

Mehrere Ethernet-Karten in einem Rechner HOWTO

Philipp JW Grau (phgrau@wi-bw.tfh-wildau.de)

v0.2, 5. Februar 1998

Dieses Dokument beschreibt die Nutzung mehrere Ethernet-Karten in einem Linux-Computer.

Inhaltsverzeichnis

1	Copyright	1
2	Schnellstart: Ladbare Kernel-Module für Netzwerkkarten	2
2.1	PCI-Karten	2
2.2	ISA-Karten	2
3	Schnellstart: Eingebaute Netzwerkkarten-Treibern	2
4	Ladbare Kernel-Module für Netzwerkkarten	2
5	Eincompilierte Treiber: Was gemacht wurde und wie...	3
6	Parameterübergabe mit dem Boot-Loader	3
7	Modifizierung des Kernels	4
8	Hinweise auf spezielle Karten	5
8.1	PCI-Karten	5
8.2	LANCE/PCnet Karten	5
8.3	3c509 im ISA-Modus	5
8.4	EISA 3c579 und 3c509 im EISA-Modus (alt)	5
9	Ergänzungen (Neu in der deutschen Version)	5
9.1	Auswahl der Karten bei Kernel-Konfiguration	5
9.2	Andere deutschsprachige HOWTOs	5
9.3	Andere Verweise	6

1 Copyright

Dieses Dokument ist urheberrechtlich geschützt. Das Copyright für die englische *Multiple-Ethernet mini-HOWTO*, auf der dieses Dokument basiert, liegt bei Don Becker. Das Copyright für die deutsche Version liegt bei Philipp JW Grau.

Das Dokument darf gemäß der GNU *General Public License* verbreitet werden. Insbesondere bedeutet dieses, daß der Text sowohl über elektronische wie auch physikalische Medien ohne die Zahlung von Lizenzgebühren verbreitet werden darf, solange dieser Copyright Hinweis nicht entfernt wird. Eine kommerzielle Verbreitung ist erlaubt und

ausdrücklich erwünscht. Bei einer Publikation in Papierform ist das Deutsche Linux HOWTO Projekt hierüber zu informieren.

2 Schnellstart: Ladbare Kernel-Module für Netzwerkkarten

2.1 PCI-Karten

PCI-Karten können vom Kernel sicher erkannt werden, d.h. man muß nur einen Eintrag in die Modul Konfigurationsdatei `/etc/conf.modules` machen.

```
alias eth1 tulip
```

Hierbei steht `tulip` für das gewünschte Karten-Modul

2.2 ISA-Karten

Die Erkennung von ISA-Karten ist meist etwas gefährlich, deshalb muß man die I/O-Adresse der Ethernetkarten angeben. Wenn mehrere Karten von einem Modul unterstützt werden, *muß* für jede Karte die I/O-Adresse angegeben werden. Das sieht in `/etc/conf.modules` z.B. so aus:

```
alias eth0 ne
alias eth1 ne
alias eth2 ne
options ne io=0x280,0x300,0x320
```

Oder für eine einzelne Karte:

```
options ne io=0x280 irq=5
```

Hierbei steht `ne` für das gewünschte Karten-Modul

3 Schnellstart: Eingebaute Netzwerkkarten-Treibern

Für die meisten Anwender, die einen Kernel mit eincompilierten Netzwerkkarten-Treibern benutzen, sollte es reichen, folgende Zeilen an den Anfang ihrer `/etc/lilo.conf` zu schreiben und 'lilo' aufzurufen:

```
append = "ether=0,0,eth1 ether=0,0,eth2"
```

Das ist alles, was zu tun ist. Wenn Linux nun neu gebootet wird, sollte die zweite und dritte Ethernet-Karte erkannt werden. Voraussetzung ist, daß für alle Netzwerkkarten die Unterstützung im Kernel enthalten ist.

4 Ladbare Kernel-Module für Netzwerkkarten

Für Distributionen wie Debian und RedHat, die fast nur ladbare Device-Treiber nutzen, ist die Situation etwas komplizierter und Distributions-abhängig. Wenn man wissen möchte, ob schon Module geladen sind, kann man mit dem Befehl `lsmod` nachsehen. Man kann auch `cat /proc/modules` benutzen, das alle geladenen Module anzeigt.

Ladbare Module werden entweder von einem Prozeß *kerneld*, der die Kernelanfragen für Netzwerkschnittstellen verwaltet, oder von `modprobe`, das jedes Treiber-Modul lädt, um ein passendes zu finden, geladen.

Der `kerneld`-Prozeß liest die Konfigurationsdatei `/etc/conf.modules`, die jeden Treiber beschreibt, der geladen werden soll.

Außerdem kann man noch dem Kernel sagen, welche Module schon beim Booten zu laden sind. Hierzu kann man die zu ladenden Module in die Datei `/etc/modules` eintragen.

```
# /etc/modules: kernel modules to load at boot time.
#
# This file should contain the names of kernel modules that are
# to be loaded at boot time, one per line.  Comments begin with
# a '#', and everything on the line after them are ignored.
# An entry named 'auto' will cause the system to start kerneld immediately.
# Kerneld then loads modules on demand. 'noauto' disables kerneld completely.

#auto
ne
lp
sg
```

5 Eincompilierte Treiber: Was gemacht wurde und wie...

Normalerweise sucht ein gewöhnlicher Linux-Kernel nach einer einzelnen Ethernet-Karte. Wenn diese gefunden wird, ist der Test auf Netzwerkkarten abgeschlossen.

Es gibt drei Wege, um den Kernel nach weiteren Karten suchen zu lassen.

1. Parameterübergabe zur Bootzeit, am LILO-Prompt;
2. den Bootlader so zu konfigurieren, daß die Parameter immer mit übergeben werden;
3. die Netzwerkkarten-Tabelle des Kernels zu modifizieren.

Für die meisten Anwender ist die zweite Methode das richtige Verfahren, und oben wurde eben dieses beschrieben.

6 Parameterübergabe mit dem Boot-Loader

Im folgenden wird angenommen, daß der Standard Linux-Boot-Lader "LILO" benutzt wird.

Der Linux-Kernel kennt viele Parameter, die zur Bootzeit übergeben werden können. Die meisten Parameter spezifizieren Aspekte, die nicht von Kernel selbstbestimmt werden können. Für Netzwerk-Adapter werden die folgenden Parameter ausgewertet:

```
ether=<IRQ>,<IO-ADDR>,<PARAM1>,<PARAM2>,<NAME>
```

Gültige numerische Argumente können als dezimale, oktale (mit führender "0") oder hexadezimale (mit dem Prefix "0x") Zahl angegeben werden. Das erste nicht-numerische Argument wird als NAME der Device genommen. Leere Argumente werden als Null angenommen, und leere Argumente vor dem Namen werden nicht verändert.

IRQ:

Dieser Parameter spezifiziert den IRQ, der gesetzt werden soll, auf Karten, die den IRQ per Software setzen, oder der benutzte IRQ, auf Karten mit gejumperten IRQ. Der Wert "0" bedeutet, daß der IRQ wenn möglich von der Ethernetkarte gelesen werden soll, oder das "autoIRQ" genutzt werden soll, wenn die Netzwerkkarte keine Weg bereitstellt den IRQ zu lesen.

IO-ADDR:

Dieser Parameter legt die Basis-I/O-Adresse fest, auf der nach der Karte gesucht werden soll. Ein Wert von Null sagt, daß alle sinnvollen Adresse getestet werden sollen.

Normalerweise wird ein reservierter Adressbereich benutzt, um zu entscheiden, ob eine Stelle getestet werden soll. Dieser Adressbereich wird ignoriert, wenn eine I/O-Adresse angegeben wird. Dies erlaubt dem "reserve"-Parameter Device-Proben in einer I/O-Region zu unterbinden.

PARAM1, PARAM2:

Ursprünglich wurden diese Parameter dazu benutzt, um die Speicher-Adressen für Adapter, die Shared-Memory benutzen, wie der WD8013, festzulegen. Im Laufe der Zeit wurden sie erweitert, um anderen Treibern spezifische Optionen zu übergeben.

NAME:

Der Name eines definierten Device. Der Standard-Kernel definiert im allgemeinen "eth0", "eth1", "eth2" und "eth3". Andere Device-Namen (z.B. für PPP, SLIP oder ein Pocket-Ethernet-Adapter) sind vielleicht vorhanden, gehorchen aber einer anderen Semantik.

LILO stellt zwei Wege bereit, Parameter zur Bootzeit dem Kernel zu übergeben. Der am häufigsten genutzte Weg, ist die direkte Eingabe am Boot-Prompt, hinter der Angabe des Kernelimages. Das folgende Beispiel aktiviert alle vier möglichen Devices.

```
boot: linux ether=0,0,eth1 ether=0,0,eth2 ether=0,0,eth3
```

Natürlich ist es ziemlich kompliziert, bei jedem Booten diese Parameter einzugeben, und erfordert unbeaufsichtigte Roboter. Man kann Kernel-Parameter permanent festlegen, durch die Einfügung einer "append" Zeile in die LILO-Konfigurations-Datei, `/etc/lilo.conf`, und einen anschließenden Aufruf von `lilo` um die geänderte Konfiguration zu aktivieren.

```
append = "ether=0,0,eth1 ether=0,0,eth2 ether=0,0,eth3"
```

7 Modifizierung des Kernels

Wenn möglich sollte das System ohne Änderung der Kernelquellen konfiguriert werden. Eine Änderung der Kernelquellen ist nicht selbstdokumentierend und führt zu Problemen, wenn ein Upgrade durchgeführt wird. Es gibt aber ein paar Fälle, bei denen die Modifizierung notwendig ist.

- Wenn man mehr als vier Devices benötigt. Die Datei `drivers/net/Space.c` hat nur Einträge für eth0 bis eth3.
- Wenn das Testen nach Karten limitiert werden muß, um Konflikten mit anderer Hardware vorzubeugen.
- Wenn andere Device-Namen als ethN verwendet werden sollen.

Wenn man diesen Weg gehen will, muß man die Device-Liste in `drivers/net/Space.c` editieren. Man trage die gewünschten Werte ein. Wird ein neues Device eingefügt, sollte man darauf achten, daß die Verkettung erhalten bleibt. Man sollte die vorhandenen Einträge als Hilfe nutzen.

8 Hinweise auf spezielle Karten

8.1 PCI-Karten

PCI- und EISA-Karten können sicher und zuverlässig vom Kernel erkannt werden. Deshalb finden die meisten PCI-Treiber alle unterstützten Karten ohne zusätzliche Parameter. Die Ausnahme bilden Treiber, die sowohl die ISA- als auch die PCI-Versionen von Netzwerkkarten unterstützen: NE2000- und der ältere LANCE/PCnet-Netzwerkkarten-Treiber.

8.2 LANCE/PCnet Karten

Der Kernel 1.2.13 und frühere Versionen des LANCE-Treiber erfordern spezielle DMA-Buffer im "Low-Memory". Aus diesem Grunde ist die LANCE-Erkennung etwas anders und früher als andere Tests nach Netzwerkkarten. Erfreulicherweise kann man diesen ganzen Abschnitt ignorieren: mehrere LANCE-Karten werden automatisch gefunden. Andererseits kann der LANCE-Treiber z.Z. (noch) keine LILO-Parameter nutzen, z.B. die IRQ-Einstellung.

8.3 3c509 im ISA-Modus

Die 3c509 hat ein einzigartiges Feature, das wirklich sicheres Testen auf dem ISA-Bus erlaubt. Das ist prima, aber leider paßt diese Methode nicht mit den anderen Proben zusammen.

Der wichtigste Aspekt ist, daß es schwierig ist, *a priori* festzulegen, welche Karte als *erste* erkannt wird. Die Reihenfolge hängt von der Ethernet-Adresse der Karte ab. Das heißt, daß die Ethernetkarte mit der niedrigsten Nummer `eth0` zugewiesen wird, und der nächsten `eth1`, etc. Wenn die Karte `eth0` entfernt wird, rutschen alle eine Nummer niedriger.

Daraus folgt, daß es nicht möglich ist, eine *frühe* Karte unbenutzt zu lassen, eine Karte auf einer Adresse oder IRQ verschieden von den EEPROM-Einstellungen zu verwenden oder eine Karte auf einer bestimmten Adresse zu enablen.

8.4 EISA 3c579 und 3c509 im EISA-Modus (alt)

Kernel vor 1.1.25 testen nicht sauber auf mehrere EISA-Modus Karten. Wenn mehrere `ethN` angegeben werden, wird die *gleiche* 3c5x9 mehrmals gefunden. Um dies zu umgehen, kann man die I/O-Adresse für jede Karte einzeln angeben. Kernel nach 1.1.25 finden mehrere EISA-Modus Karten, und suchen danach zusätzliche ISA-Modus Karten, nach dem alle potentiellen EISA-Modus-Adressen getestet wurden.

9 Ergänzungen (Neu in der deutschen Version)

9.1 Auswahl der Karten bei Kernel-Konfiguration

Wenn ein neuer Kernel konfiguriert wird, sollten natürlich nur die Karten ausgewählt werden, die tatsächlich gebraucht werden. Vorteile: kleinerer Kernel.

9.2 Andere deutschsprachige HOWTOs

Andere deutsche Linux-Dokumentation, insbesondere die deutschen HOWTOs gibts an folgenden Stellen:

- <http://www.tu-harburg.de/~semb2204/dlhp/>
- <ftp://ftp.uni-stuttgart.de/pub/systems/linux/local/doc/>

Verweisen möchte ich außerdem noch auf das deutsche *Bridging mini-HOWTO* von Ralph Wetzel, das beim Bau einer Brücke mit Linux sehr nützlich ist.

9.3 Andere Verweise

- Die Seiten von Don Becker, dem Autor dieses HOWTOs und Maintainers der 3Com Netzwerkkarten Kernel-Module: <http://cesdis.gsfc.nasa.gov/pub/linux/drivers/index.html>
- Don Beckers Linux-Seiten: <http://cesdis.gsfc.nasa.gov/pub/linux/linux.html>
- Diagnose-Programme für Netzwerkkarten: <http://cesdis.gsfc.nasa.gov/pub/linux/diag/diagnostic>