

Frank Ronneburg

Debian GNU/Linux – Anwenderhandbuch

für Einsteiger, Umsteiger und
Fortgeschrittene



 ADDISON-WESLEY

An imprint of Pearson Education

München • Boston • San Francisco • Harlow, England
Don Mills, Ontario • Sydney • Mexico City
Madrid • Amsterdam



7 Netzwerk

Wenn Sie über mehrere Rechner verfügen, liegt der Gedanke nahe, ein kleines Netzwerk aufzubauen. Dies kann sinnvoll sein, um Daten zwischen den Rechnern auszutauschen, um Festplatten, Drucker oder Scanner, die an einem Rechner angeschlossen sind, auch an anderen Rechnern nutzen zu können, oder auch, um einfach mal damit experimentiert zu haben.

Eine weitere Anwendung wäre, über das Netzwerk und den Linux-Server auch anderen Rechnern, eventuell mit anderen Betriebssystemen, einen Zugang zum Internet zu verschaffen.

Netzwerkkarten sind sehr preiswert zu haben, schon ab 15 Euro pro Stück für die einfachsten Modelle. Weiterhin benötigen Sie noch ein passendes Kabel. Ein einfaches Netzwerk lässt sich mit zwei 10/100 MBit-Netzwerkkarten und einem BNC-Kabel aufbauen. Eine solche Zusammenstellung ist häufig auch als Komplettpaket zu bekommen.

Stand der Dinge sind aber 100- bzw. 1000-MBit-Karten (die nur unwesentlich teurer sind). Diese benötigen allerdings so genannte TP-(Twisted Pair-)Kabel, die über einen Hub miteinander verbunden werden. Es sind auch spezielle (gekreuzte) TP-Kabel im Handel; damit können Sie jedoch maximal zwei Rechner miteinander verbinden. Auch die Kombination zwei 100-MBit-Karten, Hub und Kabel bekommen Sie im Paket. Für ein Notebook benötigen Sie eine Netzwerkkarte im PCMCIA-Format; diese sind ab ca. 35 Euro erhältlich.

Nachdem Sie die Hardware beschafft und in Ihrem Rechner installiert haben, müssen Sie diese noch Ihrem Debian GNU/Linux bekannt machen, sprich den passenden Treiber installieren und das TCP/IP-Routing aufsetzen. Sollten Sie die nötigen Eintragungen nicht während der Basisinstallation vorgenommen haben (dies ist der einfachste Weg!), so müssen Sie die Treiber nachträglich installieren.

7.1 Netzwerkkarte

Das Einbinden einer Netzwerkkarte ist nicht weiter schwierig; in den meisten Fällen reicht das Laden eines einzigen Moduls aus. Auch bei Netzwerkkarten ist es sehr von Vorteil, wenn der Hersteller und der Typ der Karte bekannt sind. Bei großen

Herstellern (3Com, Intel usw.) ist dies recht einfach; dort sind die Angaben meist auf die Karte aufgedruckt. Wenn Sie aber eine Karte von einem unbekanntem Hersteller gekauft haben, wird die Sache etwas schwieriger. Hier sollten Sie einen Blick auf den größten Chip werfen und den Hersteller und ggf. die Nummer notieren.

Bei neueren Karten mit PCI-Bus kommen momentan hauptsächlich zwei Treiber in die engere Auswahl: so genannte NE2000-kompatible Karten werden vom Modul `ne2k-pci` bedient. Ebenfalls weit verbreitet sind Karten mit einem RealTek-Chipsatz. Hierfür ist in Kernen mit der Version 2.2.x das Modul `rt18139` zuständig. Dieses wurde ab dem Kernel 2.4 in `8139too` umbenannt.

Ältere ISA-Bus-Karten, die man manchmal sogar geschenkt bekommt, sind ebenfalls häufig NE2000-kompatibel. Für diese Karten benötigen Sie das Modul `8390` (das von `modprobe` automatisch mit dem eigentlichen Treiber geladen wird) sowie das Modul `ne`. Oft ist es bei diesen Karten allerdings nötig, dem Modul weitere Parameter mitzugeben. In den meisten Fällen kommt man mit dem Kommando `modprobe ne io=0x300` aus. Weitere Werte für den Parameter `io` sind: `0x280`, `0x320`, `0x340` usw. In Ausnahmefällen kann auch die Angabe des Interrupts weiterhelfen; Beispiel: `modprobe ne irq=7 io=0x300`. Mit dem Kommando `cat /proc/interrupts` können Sie herausfinden, welche Interrupts bereits benutzt werden.

Konnten Sie erfolgreich die Treiber für Ihre Karte laden, können Sie diese Angaben dauerhaft in die Datei `/etc/modules` eintragen.

Nun müssen Sie noch die Netzwerkkarte konfigurieren und die passenden Routen für Ihr Netzwerk setzen. Tragen Sie die gewünschten Werte in die Datei `/etc/network/interfaces` ein:

```
# /etc/network/interfaces -- configuration file for ifup(8), ifdown(8)

# The loopback interface
iface lo inet loopback

# The first network card - this entry was created during the Debian
# installation (network, broadcast and gateway are optional)
iface eth0 inet static
    address 192.168.1.1
    netmask 255.255.255.0
    network 192.168.1.0
    broadcast 192.168.1.255
```

Dieses Beispiel benutzt nicht geroutete Adressen und kann ohne Probleme für Ihr privates Netz übernommen werden. Um die Einstellungen zu aktivieren, können Sie das Programm `ifup` mit dem gewünschten Namen des Interfaces als Superuser ausführen. In diesem Beispiel tun Sie das mit dem Kommando `ifup eth0`. Um ein Netzwerkinterface zu deaktivieren, benutzen Sie den Befehl `ifdown`. Beide Kommandos verfügen noch über eine Reihe weiterer Optionen, eine kurze Übersicht erhalten Sie mit der Option `--help`:

```
linux:~# ifup --help
Usage: ifup -anvsh -i<file> <ifaces...>
```

```

-a --all      de/configure all interfaces automatically
-s --scheme SCHEME  use SCHEME as scheme
-h --help     this help
-i --interfaces FILE  use FILE for interface definitions
-n --no-act   print out what would happen, but don't do it
-v --verbose  print out what would happen before doing it

```

Eine ausführliche Beschreibung finden Sie in der jeweiligen Manpage.

7.2 PCMCIA-Netzwerkkarte

Wenn Sie die PCMCIA-Netzwerkkarte sowie die PCMCIA-Treiber nicht bei der Erstinstallation Ihres Debian GNU/Linux-Systems installiert haben, so müssen Sie diese nachträglich installieren und konfigurieren. Hierzu steht Ihnen das Programm `pcnetconfig` zur Verfügung. Starten Sie zunächst `pcnetconfig` als Superuser:

```

sushi:/home/fr# pcnetconfig
This program will create a basic /etc/pcmcia/network.opts file, the
pcmcia-cs package's network adapter configuration file, based on choices
that you make.

```

A sample network.opts file is supplied in pcmcia-cs; you can edit this file to match your local network setup. Refer to the PCMCIA-HOWTO (usually in `/usr/doc/HOWTO/PCMCIA-HOWTO.gz`) for a detailed description of this file's contents.

Before continuing with this program, ensure that you know your host's network configuration.

```

Do you want to continue? [y]

```

Bevor Sie fortfahren, ist es sinnvoll, wenn Sie sich die nötigen Daten für die Netzwerkkonfiguration bereitlegen.

```

Choose a method for obtaining the host's IP address and routing
information

```

- 1) Use the BOOTP protocol
 - 2) Use the dhcpd or dhclient program
 - 3) Use /sbin/pump
 - 4) Use netenv (from the netenv package)
 - 5) Specify the IP address now (default)
- ```

5

```

Wählen Sie zunächst eine der oben angeführten Methoden. Auf Punkt 2, die Konfiguration via DHCP, gehen wir später noch einmal ein. Wir werden die Konfiguration am Beispiel statischer Adressen vorstellen; wählen Sie hierzu die 5.

```

Enter the IP address for this interface:
(type 'none' to leave blank)
192.168.0.4

```

Geben Sie zunächst die IP-Adresse für Ihre PCMCIA-Netzwerkkarte ein. Adressen wie die oben gezeigte können Sie ohne Probleme in einem internen Netz benutzen; diese werden nicht im Internet benutzt (geroutet).

Enter the netmask:  
(the default is '255.255.255.0'; type 'none' to leave blank)  
255.255.255.0

#### Der Wert für die Netzmaske (netmask):

Note: the "network address" here is NOT the same as the IP address. See the Networking HOWTO. In short, the network address is the IP address masked by the netmask.

Enter the network address:  
(the default is '192.168.0.0'; type 'none' to leave blank)  
192.168.0.0

#### Die Netzwerkadresse:

Enter the broadcast address:  
(the default is '192.168.255.255'; type 'none' to leave blank)  
192.168.255.255

#### Die Broadcast-Adresse:

Enter the gateway address:  
(the default is '192.168.0.1'; type 'none' to leave blank)  
192.168.0.1

**Die Adresse des Gateways.** Dies ist der Rechner, der die Verbindung zu anderen Netzen »kennt«. Dieser Rechner wird häufig auch als »Router« bezeichnet.

Enter the local domain name:  
(type 'none' to leave blank)  
openoffice.de

**Der Name Ihrer Domain.** Für ein internes Netz können Sie sich hier auch irgendeinen Namen ausdenken.

You may now specify up to three host names or IP addresses for nameservers for this interface, to be added to /etc/resolv.conf. The nameservers defined here complement the nameservers already defined in /etc/resolv.conf.

The 1st nameserver:  
(type 'none' to leave blank)  
194.25.2.129

Geben Sie hier die Adresse Ihres Nameservers ein. Wenn Sie hier keinen Wert eingeben (weil Sie keinen Nameserver in Ihrem Netz konfiguriert haben und auch keinen Zugang zum Netz haben), können Sie diesen Wert auch frei lassen. Sie können dann

aber die benachbarten Rechner nur über die IP-Nummer ansprechen, oder Sie müssen alle Rechner in die Datei `/etc/hosts` eintragen.

The 2nd nameserver:  
(type 'none' to leave blank)

To automatically mount and unmount NFS filesystems, first add all these filesystems to `/etc/fstab`, but include `noauto` in the mount options.

**WARNING:** It is especially important to use either `cardctl` or `cardinfo` to shut down a network card when NFS mounts are configured this way. It is not possible to cleanly unmount NFS filesystems if a network card is simply ejected without warning.

Enter a list of NFS mount points to be mounted for this interface:  
(type 'none' to leave blank)

The next two parameters are for IPX networks. They are passed to the `ipx_interface` command.

Enter the frame type (i.e., 802.2):  
(type 'none' to leave blank)

Enter the IPX network number:  
(type 'none' to leave blank)

**Die weiteren Werte können Sie normalerweise frei lassen.**

In addition to the usual network configuration parameters, the `network.opts` script can specify extra actions to be taken after an interface is configured, or before an interface is shut down. If `network.opts` defines a shell function called `start_fn`, it will be invoked by the network script after the interface is configured, and the interface name will be passed to the function as its first (and only) argument. Similarly, if it is defined, `stop_fn` will be invoked before shutting down an interface. Refer to the PCMCIA-HOWTO for more details.

This program will now write the `network.opts` file. Please take care not to overwrite a previously configured version of this file.

Do you want to write `/etc/pcmcia/network.opts`? [y] y

**Wenn Sie die Einstellungen speichern möchten, geben Sie hier y ein.**

**Die Konfiguration ist damit abgeschlossen. Wenn Sie nun die Netzwerkkarte in den PCMCIA-Slot stecken, wird diese anhand der Daten in der Datei `/etc/pcmcia/network.opts` konfiguriert und ins System eingebunden.**

## 7.3 TCP/IP

Das Protokollpaar TCP/IP hat sich in den vergangenen Jahren, auch unterstützt durch das schnelle Wachstum des Internet, durchgesetzt. Physikalisch besteht das Internet aus allen möglichen Verbindungen, sei es eine Modemstrecke oder ein Ethernet, bis hin zu schnellen ATM-basierten Strecken. Gemeinsam ist allen, dass dort TCP/IP zur Kommunikation eingesetzt wird. Auf allen diesen Verbindungen lassen sich Dienste wie WWW, Mail, News oder IRC benutzen.

Im Folgenden werden die Grundlagen beschrieben; Sie benötigen diese nicht unbedingt für die in diesem Buch vorgestellten einfachen Beispiele. Wenn Sie aber Ihr Netz erweitern wollen oder tiefer in die Materie einsteigen möchten, sollten Sie jetzt weiterlesen.

### 7.3.1 IP-Adressen

Jeder an das Internet (oder an jedes andere IP-basierte Netzwerk) angeschlossene Computer wird über eine einmalige IP-Adresse identifiziert. IP-Adressen haben eine Länge von vier Byte, die normalerweise als vier dezimale Zahlen, getrennt durch Punkte, dargestellt werden. Im Folgenden sehen Sie einige gültige Beispiele:

127.0.0.1

192.168.0.5

240.250.240.250

Anhand einer IP-Adresse lassen sich zwei Dinge feststellen: erstens das Netzwerk, in dem sich ein bestimmter Rechner befindet, und zweitens der Rechner selbst – dies entspricht dem so genannten Netzwerk- und dem Host-Teil der Adresse.

Für den Host-Teil der Adresse gibt es zwei besondere Werte: Wenn alle Bits auf 0 gesetzt sind, nennt man dies die Netzwerkadresse (network address, 0), wenn alle auf 1 gesetzt sind, beschreibt dies alle Rechner in diesem Netz – die so genannte Broadcast-Adresse (255).

### 7.3.2 IP-Interface-Konfiguration

Oder: »Wie kommt die Netzwerkkarte zu Ihrer Adresse?« In den meisten Fällen werden Sie die Konfiguration Ihrer Netzwerkkarte bereits bei der Installation von Debian GNU/Linux vorgenommen haben. Wenn Sie eine solche Karte später einbauen oder das System mit einer zweiten Karte erweitern, müssen Sie diese von Hand einbinden.

Das Kommando `ifconfig` wird benutzt, um einem Netzwerkinterface (dies kann eine Netzwerkkarte, ein Modem, eine ISDN-Verbindung oder Ähnliches sein ...) die benötigten Einstellungen zuzuweisen.

- Loopback – Ein »virtuelles« Netzwerkinterface. Dieses bezieht sich immer auf den Rechner selbst und hat die IP-Adresse: 127.0.0.1. Interface: lo.
- Ethernet – Um zwei oder mehrere Rechner miteinander zu vernetzen, verwendet man eine Netzwerkkarte. Standard ist dabei der Einsatz von Ethernet-Karten. Interface: eth0, eth1, eth2 usw.
- Tokenring – Ebenfalls zur Vernetzung über relativ kurze Strecken. Interface: tr0, tr1, tr2 usw.
- PPP – »Point-to-Point Protocol« (Punkt-zu-Punkt-Protokoll). Zur Verwendung über serielle Verbindungen, wie zum Beispiel Modemstrecken, ISDN usw. Interface: ppp0, ppp1, ppp2 usw.
- Dummy – Für Systeme, die ansonsten keine Netzwerkverbindungen haben. Interface: dummy, dummy0, dummy1 usw.

Es gibt noch einige weitere Beispiele. Wir werden uns hier mit dem am meisten verbreiteten, dem Ethernet, beschäftigen.

Wenn Ihr Kernel die installierte Netzwerkkarte nicht erkennt, müssen Sie das passende Modul laden. Sie können dies temporär mit dem Kommando `insmod` oder `modprobe` durchführen. Wenn Sie das richtige Modul gefunden haben, tragen Sie die nötigen Werte in die Datei `/etc/modutils` ein. Danach ist das Kommando `update-modules` aufzurufen. Für eine NE2000-kompatible Karte wäre dies beispielsweise die Zeile `alias eth0 ne` (in der Datei `/etc/conf.modules`).

Mit dem Kommando `ifconfig` weisen Sie nun der Karte eine IP-Adresse zu. Im einfachsten Fall, also ohne weitere Optionen, gibt `ifconfig` die bisher konfigurierten Interfaces aus:

```
/sbin/ifconfig
lo Link encap:Local Loopback
 inet addr:127.0.0.1 Bcast:127.255.255.255 Mask:255.0.0.0
 UP BROADCAST LOOPBACK RUNNING MTU:3584 Metric:1
 RX packets:18584 errors:0 dropped:0 overruns:0 frame:0
 TX packets:18584 errors:0 dropped:0 overruns:0 carrier:0
 Collisions:0
```

Bisher sollte auf Ihrem Rechner lediglich das Loopback-Interface (lo) konfiguriert sein.

Rufen Sie nun `ifconfig` mit dem Namen des Interfaces (eth0 für die erste, eth1 für die zweite Karte usw.) sowie der gewünschten IP-Nummer auf, und sehen Sie sich die Veränderung an:

```
/sbin/ifconfig eth0 192.109.42.23
/sbin/ifconfig
lo Link encap:Local Loopback
 inet addr:127.0.0.1 Bcast:127.255.255.255 Mask:255.0.0.0
 UP BROADCAST LOOPBACK RUNNING MTU:3584 Metric:1
 RX packets:18584 errors:0 dropped:0 overruns:0 frame:0
 TX packets:18584 errors:0 dropped:0 overruns:0 carrier:0
 Collisions:0
```



```
eth0 Link encap:Ethernet HWaddr 00:00:F7:C3:45:1B
 inet addr:192.109.42.23 Bcast:192.109.42.255 Mask:255.255.255.0
 UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
 RX packets:55 errors:0 dropped:0 overruns:0 frame:0
 TX packets:5 errors:0 dropped:0 overruns:0 carrier:0
 Collisions:7
 Interrupt:10 Base address:0x300
```

`ifconfig` gibt Ihnen nun auch die Informationen zur Ethernet-Karte aus. Die Werte für die Netzmaske, Broadcast-Adresse usw. wurden automatisch gesetzt. Wenn Sie bei diesen Werten andere Einstellungen benötigen, so müssen Sie diese ebenfalls mit `ifconfig` setzen:

```
/sbin/ifconfig eth0 192.109.42.23 netmask 255.255.255.240
```

Dies verändert die zuvor angegebenen Werte. Wenn Sie ein Interface komplett deaktivieren wollen, benutzen Sie die Option `down`:

```
/sbin/ifconfig eth0 down
```

Wenn Sie Ihre Netzwerkkarte bereits bei der Installation von Debian GNU/Linux eingerichtet haben, so werden Sie feststellen, dass einfach schon alles funktioniert ... Hierzu wurden die notwendigen Werte in die Datei `/etc/network/interfaces` geschrieben.

```
/etc/network/interfaces -- configuration file for ifup(8), ifdown(8)

The loopback interface
iface lo inet loopback

The first network card - this entry was created during the Debian
installation (network, broadcast and gateway are optional)
iface eth0 inet static
 address 192.109.42.18
 netmask 255.255.255.240
 network 192.109.42.16
 broadcast 192.109.42.16
 gateway 192.109.42.17
```

In dieser Datei sind einige Variablen definiert, die beim Aufruf des Programms `ifup` ausgelesen werden.

Wenn Sie Ihre Netzwerkkarte erst später eingerichtet haben, müssen Sie diese Datei erstellen und sie an Ihre Bedürfnisse anpassen.



### Netzwerkinformationen berechnen

Aus wenigen Angaben eines Netzwerks die fehlenden Informationen zu generieren ist oft auch für den erfahrenen Administrator nicht ganz einfach und zudem fehleranfällig, wenn diese Berechnungen von Hand gemacht werden. Abhilfe schafft hier das Programm `sipcalc` (<http://www.routemeister.net/projects/sipcalc/>).

```
wasabi:~# sipcalc 192.168.1.1/24
-[ipv4 : 192.168.1.1/24] - 0

[CIDR]
Host address - 192.168.1.1
Host address (decimal) - 3232235777
Host address (hex) - C0A80101
Network address - 192.168.1.0
Network mask - 255.255.255.0
Network mask (bits) - 24
Network mask (hex) - FFFFFFF0
Broadcast address - 192.168.1.255
Cisco wildcard - 0.0.0.255
Addresses in network - 256
Network range - 192.168.1.0 - 192.168.1.255
Usable range - 192.168.1.1 - 192.168.1.254
```

### dhclient

Mit dem Kommando `dhclient` lässt sich eine Anfrage an einen DHCP- (Dynamic Host Configuration Protocol) Server stellen. Von diesem können dann alle notwendigen Informationen für die Netzwerkkonfiguration bezogen werden.

Als wichtigste Option kann auf der Kommandozeile das gewünschte Netzwerkinterface angegeben werden, welches konfiguriert werden soll.



### Nameserver trotz DHCP vorgeben

Mit einer Anfrage an einen DHCP-Server wird auch meistens neben der IP-Nummer für den anfragenden Client, eine IP-Nummer für einen geeigneten Nameserver mitgeliefert. Dies ist nicht in jedem Fall gewünscht oder sinnvoll. Durch den Eintrag

```
prepend domain-name-servers 192.168.15.4;
```

in der Datei `/etc/dhclient.conf` wird der angegebene Nameserver als erster in die Datei `/etc/resolv.conf` eingetragen und verwendet.

### 7.3.3 IP-Routing

Wenn Sie der Netzwerkkarte die eigenen Einstellungen mitgeteilt haben, müssen Sie dem Linux-Kernel noch beibringen, die IP-Pakete auch an die richtige Stelle zu schicken. Dies kann bei mehreren Netzwerkkarten durchaus richtige Arbeit werden ...

Der Kernel hält die Informationen, welcher Rechner oder welches Netzwerk über welchen Weg zu erreichen ist, in der so genannten Routing-Tabelle (»routing table«). Das Kommando, um sich diese Tabelle anzusehen, lautet: `route`. Wenn Sie bisher nur das Loopback-Device konfiguriert haben, wird das Ergebnis Folgendes sein:

```
/sbin/route
Kernel IP routing table
Destination Gateway Genmask Flags Metric Ref Use Iface
127.0.0.0 * 255.0.0.0 U 0 0 2 lo
```

Dies bedeutet, dass alle IP-Pakete (auch »traffic« genannt) für das Netzwerk 127 über das Loopback-Device (lo) geschickt werden sollen.

Wenn Sie nun eine Netzwerkkarte konfiguriert haben, müssen Sie diese auch in die Routing-Tabelle des Kernels eintragen:

```
/sbin/route add -net 192.109.42.0
/sbin/route
Kernel IP routing table
Destination Gateway Genmask Flags Metric Ref Use Iface
127.0.0.0 * 255.0.0.0 U 0 0 2 lo
192.109.42.0 * 255.255.255.0 U 0 0 137 eth0
```

Dies führt dazu, dass alle IP-Pakete für das Class-C-Netz 192.109.42.xxx über das Interface eth0 zu erreichen sind. Wenn Sie nur einen kleinen Teil eines Netzes zur Verfügung haben, müssen Sie dies über die Netzmaske (netmask) steuern:

```
/sbin/route add -net 192.109.42.16 netmask 255.255.255.240
```

In diesem Beispiel stehen 16 IP-Nummern zur Verfügung (.16 bis .31).

Dies reicht schon aus, um ein kleines internes Netzwerk zu betreiben. Wenn Sie jedoch zwei Netzwerke miteinander verbinden wollen, müssen Sie dies über ein so genanntes »Gateway« (oder auch »Router« genannt) tun.

Wenn über das neue Interface nur ein bestimmtes Netzwerk zu erreichen ist und Sie die Adresse des Gateways (Routers) für dieses Netz kennen, müssen Sie die Routing-Tabelle um genau diesen Eintrag ergänzen. Sie erreichen dies mit der Option `-net`:

```
route add -net 193.174.1.0 gw 192.109.42.17
```

Dies bedeutet, dass das Netz 193.174.1.0 (Class C) über das Gateway 192.109.42.17 zu erreichen ist. Es setzt aber auch voraus, dass schon bekannt ist, wie das Netz 192.109.42.x zu erreichen ist.

Eine andere häufig anzutreffende Möglichkeit ist es, alle IP-Pakete, für die es keinen passenden Eintrag in der Routing-Tabelle gibt, an ein Gateway zu schicken, die so genannte »Default-Route«.

```
route add default gw 192.109.42.17
```

Dies würde zu folgender Routing-Tabelle führen:

```
Kernel IP routing table
Destination Gateway Genmask Flags Metric Ref Use Iface
127.0.0.0 * 255.0.0.0 U 0 0 2 lo
192.109.42.16 0.0.0.0 255.255.255.240 U 0 0 0 eth0
0.0.0.0 192.109.42.17 0.0.0.0 UG 1 0 0 eth0
```

Sehen wir uns die Einträge im Einzelnen an: Die erste Zeile bezieht sich auf das Loopback-Device (127.0.0.0, lo). Wenn Sie einen neueren Kernel verwenden, wird dies unter Umständen nicht angezeigt.

Werfen wir jedoch zunächst noch einmal einen genaueren Blick auf die Datei `/etc/network/interfaces`:

```
/etc/network/interfaces -- configuration file for ifup(8), ifdown(8)

The loopback interface
iface lo inet loopback

The first network card - this entry was created during the Debian
installation (network, broadcast and gateway are optional)
iface eth0 inet static
 address 192.109.42.18
 netmask 255.255.255.240
 network 192.109.42.16
 broadcast 192.109.42.16
 gateway 192.109.42.17
```

Diese Datei definiert einige Variablen; wenn Sie eine Änderung an dieser Datei vornehmen, können Sie diese einfach mit dem Kommando `ifup eth0` aktivieren. In den ersten Zeilen wird das so genannte Loopback-Device konfiguriert.

Die folgenden Zeilen beinhalten die benötigten Variablen für Ihr Netz.

In dieser Datei werden IP-Nummern verwendet. Wenn Sie das Kommando `route` aufrufen, um sich die Routing-Tabelle anzeigen zu lassen, werden Sie bemerken, dass dort die Namen der Rechner und Netze verwendet werden. Sie können die Ausgabe der IP-Nummern mit der Option `-n` hinter dem Kommando `route` erzwingen.

In der Datei `/etc/hosts` finden Sie den Namen des lokalen Rechners und in der Datei `/etc/networks` den Namen des Netzes. Systeme, deren Name nicht in der Datei `/etc/hosts` beschrieben sind, werden im Allgemeinen über den Nameserver aufgelöst.

### 7.3.4 Domain Name Server – DNS

Jeder mit dem Internet verbundene Rechner oder jedes Netzwerk, das IP als Netzwerk-Protokoll benutzt, hat eine (oder mehrere) IP-Adressen, mit der das Routing von und zu diesem System sichergestellt wird. Da die Benutzer sich diese Zahlenkolonnen nur schwer merken können, kann jedem Rechner (eigentlich jeder IP-Nummer) ein Name zugewiesen werden. Sicher haben Sie dies schon bei der Installation von Debian GNU/Linux bemerkt: Sie werden nach einem Rechnernamen (Hostnamen) gefragt.

Dieser Name wird an verschiedenen Stellen gespeichert. Sie können mit dem Kommando `hostname` den Namen des Rechners herausfinden; es wird der Wert aus der Datei `/etc/hostname` ausgegeben. Wenn Sie dies an einem (dauerhaft) ans Internet angeschlossenen Rechner probieren, besteht dieser Name aus verschiedenen Teilen, die durch Punkte getrennt sind, zum Beispiel `linux.openoffice.de`.

Nur der erste Teil ist der Name des Rechners (Hostname), der Rest wird als »Domain Name« bezeichnet (hier: `openoffice.de`). Da es natürlich viele Menschen gibt, die möchten, dass ihr Rechner den Namen `linux` bekommt, wurde jeder dieser Namen dadurch einmalig gemacht, dass man ihn einer bestimmten Domäne zuordnet.

Innerhalb einer Domäne (also einer Gruppe von Computern, einem Netzwerk) ist meist eine Person für die Vergabe von Computernamen zuständig und pflegt die Daten(-bank) mit den notwendigen Informationen. Dieses System ist als DNS (Domain Name Service) bekannt. Es ist mit einem Telefonverzeichnis vergleichbar. Sie können nach einem Computernamen suchen und erhalten dessen IP-Adresse.

Vor der Einführung von DNS musste man sich die IP-Adressen der Rechner merken, um sich im Internet bewegen zu können. Namen wurden nur innerhalb kleinerer Netzwerke benutzt und konnten in verschiedenen Netzen doppelt vergeben werden. Diese (lokalen) Namen wurden in der Datei `/etc/hosts` gespeichert. Natürlich können Sie auch heute noch diese Datei benutzen, sie wird auch bei der Installation von Debian GNU/Linux angelegt. Die Datei enthält den Namen des eigenen Rechners und die dazugehörige IP-Adresse.

DNS ist ein sehr umfangreiches Thema; wir werden uns an dieser Stelle nur aus der Sicht des Nutzers (Clients) damit befassen.

Wenn ein Debian GNU/Linux-System die IP-Adresse eines anderen Rechners benötigt (zum Verschicken von Mail oder um eine Webseite aufzurufen), wird ein Teil der C-Library benutzt, um die Informationen zu bekommen – der »Resolver«. Zuerst wird ein Blick in die Datei `/etc/nsswitch.conf` geworfen (können Sie an dieser Stelle auch tun ...). In dieser ist aufgeführt, an welchen Stellen versucht werden soll, die IP-Nummer zu finden. Hier sind drei verschiedene Einträge möglich. Wenn mehrere vorhanden sind, werden sie der Reihenfolge nach durchsucht.

- `files` – Versucht, den Hostnamen in der Datei `/etc/hosts` zu finden.
- `dns` – Probiert es über eine Nameserver-Abfrage.
- `nis` – Fragt eine NIS-Datenbank.

Ein üblicher Eintrag in der Datei `/etc/nsswitch.conf` wäre:

```
hosts: files dns
```

Dieser Eintrag bringt den Resolver dazu, zuerst die Datei `/etc/hosts` und dann den (in `/etc/resolv.conf` definierten) Nameserver (DNS) nach dem Rechnernamen zu durchsuchen.

Die Datei `/etc/resolv.conf` enthält, neben dem Eintrag für einen oder mehrere Nameserver, zunächst den Namen der lokalen Domain.

```
domain openoffice.de
nameserver 194.25.2.129
nameserver 192.168.22.33
```

Die erste Zeile bewirkt, dass die lokale Domain an den Rechnernamen (Hostname) angehängt wird, wenn nur ein Rechnername ohne Angabe der Domain angegeben wird. Die folgenden Zeilen beschreiben die (IP-technisch gesehen möglichst gut erreichbaren) Nameserver. Sie sollten hier in jedem Fall IP-Nummern und nicht die Namen der Rechner verwenden.

Alle Programme, die Sie benutzen (zum Beispiel Webbrowser), erfragen automatisch die IP-Nummer, wenn Sie einen Rechnernamen eingeben. Natürlich können Sie aber auch selbst solche Abfragen starten. Das zu Debian GNU/Linux gehörende Paket `dnsutils` enthält das Programm `nslookup`. Sie können `nslookup` interaktiv benutzen, um mehrere Abfragen nacheinander zu starten. Die häufigste Anwendung ist aber, `nslookup` für eine einzelne Abfrage zu benutzen. Hierzu geben Sie nach dem Kommando einfach den Namen des gesuchten Rechners an:

```
nslookup www.debian.org

Server: 194.25.2.129
Address: 194.25.2.129#53

Non-authoritative answer:
Name: www.debian.org
Address: 198.186.203.20
```

Nach dem befragten Server wird weiter unten die IP-Adresse für den gewünschten Server angegeben.

### 7.3.5 resolv.conf

Auf die Datei `/etc/resolv.conf` wurde ja schon im vorherigen Abschnitt eingegangen. Hier noch ein etwas erweitertes Beispiel:

```
domain openoffice.de
search openoffice.de debiananwenderhandbuch.de debian.org
nameserver 194.25.2.129
nameserver 192.168.22.33
```

Hinzugekommen ist die zweite Zeile. Diese ist nicht zwingend notwendig, erleichtert aber die Arbeit. Mit dieser Zeile werden einfach (wenn Sie nur den Hostnamen angeben) die angegebenen Domains nach dem gewünschten Namen durchsucht, und der Hostname wird automatisch um den Domainnamen ergänzt.

### 7.3.6 resolvconf

Das Paket `resolvconf` verwaltet Informationen über die aktuell verfügbaren Name-server.

Die Datei `/etc/resolv.conf` war ursprünglich eine statische Konfigurationsdatei, in der nur sehr selten Anpassungen vom Administrator vorgenommen wurden. Mit der Verbreitung von mobilen Computern wurde es jedoch üblich, dass ein Rechner sehr häufig in verschiedenen Netzwerken eingesetzt wurde. Um am jeweils aktuellen Standort eine Auflösung von Rechnernamen in IP-Adressen zu erlauben, muss die Datei `/etc/resolv.conf` an den aktuellen Standort angepasst werden. Eine ganze Reihe von Paketen erledigen dies durch direktes Ändern der Datei `/etc/resolv.conf`. Dies sind beispielsweise `pcmcia-cs`, `ppp`, `pppconfig`, `pppoeconf`, `webmin` und `dhcp-client` und noch viele andere. Alle diese Pakete kooperieren aber nicht bei Änderungen an der Datei, jedes Paket überschreibt die Anpassung eines anderen Pakets. `resolvconf` soll zu einer Vereinheitlichung bei der Verwaltung der Datei `/etc/resolv.conf` beitragen.

Im Zusammenhang mit der Installation von `resolvconf` ist darauf zu achten, auch alle anderen Pakete zu aktualisieren, die Änderungen an der Datei `/etc/resolv.conf` vornehmen. Diese Pakete werden ständig aktualisiert und auf die Zusammenarbeit mit `resolvconf` hin optimiert. Mit der Version *Sarge* von Debian ist ein Großteil der Pakete bereits angepasst.

Nach der Installation von `resolvconf` sind zunächst alle Programme, die bisher in die Datei `/etc/resolv.conf` geschrieben haben, zu stoppen und hinsichtlich der Konfiguration anzupassen. Einige wenige müssen komplett entfernt werden. Eine detaillierte Auflistung findet sich in der Datei `/usr/share/doc/resolvconf/README.gz`.

Für jedes logische Interface, welches in der Datei `/etc/network/interfaces` aufgeführt ist, muss nun ein Eintrag in der Form

```
dns-nameservers 11.22.33.44 55.66.77.88
dns-search bla.org blub.org
```

hinzugefügt werden. Weiterhin sind hier Einträge mit den Schlüsselwörtern `dns-domain` und `dns-sortlist` erlaubt. Die zu jedem Interface aufgeführten Optionen werden beim Aktivieren eines Interfaces in die Datei `/etc/resolv.conf` geschrieben, jeweils ohne die vorangestellte Zeichenkette `dns-`.