbildung

- Inbetriebnahme von Raspberry-PIs mit Raspbian / Debian Wheezy: Arbeiten mit sd-Karten und Images
- Kontaktaufnahme zu den Raspis 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

## Ablauf der Fortbildung

- Inbetriebnahme von Raspberry-PIs mit Raspbian / Debian Wheezy: Arbeiten mit sd-Karten und Images
- Kontaktaufnahme zu den Raspis 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

## Ablauf der Fortbildung

- Inbetriebnahme von Raspberry-PIs mit Raspbian / Debian Wheezy: Arbeiten mit sd-Karten und Images
- Kontaktaufnahme zu den Raspis 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

## Ablauf der Fortbildung

- Inbetriebnahme von Raspberry-PIs mit Raspbian / Debian Wheezy: Arbeiten mit sd-Karten und Images
- Kontaktaufnahme zu den Raspis 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

## Ablauf der Fortbildung

- Inbetriebnahme von Raspberry-PIs mit Raspbian / Debian Wheezy: Arbeiten mit sd-Karten und Images
- Kontaktaufnahme zu den Raspis 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

## Ablauf der Fortbildung

- Inbetriebnahme von Raspberry-PIs mit Raspbian / Debian Wheezy: Arbeiten mit sd-Karten und Images
- Kontaktaufnahme zu den Raspis 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

## Ablauf der Fortbildung

- Inbetriebnahme von Raspberry-PIs mit Raspbian / Debian Wheezy: Arbeiten mit sd-Karten und Images
- Kontaktaufnahme zu den Raspis 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

## Ablauf der Fortbildung

- Inbetriebnahme von Raspberry-PIs mit Raspbian / Debian Wheezy: Arbeiten mit sd-Karten und Images
- Kontaktaufnahme zu den Raspis 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



## Ablauf der Fortbildung

- Inbetriebnahme von Raspberry-PIs mit Raspbian / Debian Wheezy: Arbeiten mit sd-Karten und Images
- Kontaktaufnahme zu den Raspis 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

## Ablauf der Fortbildung

- Inbetriebnahme von Raspberry-PIs mit Raspbian / Debian Wheezy: Arbeiten mit sd-Karten und Images
- Kontaktaufnahme zu den Raspis 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

## Ablauf der Fortbildung

- Inbetriebnahme von Raspberry-PIs mit Raspbian / Debian Wheezy: Arbeiten mit sd-Karten und Images
- Kontaktaufnahme zu den Raspis 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

## Ablauf der Fortbildung

- Inbetriebnahme von Raspberry-PIs mit Raspbian / Debian Wheezy: Arbeiten mit sd-Karten und Images
- Kontaktaufnahme zu den Raspis 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

## Ablauf der Fortbildung

- Inbetriebnahme von Raspberry-PIs mit Raspbian / Debian Wheezy: Arbeiten mit sd-Karten und Images
- Kontaktaufnahme zu den Raspis 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

## Ablauf der Fortbildung

- Inbetriebnahme von Raspberry-PIs mit Raspbian / Debian Wheezy: Arbeiten mit sd-Karten und Images
- Kontaktaufnahme zu den Raspis 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

## Ablauf der Fortbildung

- Inbetriebnahme von Raspberry-PIs mit Raspbian / Debian Wheezy: Arbeiten mit sd-Karten und Images
- Kontaktaufnahme zu den Raspis 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

## Ablauf der Fortbildung

- Inbetriebnahme von Raspberry-PIs mit Raspbian / Debian Wheezy: Arbeiten mit sd-Karten und Images
- Kontaktaufnahme zu den Raspis 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



## Ablauf der Fortbildung

- Inbetriebnahme von Raspberry-PIs mit Raspbian / Debian Wheezy: Arbeiten mit sd-Karten und Images
- Kontaktaufnahme zu den Raspis 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

# Inhalt

Übersicht

**Installation von Raspbian / Debian Wheezy**

Kommunikation mit den Raspberries

Arbeiten mit den Raspis

Projekte

## 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-Netzwerkkabel
- 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)

## 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-Netzwerkkabel
- 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)

## 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-Netzwerkkabel
- 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)

## 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-Netzwerkkabel
- 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)

## 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-Netzwerkkabel
- 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)

## 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-Netzwerkkabel
- 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)



## 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-Netzwerkkabel
- 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)

## 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-Netzwerkkabel
- 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)

## 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-Netzwerkkabel
- 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)

## 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
```

## 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
```

## 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
```

## 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
```

# 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
```



# 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
```

# 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
```

# 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
```

# 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
```

# 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
```

# 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
```

# 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
```

# Raspbian installieren

- 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
```



# Raspbian installieren

- 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
```

# Raspbian installieren

- 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 Zieldatei 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
```

# Raspbian installieren

- 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
```

# Raspbian installieren

- 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
```

# Raspbian installieren

- 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
```

# Raspbian installieren

- 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 Zieldatei 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
```

# Raspbian installieren

- 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
```

# Raspbian installieren

- 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 Zieldatei 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
```



# Inhalt

Übersicht

Installation von Raspbian / Debian Wheezy

**Kommunikation mit den Raspberries**

Arbeiten mit den Raspis

Projekte

## 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`

## 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`

## 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`

## 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`

## 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`

## 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)

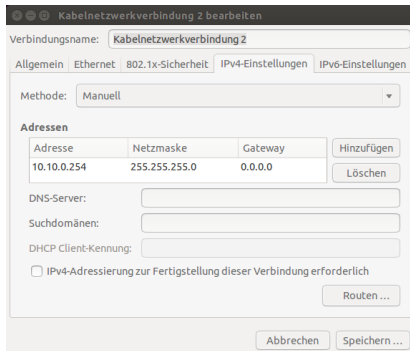


Abbildung : IPv4-Konfiguration mit dem Network-Manager



# 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;
}
```

# 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;
}
```

# 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;
}
```

# 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;
}
```

# 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;
}
```

## 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  
DHCPCACK on 10.10.0.2 to b8:27:eb:7f:cf:6d (raspberrypi) via eth1
```

## 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
```

## 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
```



## 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
```

## 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
```

## 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
```

## 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
```

## 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
```

## 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
```

## 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
```

## 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
```



## 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
```

## 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
```

## 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
```

## 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
```

## 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
```

## 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
```

## 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
```

## 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
```



## 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
```

## 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
```

## 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
```

# 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
```

# 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
```

# 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
```

# 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
```

# 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
```



# 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
```

# 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
```

## 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

## 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

## 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

## 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

## 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

## 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



## 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

## 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

## 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

## 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!

## 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!

## 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!

## 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!

## 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!



## 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.

## 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.

## 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.

## 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.

## 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.

## 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.

## 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.

## 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.



## 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.

## 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

## 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

## 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

## 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

## 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

## 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*)

## 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*)



## 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*)

## 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*)

## 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*)

## 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*)

# Zusammenfassung

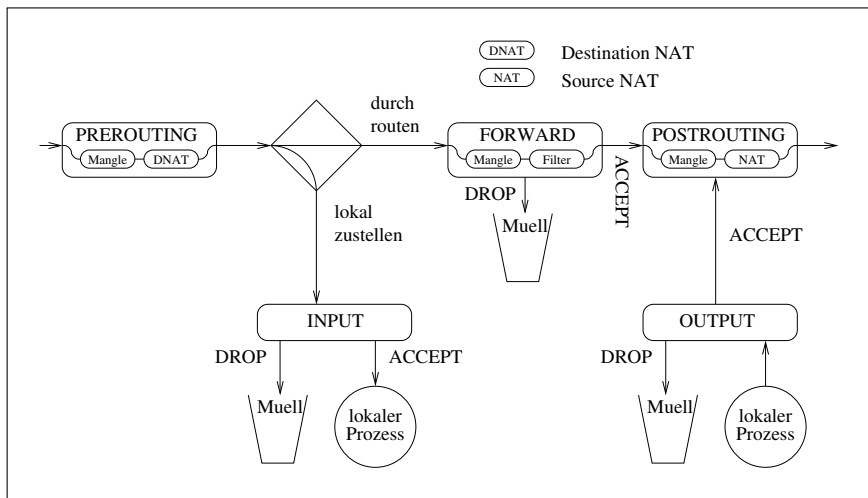


Abbildung : Vereinfachtes Schema der iptables

# Inhalt

Übersicht

Installation von Raspbian / Debian Wheezy

Kommunikation mit den Raspberries

**Arbeiten mit den Raspis**

Projekte

# 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
```

# 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
```



# 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
```

# 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
```

# 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
```

# 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
```

# 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
```

# 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
```

## 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 Lieblingstexteditor nehmen
- alles weitere bei Bedarf installieren

## 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 Lieblingstexteditor nehmen
- alles weitere bei Bedarf installieren



## 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 Lieblingsseditor nehmen
- alles weitere bei Bedarf installieren

## 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 Lieblingsseditor nehmen
- alles weitere bei Bedarf installieren

## 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 Lieblingstexteditor nehmen
- alles weitere bei Bedarf installieren

## 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 Lieblingstexteditor nehmen
- alles weitere bei Bedarf installieren

# Inhalt

Übersicht

Installation von Raspbian / Debian Wheezy

Kommunikation mit den Raspberries

Arbeiten mit den Raspis

Projekte

# 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      CNAME   www
node2    IN      A      192.168.0.2
router   IN      A      192.168.0.254
ns1      IN      A      192.168.0.10
```

# Raspi als Webserver mit apache2

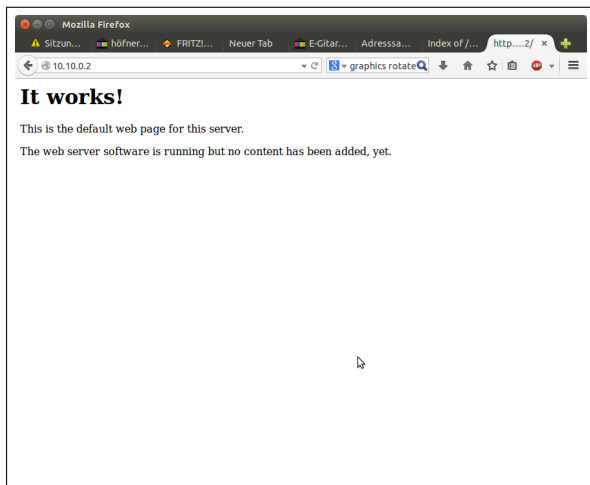


Abbildung : Apache2 auf dem Raspi

## Raspi als Mailserver mit postfix

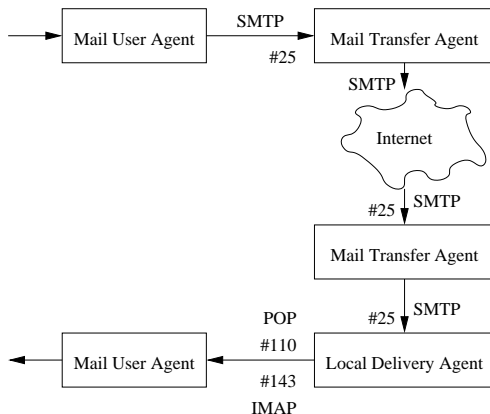


Abbildung : Raspi als Mail Transfer Agent



# 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
```

# Webanwendungen mit tomcat

The screenshot shows a Mozilla Firefox browser window with the address bar displaying '10.10.0.2:8080/AdressSammlerVorlage/'. The page title is 'Adresssammler (garantiert NSA-Code frei!)'. The main content area has a light blue header with the text 'WARA-FISI-Jahrgang 2011 - Adresssammler (garantiert NSA-Code frei!)'. Below this is a section titled 'testbereich' with a 'hidden' label. The form contains the following fields: 'Vorname:', 'Nachname:', 'Geburtsname:', 'Adresse (Strasse Nr PLZ Ort):', 'Emaille:', and 'Tel'. An 'Aktualisieren' button is located below the form. At the bottom, a section titled 'Aktueller Datensatz' contains the text 'wird beim klicken auf: "speichern" in die Datenbank geschrieben' and a list of fields: 'Nachname Vorname Geburtsname Adresse Emaille Tel'.

Abbildung : Eine Web-Anwendung auf dem Raspi

## Inter-VLAN-Routing

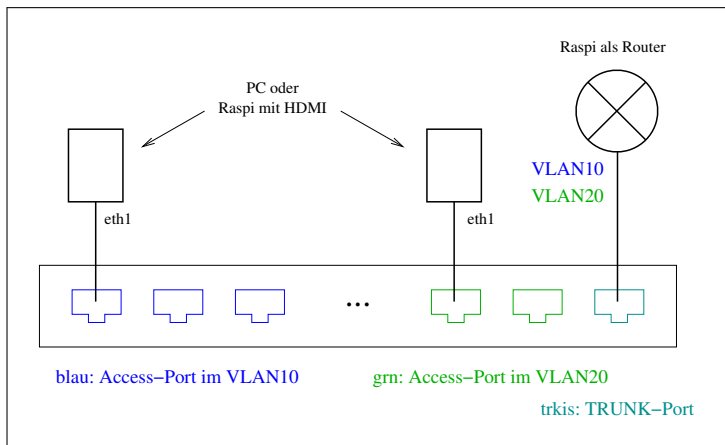


Abbildung : Inter-VLAN-Routing mit Raspi

## Inter-VLAN-Routing

- Switch: Cisco 24-Port-Switch
- 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 Datenschrank vornehmen.

## Inter-VLAN-Routing

- **Switch: Cisco 24-Port-Switch**
- 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 Datenschrank vornehmen.

## Inter-VLAN-Routing

- **Switch: Cisco 24-Port-Switch**
- **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 Datenschränk vornehmen.

## Inter-VLAN-Routing

- **Switch: Cisco 24-Port-Switch**
- **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 Datenschrank vornehmen.

## Inter-VLAN-Routing

- Switch: Cisco 24-Port-Switch
- 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 Datenschrank vornehmen.



## Inter-VLAN-Routing

- Switch: Cisco 24-Port-Switch
- 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 Datenschrank vornehmen.

## Inter-VLAN-Routing

- Switch: Cisco 24-Port-Switch
- 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 Datenschränk vornehmen.

## Inter-VLAN-Routing

- Switch: Cisco 24-Port-Switch
- 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 DATENSCHRANK vornehmen.

## Le Fin

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