5.2 Netzwerk/DHCP

5.2.1 Protokolle zur Zuordnung von MAC-Adressen zu IP-Adressen

Aufgabe von DHCP ist es, zwischen Schicht-2-Adressen (MAC) und Schicht-3-Adressen (IP) zu vermitteln. DHCP liegt damit, was seine Auswirkung betrifft, zwischen den Schichten 2 und 3.

5.2.1.1 RARP Insofern ähnelt DHCP dem Protokoll ARP (*address resolution protocol*). Mit ARP sendet der fragende Rechner einen Schicht-2-Broadcast mit der MAC-Adresse 0xFFFFFF und bekommt die gesuchte MAC-Adresse in der ARP-Antwort zurück. Ähnlich verhält es sich mit dem Gegenteil von ARP, genannt RARP (*reverse address resolution protocol*). Auch mit RARP schickt der fragende Rechner einen Schicht-2-Broadcast. Dieser enthält die eigene MAC-Adresse; als Antwort bekommt der Rechner von einem RARP-Server eine IP-Adresse. Das Problem bei RARP ist, dass der Schicht-2-Broadcast nicht über Router hinweggeleitet wird. Oft möchte man aber die IP-Adressen zentral verwalten; das ist mit RARP nicht so einfach möglich.

5.2.1.2 BOOTP Eine Lösung bietet das Protokoll BOOTP. Zum Routing braucht man mindestens Schicht 3, also IP. Deshalb muss ein Protokoll, das dieses Problem lösen soll, oberhalb von Schicht 3 liegen. BOOTP benutzt die Ports UDP/67 (Server) und UDP/68 (Client).

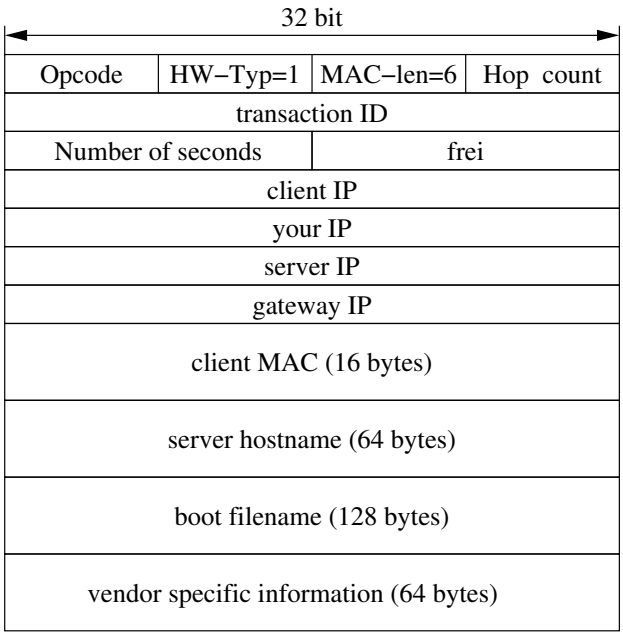

Abbildung 1: BOOTP-Protokoll

Eine graphische Darstellung eines BOOTP-Datagramms zeigt Abbildung 1. Ein BOOTP-Datagramm kann entweder ein Request (Frage) oder ein Reply (Antwort) sein, beide haben eine Länge von jeweils 300 Bytes. Folgende Felder sind wichtig:

a) Opcode: 1=request, 2=reply

b) HopCount: Normalerweise 0, kann durch Proxy erhöht werden

c) Transaction ID: Zufallszahl; sinnvoll falls mehrere Anfragen vorhanden sind

d) Number of seconds: Zeit seit Start des Bootvorgangs

e) Client IP (Quelle): 0.0.0.0, falls nicht bekannt

f) Your IP (Quelle): Vom Server auszufüllen, falls Client-IP=0.0.0.0 ist

g) Server IP (Ziel): Vom Server auszufüllen, ggf. Gateway-IP

h) Client MAC (Quelle): Vom Client anzugeben; der Prozess kommt auf diese Art leichter an die Info als mit Hilfe von Schicht 2

i) Server Hostname: String, optional vom Server auszufüllen

j) Boot Filename: String, optional vom Server auszufüllen; Name einer Datei, die per TFTP geholt werden kann

k) Vendor specific Information: 64 Bytes

Der Request ist ein Schicht-3-Broadcast mit der Source-IP 0.0.0.0 und der Destination-IP 255.255.255.255. Der Reply darf (muss aber nicht) ein Schicht-3-Broadcast sein, und zwar an den Port UDP/68. Das liegt an folgendem Problem: Das Serverprogramm schickt per UDP/IP an die Client-IP die Antwort und sendet dafür vorher eine ARP-Anfrage. Der Client weiß aber zu diesem Zeitpunkt noch gar nicht, dass er die entsprechende IP hat. Deshalb muss der Server entweder vorher seinen eigenen ARP-Cache ergänzen (wozu er aber root sein muss) und verletzt damit das Schichtenmodell oder sendet den Reply als Schicht-3-Broadcast.

Falls der Router ein BOOTP-Relay-Agent ist, setzt er seine eigene IP in das Feld Gateway-IP, erhöht den Hop-Count um 1 und schickt den Request weiter in Richtung des richtigen BOOTP-Servers, jedoch nicht als Broadcast, sondern mit normaler IP-Zieladresse. Damit wird der Request regulär durch das Netzwerk geroutet.

Der BOOTP-Server erkennt anhand des Feldes Gateway-IP, dass es sich um einen umgeleiteten Request handelt. Deshalb schickt er die Antwort zurück an den BOOTP-Relay-Agent. Der wiederum schickt sie weiter an den Client.

Das Feld *vendor specific information* sieht man in Abbildung 2. Im oberen Bereich ist das

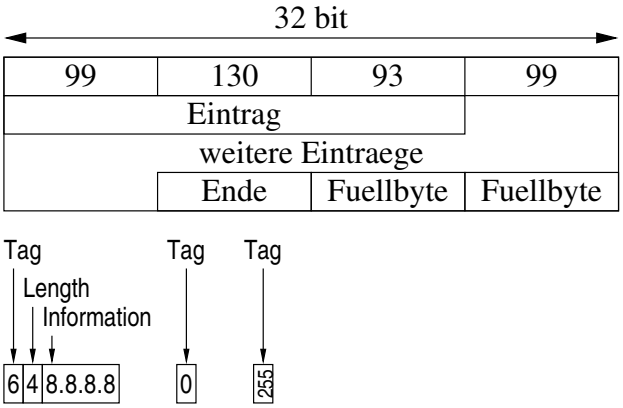

Abbildung 2: *vendor specific information*

gesamte Feld dargestellt, darunter drei Beispiele für Einträge. Jeder Eintrag besteht aus einem Eintrags-Typ (genannt *tag*). Bei allen Eintragstypen außer 0 (Füllbyte) und 255 (Ende-Byte) folgt darauf die Länge der Information in Byte (*tag* und Länge selbst nicht mitgerechnet) und darauf die eigentliche Information. Hier sind ein paar Beispiele:

- tag=0: Füllbyte
- tag=1: Subnetzmaske
- tag=2: Zeit seit 1.1.1900
- tag=3: Gateways

- tag=6: DNS-Server
- tag=255: Ende-Byte (danach nur noch Füllbytes)

5.2.1.3 DHCP Das BOOTP-Protokoll hat einen Mangel: Eine IP-Adresse, die einmal vergeben wurde, kann anschließend nicht mehr an einen anderen Client vergeben werden; es fehlt eine zeitliche Begrenzung. Diesen Mangel behebt das Protokoll DHCP, das als eine Erweiterung zu BOOTP etabliert wurde. Die graphische Darstellung eines BOOTP-Datagramms zeigt Abbildung

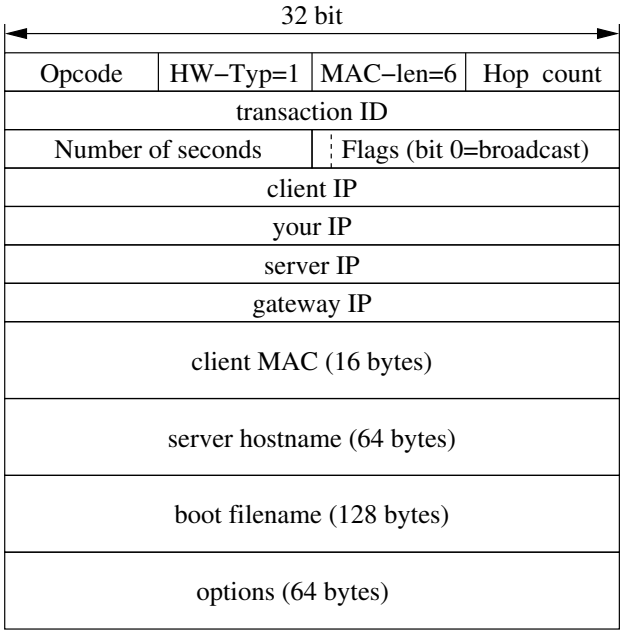

Abbildung 3: DHCP-Protokoll

3. Die einzigen Unterschiede liegen in einem Flag-Feld (dritte Reihe), von dem aber nur ein Bit genutzt wird, und in der Umbenennung der *vendor specific information* in *options*. Aus der Bezeichnug *tag* wird nun *option type*. Alle Erweiterungen von BOOTP zu DHCP sind in diesem Feld implementiert.

Unterschiede ergeben sich im Ablauf: Das Opcode-Feld ist weiterhin auf Request oder Response beschränkt, nun gibt es aber außerdem als eine der Optionen eine *DHCP message type option*. Sie wird durch *option type*=53 festgelegt. Mögliche Werte stehen in Tabelle 1 (nicht vollständig). Ein

Wert	*dhcp message type*	Richtung	Bedeutung
1	discover	`C>S`	brauche eine Adresse
2	offer	`S>C`	biete Adresse an
3	request	`C>S`	nehme angebotene Adresse
4	decline	`C>S`	Adresse ist bereits vergeben
5	acknowledge	`S>C`	Bestätigung eines Requests
6	not acknowledge	`S>C`	Ablehnung eines Requests
7	release	`C>S`	Adresse wird nicht mehr gebraucht
8	inform	`C>S`	

Tabelle 1: *DHCP message type options*

typischer Ablauf einer Vergabe einer IP-Adresse mit DHCP ist so:

```
C>S discover (broadcast)
```

```
S>C offer
C>S request (will Adr. haben/wiederhaben/lease verlaengern)
S>C ack
```

Eine weitere wichtige Option legt die Zeitspanne fest, für die eine Adresse zugewiesen (auf englisch: *lease*) wird. Vor Ablauf der Zeitspanne (Lease-Dauer) muss die Adresszuweisung erneuert werden werden.

5.2.2 DHCP-Clients

Ein DHCP-Client hat zwei Aufgaben:

a) Ausführen des DHCP-Protokolls an einer Netzwerkkarte

b) Setzen der IP-Adresse dieser Karte auf den geholten Wert

Für den ersten Punkt braucht man eine root-Berechtigung, um eine Verbindung an Port 68 anzunehmen (Port-Bereich unter 1024). Für den zweiten Punkt braucht man die root-Berechtigung, um die Netzwerkkarte zu konfigurieren.

Es gibt mehrere Programme, die als DHCP- (und BOOTP-) Clients arbeiten können:

a) dhclient (aus dem Paket isc-dhcp-client)

b) dhcpcd (aus dem Paket dhcpcd5)

c) pump (aus dem gleichnamigen Paket)

Die Programme werden in der Regel im Hintergrund von Netzwerk-Skripten aufgrufen.

5.2.3 DHCP-Server

Wenn man einen DHCP- (oder BOOTP-) Server einrichten will, gibt es ebenfalls mehrere Möglichkeiten:

a) bootpd (aus dem Paket bootp) kennt vor allem BOOTP und nur wenige Einzelheiten von DHCP

b) dhcpd (aus dem Paket isc-dhcp-server) ist ein vollständiger DHCP-Server

c) dnsmasq (aus dem gleichnamigen Paket) kann gleichzeitig DHCP (und BOOTP) verwalten und DNS-Auflösung für lokale Rechner vornehmen

5.2.4 dhcpd als DHCP-Server

Für die Konfiguration muss man zuerst angeben, an welcher Netzwerkkarte auf DHCP-Anfragen gelauscht werden soll. Hierzu gibt es eine Datei `/etc/default/dhcp*-server`, die nur eine einzige Zeile enthält[1]:

```
1 INTERFACES="eth1"
```

Der Rest der Konfiguration von dhcpd geschieht vollständig über die Datei `dchpd.conf`, die im `/etc`-Teilbaum liegt. In dieser Datei gibt es Anweisungen und Blöcke mit geschweiften Klammern. Was innerhalb eines Blockes festgelegt wird, gilt nur dort, alles andere gilt global. Anweisungen bestehen aus einem Befehlswort, durch Leerzeichen getrennten Parametern und zum Schluss entweder einem Semikolon oder einem Block. Zeilen mit einem # am Anfang gelten als Kommentar.

Zunächst gibt es allgemeine Festlegungen:

[1] Bei manchen Distributionen erfolgt die Abfrage der gewünschten Netzwerkkarte schon bei der Installation des Pakets. Dann wird auch die besagte Datei geschrieben.

```
# will einziger DHCP-Server sein, kann auch DHCPNAK senden:
authoritative;
# Dauer eines Leases in Sekunden
default-lease-time 80000;
max-lease-time 160000;
interface eth1;
```

Einträge, die mit `option` beginnen, werden als DHCP-Optionen an die Clients weitergegeben:

```
option domain-name "musterschule.local";
option domain-name-servers 10.1.1.3;
option ntp-servers 10.1.1.1;
# option netbios-name-servers 10.1.1.3;
# option netbios-node-type 8;
```

Mit `subnet` vergibt man Eigenschaften für einen Adressbereich:

```
subnet 10.97.1.0 netmask 255.255.255.0 {
    range 10.97.1.100 10.97.1.149;
    range 10.97.1.200 10.97.1.249;
    option broadcast-address 10.97.1.255;
    option routers 10.97.1.1;
    default-lease-time 43000;
    max-lease-time 86000;
}
```

Will man einem einzelnen Rechner eine feste Adresse geben, nimmt man die `host`-Anweisung:

```
host newssrv {
        hardware ethernet 11:22:33:44:55:66;
        fixed-address 10.97.1.4;
        option host-name "newssrv";
}
```

Ausführliche Information findet man in der Manual-Page zu `dhcpd.conf` (!). Mit der folgenden Befehlszeile kann man prüfen, ob die Konfigurationsdatei in Ordnung ist:

Terminal

```
root@debian964:~# dhcpd -t
```

Nun kann man den Server starten:

Terminal

```
root@debian964:~# /etc/init.d/dhcpd start
```

Bei einer Änderung der Konfiguration muss der `dhcpd` neu gestartet werden:

Terminal

```
root@debian964:~# /etc/init.d/dhcpd restart
```

In der Datei /var/lib/dhcp?/dhcpd.leases kann man im Betrieb die aktuellen Adresszuweisungen (*leases*) finden. Meldungen findet man in `/var/log/messages` (oder in `/var/log/syslog`):

Terminal

```
root@debian964:~# grep dhcpd /var/log/messages | tail -f
```

Den DHCP-Netzwerkverkehr kann man zwar mit den üblichen Werkzeugen beobachten (`wireshark`, `tcpdump`), speziell für DHCP gibt es aber zusätzlich das Programm `dhcpdump` aus dem gleichnamigen Paket:

Terminal

```
root@debian964:~# dhcpdump -i eth1
```

5.2.5 dnsmasq als DHCP-Server

Das Programm `dnsmasq` kann gleichzeitig als DHCP-Server und als DNS-Server für lokale Netze dienen. Die Konfiguration liegt in der Datei `/etc/dnsmasq.conf` und ist im Standardfall sehr einfach:

```
1 # nur in diese Richtung DHCP-Server spielen:
2 interface=eth0
3 # IP-Bereich:
4 dhcp-range=192.168.2.10,192.168.2.99,120h
5 # das waere eine feste IP:
6 #dhcp-host=b8:27:eb:66:35:49,raspi,192.168.2.3,infinite
```

Der Server wird neu gestartet:

Terminal

```
root@debian964:~# etc/init.d/dnsmasq restart
```

Das Ergebnis kann der Client in seiner Datei `/var/lib/dhcp/dhclient.leases` finden, beim Server liegt es in der Datei `/var/lib/misc/dnsmasq.leases`:

```
1 1361901008 b8:27:eb:66:35:49 192.168.2.41 raspi *
```

Die Felder in dieser Datei sind: Datum (in der Form, die durch `date +%s` ausgegeben wird), MAC-Adresse, IP-Adresse und DNS-Name.