

SelfLinux-0.10.0



## Dynamic Host Configuration Protocol

Autor: Andreas Schockenhoff (*asc@gmx.li*)  
Formatierung: Florian Frank (*florian.frank@pingos.org*)  
Lizenz: GFDL

Die Automatisierung der TCP/IP-Netzwerkadministration lässt sich mit dem DHCP-Protokoll recht erfolgreich durchführen. In diesem Text werden die Grundzüge des Protokolls erklärt. Außerdem werden einige einfache Beispielkonfigurationen besprochen.

## Inhaltsverzeichnis

### 1 DHCP

### 2 Leistungsmerkmale

### 3 Kurzbeschreibung der Arbeitsweise

### 4 Arten der Zuteilung von IP-Adressen

4.1 Feste IP-Adresse aufgrund der MAC-Adresse

4.2 1.1 Feste IP-Adresse, automatisch zugeteilt, mit Zeitbegrenzung

4.3 Dynamische Zuteilung

### 5 Konfiguration der Clients

5.1 DHCP-Client

5.1.1 Linux

5.1.2 BSD

5.1.3 Windows

5.1.4 Mac

### 6 DHCP-Server ([www.isc.org](http://www.isc.org))

6.1 Eine erste Konfigurationsdatei

6.2 Testen des Servers

6.3 Der Server bekommt weitere Fähigkeiten

6.4 Weitere Möglichkeiten mit DHCP

### 7 Links

## 1 DHCP

Über das Dynamic Host Configuration Protocol (DHCP) werden einem Client-Rechner alle nötigen Netzparameter übermittelt.

Siehe DHCP RFC-2131 und andere.

## 2 Leistungsmerkmale

DHCP dient der einfachen Zuteilung von Netzwerkinformationen in lokalen TCP/IP-Netzen. Zu diesen Informationen gehören z.B. **IP-Nummer**, **Domainname**, **Routing** und **DNS-Server**.

Bei Diskless-Workstations bilden **DHCP** oder **BOOTP** die Grundlage.

## 3 Kurzbeschreibung der Arbeitsweise

Beim Booten kennt der Rechner nur die **MAC-Adresse** seiner Netzwerkkarte.

Diese ist meistens in der Netzwerkhardware (Ethernet, FDDI, Firewire, Wireless LAN) programmiert. Bereiche für MAC-Adressen werden nach einem internationalen Standard von einem Gremium (siehe <http://standards.ieee.org/regauth/oui/index.shtml>) an die Hersteller der Hardware vergeben. Diese wiederum teilen jedem Netzwerkport eine eindeutige **MAC-Adresse** zu.

Mit der **MAC-Adresse** sendet der Rechner eine Rundfrage (Broadcast) ins Netz mit der Bitte, ihm eine Netzwerkkonfiguration mitzuteilen.

Der **DHCP-Server** wartet auf solche Rundfragen und teilt dem anfragenden Rechner die entsprechenden Daten mit.

Außerdem speichert der Server sowohl die **MAC-Adresse** des Clients als auch die ihm zugeteilte **IP-Adresse** in einer Datei. (`dhcp.leases`)

Diese **IP-Adresse** wird für eine bestimmte Laufzeit (**Lease Time**) für diesen Client reserviert. Nach einiger Zeit versucht der Client, seine Adresse beim Server zu erneuern (**renewing**). Gelingt ihm dies nicht, so fragt er etwas später alle Server nach seiner Adresse (**rebinding**).

Wird seine Adresse auch nach dieser Anfrage nicht erneuert, muss er alle Netzwerkaktivitäten beenden. Allerdings darf er das **DHCP-Protokoll** neu beginnen.

## 4 Arten der Zuteilung von IP-Adressen

### 4.1 Feste IP-Adresse aufgrund der MAC-Adresse

**MAC-Adressen** und **IP-Adresse** werden fest in die `dhcp.conf` eingetragen.

## Ausschnitt dhcp.conf

```
host test{
  hardware ethernet 00:00:c0:5d:bd:95;
  fixed-address 192.168.0.121;
  # Bei laufenden Nameservice ist auch
  # fixed-address test.local.invalid;
  # möglich
}
```

#### 4.2 1.1 Feste IP-Adresse, automatisch zugeteilt, mit Zeitbegrenzung

Hierbei wird die IP-Adresse für eine unbegrenzte (schlecht) oder eine lange Laufzeit zugeteilt.

Hier sollte man die Laufzeit (`lease-time`) auf einen höheren Wert setzen:

## Ausschnitt dhcp.conf file

```
max-lease-time 432000;
# 24*60*60*5 -> 5 Tage
subnet 192.168.0.0 netmask 255.255.255.0 {
  range dynamic-bootp 192.168.0.32 192.168.0.40;
}
```

#### 4.3 Dynamische Zuteilung

Die Vorgehensweise ist hier dieselbe wie oben, allerdings wird die Laufzeit auf einen niedrigeren Wert gesetzt.

## 5 Konfiguration der Clients

Heute sind die meisten Rechner mit Client-seitiger DHCP-Unterstützung vorkonfiguriert. Das **DHCP-Protokoll** vereinfacht die Netzwerkkonfiguration erheblich.

Beim Client müssen keinerlei rechner spezifische Netzwerkeinstellungen mehr vorgenommen werden. Im Prinzip muss man den Rechnern nur mitteilen, dass **DHCP** verwendet werden soll.

### 5.1 DHCP-Client

#### 5.1.1 Linux

`pump`, `dhcpcclient`, `dhcpcd`, `dhclient`. .....

Auch **DHCP-Clients** können umfangreich konfiguriert werden, siehe `/etc/dhpc/config` im Paket **dhcpcd**. So kann man zum Beispiel verhindern, dass vom **DHCP-Server** Daten übernommen werden, die eine spezielle Netzwerkkonfiguration überschreiben würden.

### 5.1.2 BSD

Ebenso wie Linux unterstützen auch BSD-Varianten **DHCP** als Client und Server.

### 5.1.3 Windows

Auch neuere Windows-Systeme bieten direkt nach der Installation **DHCP-Unterstützung**.

### 5.1.4 Mac

Sofort nach der Installation ist die Unterstützung vorhanden, ob man will oder nicht.

## 6 DHCP-Server (www.isc.org)

Die Installation des Servers sollte in den gängigen Distributionen kein Problem sein. Bei *Debian* z.B. sieht sie folgendermaßen aus:

```
root@linux / # apt-get install dhcp3-server
```

Bei Debian befindet sich nach der Installation die Konfigurationsdatei `/etc/dhcp3/dhcpd.conf` auf dem System.

Die Datei ist gut kommentiert, und man sollte sie den eigenen Bedürfnissen anpassen (**Domain-Name**, **IP-Adressen** usw.)

Falls man den Server in der Testphase nur von Hand starten möchte, sollte man die Links in den entsprechenden Runleveln löschen.

*Debian:*

```
root@linux / # update-rc.d -f dhcp3-server remove
```

Nach der Testphase kann man diese wieder aktivieren, bei *Debian* z.B. folgendermaßen:

```
root@linux / # update-rc.d dhcp3-server defaults
```

### 6.1 Eine erste Konfigurationsdatei

Alles, was man nicht versteht, kommentiert man zunächst einmal aus. Ich habe das meiste der Übersichtlichkeit halber gelöscht.

#### minimal dhcp.conf file

```
#
# Beispielkonfigurationsdatei für ISC dhcpd3
# Debian 3.0 Woody
#
# Optionsdefinitionen für alle unterstützten
# Netzwerke...
# option domain-name "local.invalid";
subnet 192.168.0.0 netmask 255.255.255.0 {
    range dynamic-bootp 192.168.0.31 192.168.0.40;
}
```

Nun kann der **DHCP-Server** den **DHCP-Clients** zehn Adressen dynamisch zuweisen.

### 6.2 Testen des Servers

Auf einem Debian-System sollte nochmals geprüft werden, ob in `/etc/default/dhcp3-server` das richtige Interface angegeben ist (z.B. `eth0`).

Danach kann man zum Testen (Debuggen) übergehen.

Starten des Servers:

```
root@linux / # /etc/init.d/dhcp start
```

bzw.

```
root@linux / # /etc/init.d/dhcp3-server start
```

Es empfiehlt sich, mit `tail -f /var/log/syslog` den Startvorgang zu verfolgen, um zu sehen, ob der Server korrekt läuft. "ps ax" bzw. "pidof dhcpd3"

Der folgende Aufruf ist zum Debuggen besser geeignet:

```
root@linux / # dhcpd3 -d -f
```

Bitte unbedingt die Fehlermeldungen beachten!

Auch ein Blick in die Dateien unter `/var/log` lohnt sich!

Wird auf einem zweitem Rechner, z.B.  [Knoppix](#) (Debian von CD), getestet, so verwendet man `pump`:

```
root@linux / # pump -s
```

Auf dem Server kann die Anfrage mit

```
root@linux / # tcpdump -e
```

verfolgt werden.

Dabei muss der Client eine Anfrage absetzen, die vom Server beantwortet wird.

#### tcpdump Ausgabe

```
(ist leider eine lange Zeile mit \ getrennt)
09:47:32.690249 0:48:54:12:4c:df Broadcast ip 342: \
0.0.0.0.bootpc > 255.255.255.255.bootps: \
xid:0x1c97abed [|bootp] [tos 0x10]

09:47:32.700021 8:0:46:49:60:8d 0:48:54:12:4c:df \
ip 62: lapwoody.local.invalid > 192.168.0.40: \
icmp: echo request (DF)
```

Der erste Abschnitt ist die **Broadcast-Anfrage** des Clients. Danach folgt die Antwort des Servers, in der dieser dem Client eine **IP-Adresse** mitteilt.

Anmerkung: Wer die Pakete mit ganzem Inhalt sehen möchte, kann für die Aufzeichnung auch `ngrep`

verwenden.

Nun sollte ein einfacher **DHCP-Server** laufen.

### 6.3 Der Server bekommt weitere Fähigkeiten

Als Nächstes sollten die sinnvollen Standard-Angaben wieder aktiviert werden. Die Optionen, die nicht "ausgeklammert" sind, gelten als Vorgaben für alle späteren Angaben.

config-datei
<pre>ddns-update-style none; # Wir haben unseren DHCP- nicht mit einem # DNS-Server kombiniert.  option domain-name "local.invalid"; option domain-name-servers 192.168.0.1; # Unsere Domain und die Nameserver  default-lease-time 600; max-lease-time 7200; # Wann müssen die Adressen, die wir verteilt haben, # erneuert werden? Die Laufzeit ist in Sekunden angegeben, # d.h. alle DHCP-Rechner sollten nach 10 min anfangen, # sich neue Adressen zu holen. Falls die Adresse innerhalb # von 2 Stunden nicht erneuert werden konnte, liegt # irgendein Fehler vor und der Client-Rechner sollte # seine IP-Adressen nicht mehr benutzen.  authoritative; # Regelt, welcher DHCP-Server Priorität hat. :-) # Aktivieren, sobald wir fertig sind.</pre>

### 6.4 Weitere Möglichkeiten mit DHCP

Es gibt globale Angaben, die von lokalen Angaben überschrieben werden können.

Einige Optionen aus: "`man dhcpd.conf`" und "`man dhcp-conf`" (offizielle Optionen sind in RFC 2132 beschrieben):

config-datei
<pre>option domain-name "local.invalid"; option domain-name-servers 192.168.0.1, 192.168.0.2;</pre>

Übertragung des **Domain-Namens** und der **DNS-Server**.

config-datei
<pre>option routers 192.168.0.1;</pre>

Angabe des **Default-Gateway**.

## 7 Links

 <http://www.isc.org/products/DHCP/>  
 <http://standards.ieee.org/regauth/oui/index.shtml>  
 <http://www.phystech.com/download/dhcpd.html>

<http://www.rfc-editor.org/>  
Bitte dort die Suchmaschine benutzen.

Andere Quellen:

 <http://www.linux-praxis.de/linux3/dhcp.html>  
 [http://www.linux-fuer-alle.de/doc\\_show.php?docid=8](http://www.linux-fuer-alle.de/doc_show.php?docid=8)  
 <http://www.linuxwiki.de/DHCP>