



Dynamic Host Configuration Protocol

Autor: Andreas Schockenhoff (*asc@gmx.li*)

Layout: Matthias Hagedorn (*matthias.hagedorn@selflinux.org*)

Lizenz: GFDL

Die Automatisierung der TCP/IP-Netzwerkadministration ist mit dem DHCP-Protokoll recht erfolgreich durchzuführen. In diesem Text werden die Grundzüge des Protokolls erklärt. Außerdem werden einige einfache Beispielfiguren durchgesprochen.

Inhaltsverzeichnis

1 DHCP

2 Leistungsmerkmale

3 Kurzbeschreibung der Arbeitsweise

4 Arten der Zuteilung von IP-Adressen

- 4.1 Feste IP's aufgrund der MAC-Adresse
- 4.2 Feste IP's automatisch zugeteilt - mit Zeitbegrenzung
- 4.3 Dynamische Zuteilung

5 Konfiguration der Klienten

- 5.1 DHCP-Klienten
 - 5.1.1 Linux
 - 5.1.2 BSD
 - 5.1.3 Windows
 - 5.1.4 Mac

6 DHCP Server (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 Ausblick

1 DHCP

Das Dynamic Host Configuration Protocol dient dazu, einem Client-Rechner seine Netzparameter zu übermitteln.

Siehe DHCP RFC-2131 und andere

2 Leistungsmerkmale

Einfache Zuteilung von Netzwerkinformationen in lokalen TCP/IP-Netzen. z.B. **IP-Nummer**, **Domainname**, **Routing**, **DNS-Server** u.s.w.

Für 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 die Netzwerkhardware programmiert (Ethernet, FDDI, Firewire, Wireless LAN). 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 aufgeteilt. Diese vergeben dann an jeden Netzwerkport eine eindeutige **MAC-Adresse**.

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

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

Außerdem merkt sich der Server die **MAC-Adresse** des Klienten und die zugeteilte **IP-Adresse** in einer Datei. (`dhcp.leases`)

Diese **IP-Adresse** wird für eine bestimmte Zeit für diesen Klienten reserviert (**Lease Time**). Nach einiger Zeit versucht der Klient, seine Adresse bei dem Server zu erneuern (**renewing**). Wenn er keinen Erfolg hat, fragt der Klient etwas später alle Server nach seiner Adresse (**rebinding**).

Bekommt er seine Adresse nicht erneuert, muß er alle Netzwerkaktivitäten stoppen. Er darf dann aber von vorn mit dem **DHCP-Protokoll** beginnen.

4 Arten der Zuteilung von IP-Adressen

4.1 Feste IP's aufgrund der MAC-Adresse

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

Ausschnitt dhcp.conf file

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

4.2 Feste IP's automatisch zugeteilt - mit Zeitbegrenzung

Hier sind unbegrenzte (schlecht) oder lange Laufzeit möglich.

Hier sollte man die `lease Time` hochsetzen:

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

Wie oben, nur `lease Time` nicht so hoch setzen.

5 Konfiguration der Klienten

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

Es sind auf den Klienten keine rechnerspezifischen Netzwerkeinstellungen mehr vorzunehmen. Im Prinzip muß man den Rechnern nur sagen, daß **DHCP** genutzt werden soll.

5.1 DHCP-Klienten

5.1.1 Linux

`pump, dhcpclient, dhcpcd, dhclient.`

Auch **DHCP-Klienten** können umfangreich konfiguriert werden, siehe: `/etc/dhpcp/config` aus dem `dhcpcd`-Paket. So kann man auch verhindern, dass Daten vom **DHCP-Server** übernommen werden, die einem eine spezielle Netzwerkkonfiguration überschreiben.

5.1.2 BSD

BSD-Varianten unterstützen, genau wie Linux, **DHCP** als Client und Server.

5.1.3 Windows

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

5.1.4 Mac

Man hat die Unterstützung direkt nach der Installation, ob man will oder nicht. :-)

6 DHCP Server (www.isc.org)

Die Installation des Servers sollte in den gängigen Distributionen kein Problem sein, z.B. bei *Debian*:

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

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

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

Falls man den Server in der Testphase nur von Hand starten will, 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, z.B. *Debian*:

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

6.1 Eine erste Konfigurationsdatei

Alles, was man nicht versteht, kommentiert man erstmal aus. ;-) Ich habe das meiste der Übersicht wegen gelöscht.

minimal dhcp.conf file

```
#
# Sample configuration file for ISC dhcpd3 for
# Debian 3.0 Woody
#
# option definitions common to all supported
# networks...
# 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** zehn Adressen dynamisch an die **DHCP-Klienten** zuweisen.

6.2 Testen des Servers

Auf einem Debian System sollte nochmal in `/etc/default/dhcp3-server` geschaut werden, ob dort 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
```

Mal mit `tail -f /var/log/syslog` den Start verfolgen. Läuft der Server? "`ps ax`" bzw. "`pidof dhcpd3`"

Besser zum Debuggen ist wohl der Aufruf

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

Fehlermeldungen beachten!

Auch mal in Files unter `/var/log/*` schauen!

Testen mit einem zweitem Rechner, z.B. *Knoppix* (*Debian* von CD) mit `pump`:

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

Auf dem Server kann die Anfrage mit

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

verfolgt werden.

Die Anfrage vom Klient muß kommen und wird vom Server beantwortet.

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)
```

Das erste ist die Anfrage per **Broadcast** vom Klienten. Dann kommt die Antwort mit der **IP-Adresse** für den Klienten.

Anmerkung: Wer die Pakete mit ganzem Inhalt sehen will, kann es auch mit `ngrep` aufzeichnen.

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

6.3 Der Server bekommt weitere Fähigkeiten

Als nächstes sollte man wieder die sinnvollen Standard-Angaben anschalten. Die Optionen, die nicht "geklammert" sind, gelten als Vorgaben für alle späteren Angaben.

config-datei

```
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 name Server.

default-lease-time 600;
max-lease-time 7200;
# Wann müssen die Adressen die wir verteilt haben
# erneuert werden. In Sekunden, also alle DHCP
# Rechner sollten nach 10 min anfangen euch neue
# Adressen zu holen Wenn 2 Stunden die Adresse
# nicht erneuert werden konnte läuft was falsch
# dann sollte der Klient Rechner seine IP's
# nicht mehr benutzen.

authoritative;
# Regelt welcher DHCP Server mehr zu sagen hat. :-)
# Sobald wir fertig sind enablen
```

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

```
option domain-name "local.invalid";
option domain-name-servers 192.168.0.1, 192.168.0.2;
```

Domain Name und die **DNS Server** Übertragen.

config-datei

```
option routers 192.168.0.1;
```

Den **Default-Gateway** angeben.

7 Ausblick

In späteren Versionen soll der Text u.a. um die folgenden Schwerpunkte erweitert werden:

- * DHCP und Sicherheit
- * Ausfallsicherheit durch mehrere Server (failover)
- * DHCP Relay
- * Verknüpfung von DHCP und DNS
- * Verknüpfung von DHCP und NMB (SMB)

Links:

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

RFCs:

<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.linuxwiki.de/DHCP>