TCP/IP – Grundlagen und Praxis

Protokolle, Routing, Dienste, Sicherheit

von
Gerhard Lienemann, Dirk Larisch

2., akt. Aufl.

Thematische Gliederung:

Netzwerkprotokolle, EDI – Netzwerkprotokolle, EDI

Heise Zeitschriften Verlag 2013

Verlag C.H. Beck im Internet:
www.beck.de
ISBN 978 3 944099 02 6

3 Adressierung im IP-Netzwerk

Der Begriff der Netzwerkadressierung oder der Adressvergabe wird an verschiedenen Stellen dieses Buches verwendet. Für das Verständnis eines allgemeinen Sachverhaltes im Zusammenhang mit adressierbaren Netzwerkkomponenten bedarf dieser Ausdruck keiner ausdrücklichen Erklärung. Um Näheres über die Adressierung im IP-Netzwerk zu erfahren, muss man sich detailliert mit dem Konzept und seinen Besonderheiten beschäftigen.

3.1 Adresskonzept

Die TCP/IP-Protokollfamilie kann grundsätzlich unter verschiedenen Netzwerktopologien eingesetzt werden, um damit eine wechselseitige Kommunikation der Teilnehmer eines Netzwerks zu gewährleisten. Allerdings gibt es eine Minimalforderung, die unbedingt erfüllt sein muss: *Die Adressierung der Knoten eines Netzwerks muss eindeutig sein*. So muss jede Netzwerkressource, sei es ein Rechner (PC), eine Workstation, ein Drucker oder auch ein Mainframe-System, durch eine eindeutige Adresse bzw. durch einen eindeutigen Namen im Netzwerk identifizierbar sein. Wenn diese Grundvoraussetzung nicht gewährleistet ist, kann eine Kommunikation zwischen den Kommunikationspartnern nicht stattfinden.

Vergleichbar ist dieses Prinzip mit der Vergabe von Telefonnummern, bei der das Chaos perfekt wäre, wenn mehrere Rufnummern doppelt vergeben würden. Für Gesprächsteilnehmer in Orten mit unterschiedlichen Ortskennziffern (Vorwahl) gilt dies natürlich nicht. Eine gültige Doppelbelegung ist dabei durchaus möglich, da die Eindeutigkeit durch eine stets eindeutige Ortskennziffer garantiert ist.

3.1.1 Adressierungsverfahren

Ganz wichtig ist, dass die physikalische von der logischen Adressierung im Netzwerk unterschieden werden muss. Unter einer *physikalischen Adressierung* versteht man die mit einer Netzwerkressource unverwechselbar verbundene eindeutige Kennung, meist sogar weltweit, die zur Identifikation in einem Netzwerk herangezogen wird. Bei den heute handelsüblichen Netzwerkcontrollern ist eine solche Hard-

wareadresse (MAC-Adresse) fest in einer Komponente hinterlegt, die auch nicht modifiziert werden kann. So werden beispielsweise für Rechner Netzwerkadapter (Einschubkarten) mit *burnt-in addresses* (eingebrannte Adressen) vertrieben, die zwecks Anschluss an ein Netzwerk in die entsprechenden Rechner eingebaut werden können. Das Gleiche gilt für andere Geräte, die teilweise bereits mit fest eingebauten Netzwerkcontrollern ausgestattet sind, wie beispielsweise Smartphones, Tablets oder mittlerweile auch Digitalkameras.

Im Gegensatz dazu ist die *logische Adressierung* in erster Linie vom Netzwerkprotokoll und seinen charakteristischen Eigenschaften abhängig. Sie hat zunächst einmal nichts mit der physikalischen Adresse zu tun und es liegt allein im Verantwortungsbereich des Netzwerkbetreibers, für die notwendige Eindeutigkeit der Adressen zu sorgen. Der Systemverwalter eines Netzwerks ist damit in der Lage, eine für ihn bzw. für das Unternehmen sinnvolle Adressstruktur zu entwickeln und dann die jeweils vorgesehene logische Adresse der physikalischen Adresse »überzustülpen«.

In TCP/IP-Netzwerken kommt zur logischen Adressierung die *IP-Adresse* zum Einsatz. In der IP-Version 4 besteht diese Adresse aus vier Oktetten, die in dezimaler Form (*Dotted Notation*) formuliert wird (Beispiel: 192.205.76.6). Die Weiterentwicklung in diesem Bereich hin zur IP-Version 6 nutzt einen erweiterten Adressraum von 16 Oktetten.

HINWEIS

Nähere Angaben zu IPv6 enthält das Kapitel 10.

Ein oft vernachlässigter Aspekt, insbesondere in der Phase der Netzwerkplanung, ist das System der *Host-Namen* und der *Domänen*. Dabei wird der logischen IP-Adresse zusätzlich ein sogenannter *fully qualified host name* (vollqualifizierter Name) zugeordnet, der aus einer durch Punkt separierten Folge von Ordnungsnamen besteht. Diese Namen setzen sich in der Regel aus einer DNS-Domäne (z.B. *firma.de*) und einem Host-Namen (z.B. *host1*) zusammen. Die Adressierung auf diesem Level kann sogar den einzelnen Anwender auf dem System identifizieren: *anwender@host1.firma.de*. In der Praxis wird dieser Adresstyp meist aus Anwendungen heraus zur Adressierung verwendet.

HINWEIS

Nähere Angaben zu DNS (Domain Name System) und der Zuordnung von Domänen enthält das Kapitel 5.

Zusammenfassend lassen sich für die Adressierung in einem TCP/IP-Netzwerk folgende Layer-abhängige Verfahren nennen:

- physikalische Adressierung (*Lower Data Link Layer*)
- logische IP-Adressierung (*Network Layer*)
- logische Adressierung über Host Names und Domains (*Network Layer*)

In der hier genannten Reihenfolge bedeuten die einzelnen Adressierungsverfahren eine steigende Flexibilität gegenüber Modifikationen im Netzwerk. Der (möglicherweise relativ häufige) Austausch von Netzwerkcontrollern durch Defekt oder Wechsel der Hardware lässt die zugeordnete IP-Adresse davon unberührt. Sie bleibt bestehen, auch wenn sich die Hardware ändert. Werden allerdings Erweiterungen oder Modifikationen in der IP-Netzstruktur notwendig, so ist es möglich, dass dies unter Umständen auch eine Änderung der IP-Adresse nach sich zieht. Der symbolische Name der IP-Adresse oder gar des Anwenders ist davon natürlich nicht betroffen.

3.1.2 Adressregistrierung

Der Ursprung des Internet Protocol (IP) lag in einem vom amerikanischen Verteidigungsministerium initiierten Netzwerk und bedurfte einer besonderen Kontrollinstanz. Diese Instanz stellt das *Internet Architecture Board* (IAB) mit seinen zahlreichen Unterabteilungen und Nebeninstitutionen dar, die für eine reibungslose und sichere Funktion des Internetnetzwerks bis zum heutigen Tag verantwortlich sind. Dazu gehört u.a. die Gewährleistung der Eindeutigkeit von IP-Adressen im öffentlichen Datenverkehr, also in der Anbindung bzw. Integration von Firmennetzwerken in öffentliche Netzwerke. Ein durch die zentrale Registrierungsorganisation IANA (*Internet Assigned Numbers Authority*) durchzuführendes Verfahren sorgt im Auftrag des IAB für die geforderten Adresskonventionen. Weitere Details hinsichtlich Aufgaben, Struktur und Mitglieder des IAB lassen sich unter ihrer Homepage im Internet nachlesen (*www.iab.org*).

Zu Beginn einer Überlegung zum Aufbau eines TCP/IP-Netzwerks sollte grundsätzlich die Frage stehen: Soll das geplante Netzwerk öffentlich registriert werden oder ist es lediglich als ein unternehmensinternes IP-Netzwerk auszulegen? In der Praxis zeigt sich immer wieder, dass es durchaus sinnvoll sein kann, bereits in der Anfangsphase der Netzwerkplanung an eine offizielle Registrierung von Domänen und Subnetzen zu denken und die erforderlichen Planungen durchzuführen.

Eine Alternative zur völligen Netzöffnung ist der Einsatz eines IP-Gateways, das für eine Adressumsetzung der internen Adressen in registrierte Adressen verantwortlich ist (NAT = *Network Address Translation*).

HINWEIS

Obwohl die Internetregistrierung in Eigenregie erfolgen kann, ist es jedoch zu empfehlen, den Gesamtprozess inklusive der Netzwerkplanung in Zusammenarbeit mit einem erfahrenen Service Provider abzuwickeln.

3.1.3 Adressaufbau und Adressklassen

Eine IP-Adresse der IP-Version 4 besteht aus einer 32-Bit-Sequenz, die in vier Gruppen zu je acht Bit aufgeteilt ist. Jede dieser Gruppen wird als *Oktett* bezeichnet; ihre Darstellungsweise ist dezimal. Die logische Gliederung dieser Adresse erfolgt in

eine *Netz-ID* und eine *Host-ID*, wobei der Umfang ihres Adressbereichs aus fünf Adressklassen hervorgeht:

- *Klasse A*
  Klasse A charakterisiert einen Adresstyp, der einen sehr kleinen Adressbereich für die Netz-ID (7 Bit) und einen großen Adressbereich für die Host-ID besitzt. Die Anzahl definierbarer IP-Hosts (IP-Adressen) ist somit extrem hoch. Die Klassifizierung geht eindeutig aus den ersten Bits des ersten Oktetts der Adresse hervor. In dieser Klasse A ist das erste Bit reserviert (besitzt den Wert »0«), sodass lediglich die restlichen sieben Bits für eine Netz-ID zur Verfügung stehen (maximaler Wert also binär »1111111« oder dezimal 128). Daraus ergibt sich eine Anzahl von theoretisch 128 Netzen, von denen jedoch die 0 und 127 wegfallen, sodass letztlich 126 Netzwerke adressiert werden können (siehe Abb. 3–1).

Abb. 3–1 *Schema einer IP-Adresse (Version 4) gemäß Klasse A*

HINWEIS

Die Klasse A steht für Neuzulassungen praktisch nicht mehr zur Verfügung, da sie bereits vollständig durch Registrierungen von Unternehmen der »ersten Stunde« belegt ist.

- *Klasse B*
  In dieser Klasse sind die ersten beiden Bits des ersten Oktetts reserviert. Die »Klassen-Bits« besitzen den Wert »10«. Somit lässt sich maximal der Wert »10111111« für das erste Oktett erzeugen; dies entspricht dezimal der Zahl 191. Aufgrund der Erweiterung der Netz-ID um ein weiteres Oktett können nunmehr alle Netze im Bereich zwischen 128.1 und 191.255 adressiert werden. Dies entspricht einer Anzahl von 16.384 Netzwerken gegenüber 126 bei Klasse A. Die Anzahl von definierbaren IP-Hosts pro Netzwerk liegt in dieser Klasse B bei 65.534 (Beispiel: Die Netz-ID 154.8 umfasst einen Adressbereich von 154.8.0.1 bis 154.8.255.254.). Diese Klasse eignet sich für mittelgroße Netze mit einer mittleren Anzahl von IP-Hosts (siehe Abb. 3–2).

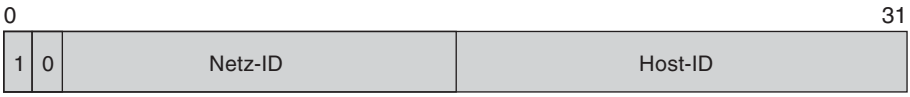

Abb. 3–2 *Schema einer IP-Adresse (Version 4) gemäß Klasse B*

HINWEIS

Genau wie bei Klasse A stehen auch Klasse-B-Adressen für Neuregistrierungen praktisch nicht mehr zur Verfügung.

- *Klasse C*
  Bei *Klasse C* erfolgt eine Reservierung der ersten drei Bits des ersten Oktetts auf die Werte »110«, so wie in Abbildung 3–3 dargestellt. Der maximal erreichbare Wert lautet demnach binär »11011111« und dezimal 223. Für die um ein weiteres Oktett erweiterte Netz-ID ergibt sich ein Anfangswert von 192.0.1 und eine theoretische Obergrenze von 223.255.255. Maximal können daher 2.097.152 Netze abgebildet werden, allerdings lediglich mit jeweils 254 Hosts. Aus diesem Kontingent wurden bzw. werden die meisten Registrierungsanträge befriedigt.

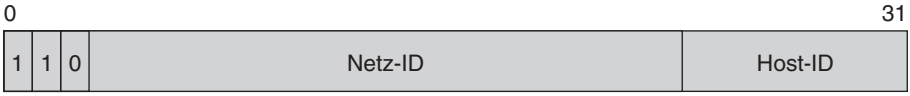

Abb. 3–3 *Schema einer IP-Adresse (Version 4) gemäß Klasse C*

- *Klasse D*
  Bei Klasse D sind die ersten vier Bits des ersten Oktetts auf binär »1110« reserviert (siehe Abb. 3–4). Hier existiert allerdings kein Definitionsbereich für eine Netz-ID. Die hier generierbaren IP-Adressen zwischen 224.0.0.0 und 239.255.255.254 besitzen Sonderstatus, werden normalerweise als »Multicast-Adressen« bezeichnet (siehe auch Kapitel 4 *Routing*) und stehen für besondere Funktionen zur Verfügung.

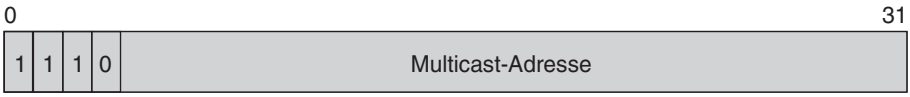

Abb. 3–4 *Schema einer IP-Adresse (Version 4) gemäß Klasse D*

- Klasse E
  Auch die Klasse E genießt Sonderstatus und steht für den Allgemeingebrauch nicht zur Verfügung. Die Reservierung erstreckt sich auch hier auf die ersten vier Bits des ersten Oktetts, allerdings erhält das vierte Bit den Wert »1«, sodass die erste benutzbare Adresse 240.0.0.0 und die letzte Adresse mit 255.255.255.254 angegeben werden kann.

In der nachfolgenden Tabelle sind die verschiedenen Klasseneinteilungen und die damit verbundene Adresszuordnung zusammengefasst:

Klasse	Klassen-Bits	Anzahl Netze	Anzahl Hosts pro Netz
A	0	126	16.777.214 (256^3-2)
B	10	16.384 (64×256^1)	65.534 (256^2-2)
C	110	2.097.152 (32×256^2)	254 (256^1-2)
D	1110	–	–
E	1111	–	–

3.2 Subnetzadressierung

Die Klassifizierung der IP-Adressen gemäß dem im letzten Abschnitt beschriebenen System in Klasse-A-, Klasse-B- und Klasse-C-Adressen (die Sonderklassen D und E bleiben unberücksichtigt, da sie nicht praktisch nutzbar sind) ist relativ starr und unflexibel. Betrachtet man die Entwicklung der öffentlichen Adressvergabe und Registrierung innerhalb der letzten Jahrzehnte, so ist es mittlerweile nahezu unmöglich, eine der überaus attraktiven (öffentlichen) Klasse-A-Adressen zu erhalten; selbst B-Adressen sind nur noch sehr schwer zu bekommen.

Adressen der Klasse C besitzen grundsätzlich einen stark eingeschränkten Adressraum (maximal 254 adressierbare Hosts pro Adresse), sodass man bei der Einrichtung umfangreicher IP-Netzwerke leicht an seine Grenzen stößt. Darüber hinaus scheint sich das Routing innerhalb des öffentlichen Internets zu einem Problem zu verdichten, da der Umfang der Adressinformationen (*Routing-Tabellen*) ein derartiges Ausmaß angenommen hat, dass er von der gegenwärtig verfügbaren Soft- und Hardware nur noch schwer zu verwalten ist. Es ist ferner zu erwarten, dass der zurzeit verfügbare Adressraum von 32 Bit innerhalb der nächsten Zeit sicher nicht mehr ausreichen wird, um den Bedarf an registrierten IP-Adressen zu befriedigen.

HINWEIS

Die Probleme bei der Adressvergabe unter IP, Version 4, und mögliche Lösungsansätze sind unter anderem im RFC 1519 vom September 1993 beschrieben: *Classless Inter-Domain Routing (CIDR): An Address Assignment and Aggregation Strategy*.

Eine mögliche Lösung der Adressproblematik von IP, Version 4, liegt in der Bildung von Subnetzen, die das statische Klassenkonzept durchbrechen. Danach ist es möglich, innerhalb einer verfügbaren Klassenadresse alter Konvention weitere Subnetze zu definieren und die Anzahl adressierbarer Hosts auf die Subnetze aufzuteilen. Dazu passt der RFC 1817 vom August 1995, der die CIDR-Betrachtungen um die Fähigkeiten relevanter Routing-Protokolle erweitert, indem festgehalten wird, dass Protokolle wie RIP, BGP-3, EGP und IGRP nicht für das CIDR-Konzept geeignet sind. Neuere Protokolle, wie OSPF, RIP II, Integrated IS-IS und Enhanced IGRP, lassen sich allerdings verwenden.

HINWEIS

Weitergehende Informationen zum Thema *Routing* und den Funktionen und Möglichkeiten der verschiedenen Routing-Protokolle enthält das Kapitel 4.

Der RFC 4632 vom August 2006 liefert wichtige Informationen zum praktischen Einsatz des CIDR-Konzepts (siehe auch hierzu Aktivitäten der ROAD Workgroup in Abschnitt 10.2.2).

3.2.1 Prinzip

Angenommen ein Unternehmen mittlerer Größe hat zwecks Aufbau eines IP-Netzwerks beim DENIC (*Deutsches Network Information Center*) eine öffentliche Adresse beantragt. Folgende Klasse-B-Adresse wird zugewiesen: 190.136.0.0. Mit dieser Adresse lassen sich bekanntlich 65.536 Hosts adressieren – eine Zahl, die nicht unbedingt dem Bedarfsprofil eines größeren Unternehmens entspricht, das durch den Einsatz von Routern das eigene Netzwerk strukturieren möchte. Für jeden Router muss ein expliziter Netzübergang definiert werden, d.h., der Router umfasst, sofern er lediglich zwei Netzwerke miteinander verbindet, zwei IP-Adressen mit zwei unterschiedlichen Netz-IDs. Da in diesem Beispiel lediglich ein einziges logisches Klasse-B-Netzwerk verfügbar ist, muss das gewünschte Ziel dadurch erreicht werden, dass überall dort, wo Router eingesetzt werden sollen, Subnetze gebildet werden.

3.2.2 Typen der Subnetzmaske

Das zur Bildung von Subnetzen erforderliche Instrumentarium ist die *Subnetwork Mask* (Subnetzmaske). Sie stellt, ebenso wie die IP-Adresse, eine Folge von 32 Bit dar, die in der Regel gemeinsam mit der IP-Adresse auf den einzelnen Systemen (Router, Hosts usw.) konfiguriert wird. Ihre Schreibweise ist der IP-Adresse angepasst: *dotted decimal*. Die Funktionsweise einer Subnetzmaske ist allerdings wesentlich besser nachvollziehbar, wenn man ihren Binärcode betrachtet. Die zu vier Oktetten zusammengefasste »Maske« codiert nämlich überall dort, wo die IP-Adresse als Netz-ID interpretiert werden soll, eine binäre 1. Der restliche Teil der *Subnetwork Mask* bleibt für die Host-ID übrig, also genau die Stellen, wo binäre Nullen codiert sind.

Für den Fall, dass lediglich mit der registrierten IP-Adresse, also ohne »Subnetting«, gearbeitet wird, gilt eine *Default Subnetwork Mask*, denn der IP-Prozess eines Systems berücksichtigt auch dann eine Subnetzmaske, wenn keine eigene Maske definiert wurde. Sie entspricht dann der jeweiligen Klasse, so wie nachfolgend dargestellt:

Klasse	dotted decimal	binary
A	255.0.0.0	11111111 00000000 00000000 00000000
B	255.255.0.0	11111111 11111111 00000000 00000000
C	255.255.255.0	11111111 11111111 11111111 00000000

Soll jedoch eine individuelle Subnetzmaske verwendet werden, so erfolgt die Verteilung der 1er-Maske nicht mehr Byte-, sondern Bit-orientiert:

B	255.255.192.0	**11111111 11111111 11**000000 00000000

3.2.3 Design der Subnetzmaske

Zurück zum Beispiel: Die bei der Klasse-B-Adresse theoretisch verfügbaren 65.536 Hosts sollen auf verschiedene Subnetze aufgeteilt werden. Aufgrund von Überlegungen zur Infrastruktur des gewünschten Netzwerks wird die Entscheidung getroffen, mindestens fünf Subnetze zu bilden (siehe Abb. 3–5).

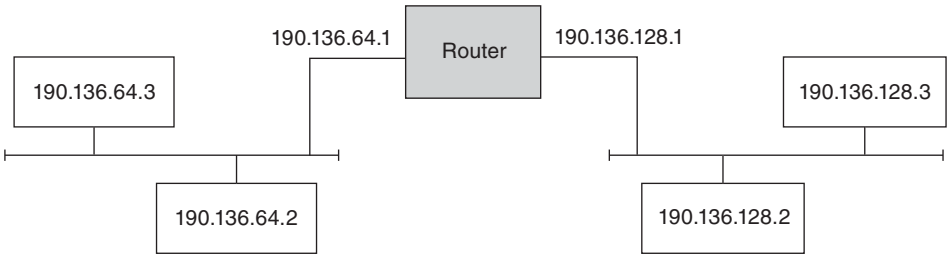

Abb. 3–5 *Aufbau und Struktur des gewünschten Netzwerks*

In diesem Fall bedeutet dies, dass vom dritten Byte der Subnetzmaske die ersten beiden Bits für die Netz-ID verwendet werden sollen; daraus ergibt sich eine Anzahl von maximal vier Subnetzen. Die restlichen sechs Bits des dritten Bytes und das vollständige vierte Byte führen zu einer Anzahl von maximal 16.384 adressierbaren Hosts pro Subnetz (64 x 256; 64 aus dem dritten und 256 aus dem vierten Oktett). Die geplante Subnetwork Mask ergibt daher folgende Teilnetze:

Netz-ID	Host-ID
190.136.0.0	10111110 10001000 **00**000000 00000000
190.136.64.0	10111110 10001000 **01**000000 00000000
190.136.128.0	10111110 10001000 **10**000000 00000000
190.136.192.0	10111110 10001000 **11**000000 00000000

Da die Mindestanzahl von fünf definierbaren Subnetzen nicht erreicht werden kann, muss der Plan, eine Subnetzbildung mit den ersten beiden Bits des dritten Bytes der Subnetzmaske durchzuführen, verworfen werden. Als Ausweg bleibt die Hinzunahme eines weiteren Bits in der Subnetzmaske (im dritten Byte), um mindestens fünf Subnetze erzeugen zu können. Daraus wiederum ergibt sich die folgende Aufteilung:

Netz-ID	Host-ID
Subnetwork Mask:	**11111111 11111111 111**00000 00000000
Subnetz	
190.136.0.0	10111110 10001000 **000**00000 00000000
190.136.32.0	10111110 10001000 **001**00000 00000000
190.136.64.0	10111110 10001000 **010**00000 00000000

→

190.136.96.0	10111110 10001000 **011**00000 00000000
190.136.128.0	10111110 10001000 **100**00000 00000000
190.136.160.0	10111110 10001000 **101**00000 00000000
190.136.192.0	10111110 10001000 **110**00000 00000000
190.136.224.0	10111110 10001000 **111**00000 00000000

Aus dieser Aufstellung geht hervor, dass die angestrebten fünf Subnetze gebildet werden können (maximal acht Subnetze). In jedem Subnetz müssen zwei Adressen gestrichen werden, wobei es sich dabei theoretisch jeweils um die erste (z.B. 190.136.96.0) und die letzte Adresse eines Subnetzes (z.B. 190.136.127.255) handelt. Dies ist notwendig, da der Wert »0« das jeweilige Subnetz bezeichnet und »255« die lokale Broadcast-Adresse repräsentiert, also beide für eine Adressierung nicht zur Verfügung stehen.

Unter diesen Voraussetzungen können als Ergebnis für die weitere Planung der Subnetzstruktur folgende Bereiche verwendet werden:

Nr.	Netz-ID	erster Host	letzter Host	Local Broadcast
1	190.136.0.0	190.136.0.1	190.136.31.254	190.136.31.255
2	190.136.32.0	190.136.32.1	190.136.63.254	190.136.63.255
3	190.136.64.0	190.136.64.1	190.136.95.254	190.136.95.255
4	190.136.96.0	190.136.96.1	190.136.127.254	190.136.127.255
5	190.136.128.0	190.136.128.1	190.136.159.254	190.136.159.255
6	190.136.160.0	190.136.160.1	190.136.191.254	190.136.191.255
7	190.136.192.0	190.136.192.1	190.136.223.254	190.136.223.255
8	190.136.224.0	190.136.224.1	190.136.255.254	190.136.255.255

Jedes der acht Subnetze kann somit 8.190 Hosts adressieren (8.192, um die Netz-ID und den lokalen Broadcast reduziert). Die Wahl einer anderen Subnetzmaske könnte eine völlig andere Aufteilung ergeben. Wird beispielsweise nicht die Subnetzmaske 255.255.192.0, sondern 255.255.255.240 verwendet, ergeben sich folgende Subnetze:

Netz-ID	Host-ID
Subnetwork Mask:	**11111111 11111111 11111111 1111**0000
Subnetz	
190.136.0.0	10111110 10001000 00000000 **0000**0000
190.136.0.16	10111110 10001000 00000000 **0001**0000
190.136.0.32	10111110 10001000 00000000 **0010**0000
190.136.0.48	10111110 10001000 00000000 **0011**0000
190.136.0.64	10111110 10001000 00000000 **0100**0000
...	

→

...	
190.136.0.224	10111110 10001000 00000000 **1110**0000
190.136.0.240	10111110 10001000 00000000 **1111**0000

Analog erfolgt die Subnetzbildung für die weiteren Werte 2 bis 255 im dritten Oktett. Es können also bei einem vorliegenden Klasse-B-Netzwerk 190.136.0.0 durch *Subnetting* mit der Mask 255.255.255.240 insgesamt 4.096 Subnetze (256×16; 256 aus dem dritten und 16 aus dem vierten Oktett) mit jeweils 16 minus 2, also 14 IP-Adressen gebildet werden.

3.2.4 Verwendung privater IP-Adressen

Neben den öffentlich vergebenen und genutzten IP-Adressbereichen gibt es eine Vielzahl privater Subnetze, die ausschließlich zur internen Verwendung in Unternehmensnetzwerken genutzt werden können. Diese werden im Internet nicht berücksichtigt und können daher auch mehrfach vergeben werden, da sie ja die Unternehmensnetze nicht verlassen. Bei diesen reservierten privaten Adressbereichen handelt es sich um folgende Subnetze:

- Klasse A: 10.0.0.0/255.0.0.0
- Klasse B: 172.16.0.0/255.240.0.0
- Klasse C: 192.168.0.0/255.255.0.0

Beim Verlassen des Unternehmensnetzwerks bzw. bei der Anbindung an das öffentliche Netzwerk (Internet) erfolgt in der Regel ein »Übersetzen« der privaten Adressen auf öffentliche, legal zugewiesene Adressen. Ein Verfahren, das in diesem Zusammenhang eingesetzt wird, ist das Prinzip der *Network Address Translation* (NAT), also einer Art »Übersetzungstabelle«.

Werden beispielsweise zwei Standorte eines Unternehmens über IP-Router verbunden, wobei die beiden Standorte über ein eigenes Netzwerk verfügen, die wiederum mit einem Standort-Backbone verbunden sind, so erfolgt über diesen Backbone die WAN-Verbindung über einen dedizierten Router. Für das interne IP-Netzwerk wird die Klasse-A-Adresse 10.0.0.0 verwendet. Für die Strukturierung in verschiedene Subnetze wird die Subnetzmaske 255.255.255.0 verwendet. Pro Abteilung (drittes Oktett) innerhalb eines Standortes (zweites Oktett) stehen somit maximal 254 Host-IDs (viertes Oktett) zur Verfügung. Für den jeweiligen Standort lassen sich 255 verschiedene Abteilungsnetze etablieren. Für den Expansionsdrang des Unternehmens ist ebenfalls gesorgt: 255 Standorte können für die Zukunft mit einer eigenen Netz-ID adressiert werden. Es ergibt sich also folgende Adressstruktur:

Standort	Abteilung	Netz-ID
Hamburg	Auftragsbearbeitung	10.2.1.0
Hamburg	Buchhaltung	10.2.2.0
Hamburg	Versand	10.2.3.0
Hamburg	Datenverarbeitung	10.2.4.0
München	Auftragsbearbeitung	10.3.1.0
München	Buchhaltung	10.3.2.0
München	Versand	10.3.3.0
München	Datenverarbeitung	10.3.4.0
WAN-Verbindung der Standorte		10.1.1.0

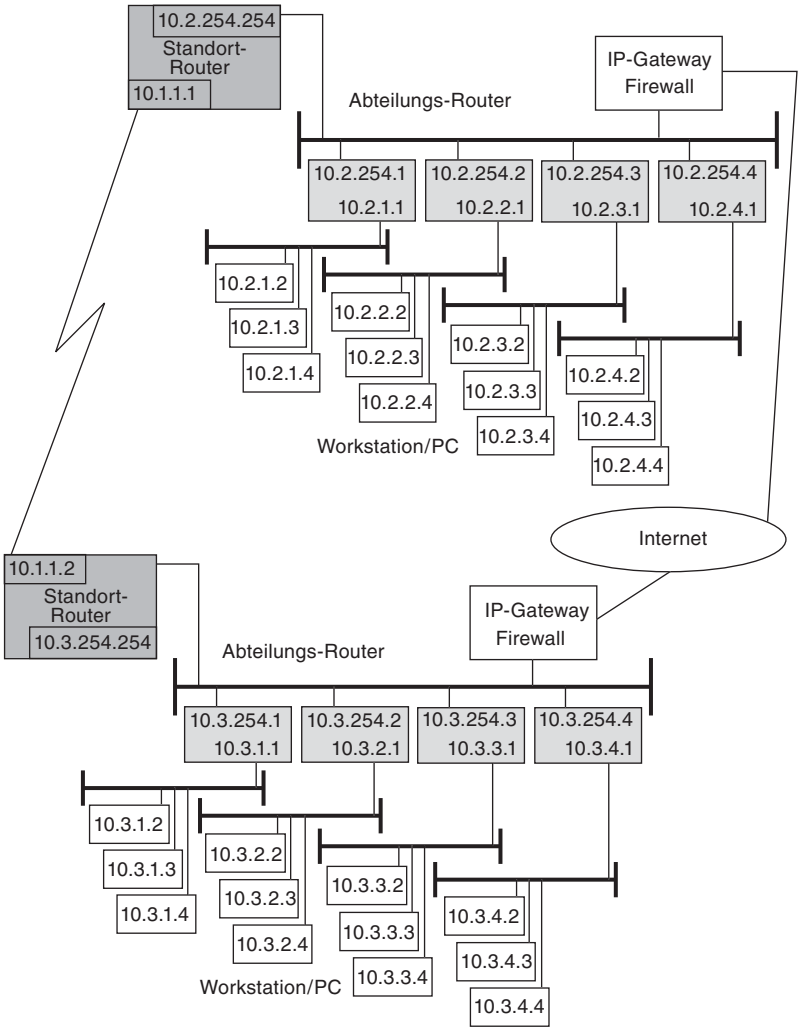

Abb. 3–6 *Beispiel für eine Adressstruktur bei einer WAN-Anbindung zweier Standorte*

3.2.5 Internetdomain und Subnetz

Ein erster Schritt zur eigenen Internetdomain ist zumeist die Kontaktaufnahme mit einem entsprechenden *Provider* bzw. *Hoster*, der in der Regel alle erforderlichen Formalitäten übernimmt. Somit entfällt ein direkter Vertragsabschluss mit dem DENIC, dem *Deutschen Network Information Center*. Das DENIC ist innerhalb Deutschlands für die kontrollierte Vergabe von Subdomains unterhalb der Top-Level-Domain »de« zuständig.

Will man eigene Internetdienste betreiben bzw. eigene Server ins Internet bringen (z.B. einen FTP- oder Webserver), so werden dazu legale Internetadressen benötigt. Die Vergabe privater Adressen (gemäß RFC 1918) ist zu diesem Zweck nicht möglich, da man schließlich für weltweite Eindeutigkeit dieser IP-Adressen sorgen muss. Diese kann aber nur dann gewährleistet werden, wenn eine zentrale Instanz darüber wacht, dass eine IP-Adresse stets nur einmal vergeben wird. Für diese Aufgabe war ursprünglich die IANA (*Internet Assigned Numbers Authority*; siehe auch www.iana.org) zuständig. Ihre Aufgaben werden aber zwischenzeitlich von der ICANN (*Internet Corporation for Assigned Names and Numbers*) als sogenannter Non-Profit-Organisation wahrgenommen.

Die ICANN verwaltet weltweit zentral eine Datensammlung von IP-Adressen und Subnetzen, wobei sie die Vergabe in anderen Ländern an weitere Instanzen delegiert. Dabei handelt es sich um das APNIC (*Asia-Pacific Network Information Center*; www.apnic.net), das ARIN (*American Registry for Internet Numbers*; www.arin.net) und das RIPE NCC (*Réseaux IP Européens*; www.ripe.net). Letzteres übernimmt diese Aufgabe für den europäischen Bereich des Internets.

Die wichtigsten Informationen zur IP-Adressierung, -Organisation und zu ihren Besonderheiten sind in den RFCs 2050 (*Internet Registry IP Allocation Guidelines*), 1918 (*Address Allocation for Private Internets*) und 1518 (*An Architecture for IP Address Allocation with CIDR*) hinterlegt. Da mit der Domainbeschaffung für Privatpersonen normalerweise kein eigenes IP-Subnetz (mehrere öffentliche IP-Adressen, die ausschließlich für den Antragsteller reserviert sind) erworben wird und der Kunde für den Internetzugang (nicht für die Bereitstellung eigener Dienste) dynamische IP-Adressen aus dem eigenen Kontingent des Providers erhält, wird der Prozess der Domain-Beantragung und Reservierung vom Provider meist in Eigenregie durchgeführt.

3.3 Dynamische Adressvergabe

Neben der statischen Zuordnung von Adressen für den Bereich der TCP/IP-Protokollfamilie gibt es je nach Anforderung auch andere Formen der Adresszuordnung, wie beispielsweise eine dynamische Vergabe der Adressen. Die beiden grundlegenden oder bekanntesten Verfahren sind dabei BootP und DHCP, wobei sich die Zuordnung nicht nur auf eine IP-Adresse bezieht, sondern damit auch weitere Angaben zur Netzwerkkonfiguration übermittelt werden können.

HINWEIS

Innerhalb von IP-Netzwerken folgt die Adressvergabe grundsätzlich sehr stringenten Vorgaben, die beispielsweise festlegen, dass in einem Netzwerk (IP-Segment) keine IP-Adresse doppelt vergeben werden darf. Sind IP-Adressen doppelt vorhanden, führt dies in der Regel zu den merkwürdigsten Effekten bis hin zum Ausfall der betreffenden Endgeräte.

3.3.1 Bootstrap Protocol (BootP)

Das *Bootstrap Protocol* (BootP) wurde als UDP-basierendes Protokoll ausschließlich entwickelt, um Startvorgänge (Booten) zu realisieren, die dazu benötigt werden, sogenannte *Diskless Workstations* (Rechner ohne Disketten- und Festplattenlaufwerke) als Arbeitsstationen im Netzwerk zu betreiben. Der Schwerpunkt lag bzw. liegt dabei auf dem Einsatz in UNIX-Umgebungen.

Mit BootP wird nicht der eigentliche Netzwerkstart (Booten) durchgeführt, sondern lediglich die Übernahme relevanter Netzwerkkonfigurationsdaten (z. B. eigene IP-Adresse oder IP-Adresse des Boot-Servers). Der Ladevorgang selbst wird mit klassischen Transferprotokollen durchgeführt, bei denen es sich meistens um TFTP handelt.

HINWEIS

Nähere Informationen zu TFTP (*Trivial File Transfer Protocol*) enthält das Kapitel 6.

Der BootP-Client beginnt seinen Boot-Vorgang mit einem IP-Broadcast, also einer Abfrage an alle Geräte im Netzwerk. Dabei gibt er die Hardwareadresse (MAC-Adresse) des eigenen Netzwerkcontrollers an. Derjenige Boot-Server, der das Gerät erkennt, antwortet mit einem *Reply* und schickt dem Client die für ihn vorgesehene IP-Adresse, den Namen (bzw. die IP-Adresse) des Boot-Servers und der Boot-Datei, die geladen werden soll. Nach Abschluss des nachfolgenden Ladevorgangs ist die *Diskless Workstation* ein vollwertiger IP-Knoten und kann an der Netzwerkkommunikation teilnehmen.

Die Aktivitäten des BootP-Clients (RFC 951) sind am einfachsten folgendermaßen zu charakterisieren:

- *Name*
  Für die letztlich physikalische Adressierung in einem Netzwerk ist die sog. Hardwareadresse verantwortlich, die zumeist im ROM des Netzwerkcontrollers eingebrannt ist (*burnt-in address*). Diese Adresse muss der Rechner lesen können, bevor er sich ins Netzwerk begibt, wobei der Leseprozess durch Aktivierung eines separaten BootP-Vorgangs vorgenommen wird.

- *Netzwerkfähigkeit*
  Über einen *BootP Request* wird netzwerkweit die Frage gestellt: *Wer kennt mich?* Der eigene Name (*burnt-in*) ist Bestandteil der Anfrage. Ist der Name einem anderen Rechner bekannt (per Definition), so erfolgt ein *Reply* an den anfragenden Rechner. Dieser enthält die eigene IP-Adresse, die IP-Adresse des BootP-Servers, ggf. die IP-Adresse eines Routers und den Namen der Datei, die gebootet werden soll. In einigen Maschinen lassen sich diese Informationen auch manuell vorkonfigurieren. Mit dieser Basiskonfiguration lassen sich nun auch *ARP-Requests/-Replies* bearbeiten und TFTP kann benutzt werden.
- *Weitere Angaben*
  Nachdem alle notwendigen Informationen für den Boot-Vorgang vorliegen, kann dieser über TFTP vorgenommen werden. Während des Ladeprozesses erfolgt die Übernahme weiterer wichtiger Netzwerkinformationen. Der eigene Rechner wird initialisiert. Im nächsten Schritt wird durch die Etablierung von Netzwerkverbindungen (z.B. TELNET über TCP) und Sitzungen, die eine grafische Oberfläche präsentieren (z.B. X11/Motif), eine funktionsfähige Arbeitsumgebung hergestellt. In Abbildung 3–7 ist die *BootP Message* dargestellt.

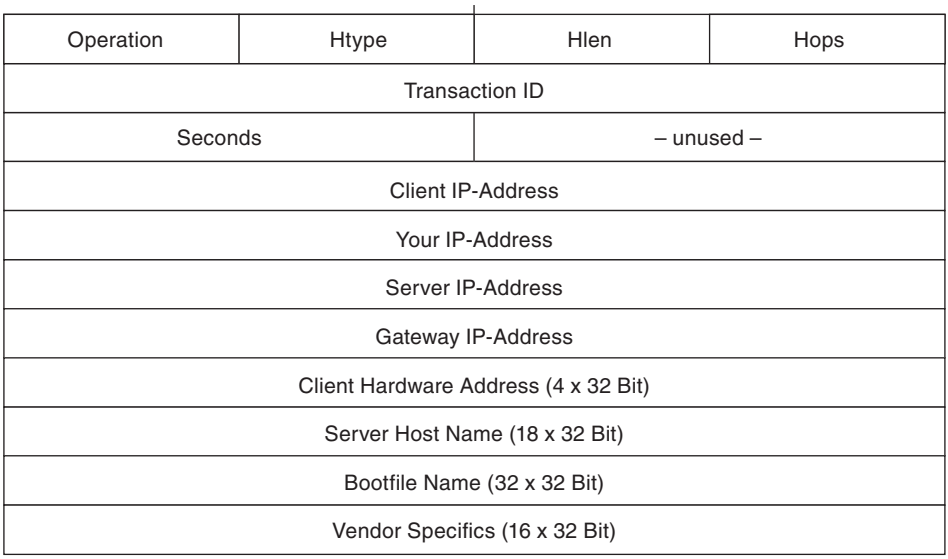

Abb. 3–7 *Aufbau der BootP Message*

Die einzelnen Einträge haben folgende Bedeutung:

- *Operation* (8)
  Wert 1 Boot-Request; Wert 2 Boot-Reply
- *Htype* (8)
  Hardwareadresstyp (z.B. Wert 1 für Ethernet)

- *Hlen* (8)
  Länge der Hardwareadresse
- *Hops* (8)
  Bei Clients wird hier der Wert 0 eingetragen. Wenn netzwerkübergreifende Boot-Vorgänge durchgeführt werden sollen, wird hier die Anzahl relevanter Hops (Netzwerkübergänge) eingetragen.
- *Transaction ID* (32)
  Die nach dem Zufallsprinzip generierte Transaction-ID ist für die Zuordnung von Request und Reply zuständig.
- *Seconds* (16)
  Hier wird vom Client die seit dem ersten Boot-Versuch verstrichene Zeit in Sekunden eingetragen.
- *Client IP-Address* (32)
  IP-Adresse (Version 4) des Clients, wenn er diese kennt
- *Your IP-Address* (32)
  IP-Adresse des Clients, wenn er diese nicht kennt (wird vom Server eingetragen)
- *Server IP-Address* (32)
  IP-Adresse des Servers, die er hier in einen Boot-Reply einträgt
- *Gateway IP-Address* (32)
  IP-Adresse des Gateways (bei Netzübergängen)
- *Client Hardware Address* (128)
  Die vom Client eingetragene Hardwareadresse
- *Server Host Name* (512)
  Dieser (optionale) Eintrag enthält den Host-Namen des Servers, der mit einer binären Null terminiert wird.
- *Bootfile Name* (1024)
  Hier wird der NULL-terminierte Bootfile-Name eingetragen. Der Boot-Request enthält zunächst einen generischen Namen, wie UNIX oder DOS, um den betriebssystemspezifischen Reply generieren zu können. Der Reply enthält den vollqualifizierten Namen des Bootfiles (vollständige Pfadangabe).
- *Vendor Specifics* (512)
  Ermöglicht die Angabe herstellerspezifischer Angaben.

3.3.2 Dynamic Host Configuration Protocol (DHCP)

Neben BootP (siehe vorhergehenden Abschnitt) stellt DHCP (*Dynamic Host Configuration Protocol*) einen weiteren Ansatz für eine dynamische Adressvergabe in einem IP-basierten Netzwerk dar. Dabei lässt sich festhalten, dass DHCP heutzutage das aktuellere Verfahren darstellt und auch an neue Technologien wie IPv6,

angepasst wird (siehe hierzu auch neuere RFCs wie beispielsweise der RFC 6653 vom Juli 2012 »DHCPv6 Prefix Delegation in LTE Networks«).

HINWEIS

DHCP stellt eine für die Verwaltung eines IP-basierenden Netzwerks interessante Technologie dar, die im RFC 1541 vom Oktober 1993 beschrieben ist.

Einführung

Der in den 90er Jahren des vorigen Jahrhunderts einsetzende »Internet-Boom« hatte den Nachteil, dass das zur Verfügung stehende Kontingent an öffentlich registrierbaren IP-Adressen bereits nach wenigen Jahren deutlich reduziert war. Das in Klassen eingeteilte IP-Adress- und Netzwerkkonzept bestand (unter IP v4) aus einem Adressraum von 32 Bit. Je nach Klassen- bzw. Subnetzeinteilung konnten innerhalb eines Subnetzes zwischen 254 bis 16,7 Millionen einzelne IP-Rechner adressiert werden. Dies schien auf den ersten Blick zwar eine unüberschaubar große Anzahl an Adressierungsmöglichkeiten, allerdings waren diejenigen Subnetze bereits seit Jahren vergeben, die eine hohe Anzahl von IP-Adressen bereithalten (Klasse-A-Adressen). Selbst Klasse-B-Adressen waren sehr schnell rar und eher selten zu bekommen. Als dann das Restkontingent von Klasse-C-Adressen ebenfalls zu versiegen drohte, wurden neue Möglichkeiten entwickelt, um den unmittelbar bevorstehenden »Internet-GAU« abzuwenden.

Dazu wurden durch eine klassenübergreifende Neustrukturierung der IP-Adressen (*Classless Inter-Domain Routing* – CIDR) individuelle Adresskonzepte möglich, die den Anforderungen der Netzwerke gerecht werden konnten. Die nächste Entwicklungsstufe führte in eine flexiblere Adressgestaltung, die eine bedarfsorientierte Adressierung ermöglicht, bei der nicht jeder IP-Rechner von vornherein eine fest zugeordnete IP-Adresse erhält. Vielmehr wird ihm erst nach Einschalten und Zugriff auf das Netzwerk eine Adresse reserviert. Dabei geht dieses Verfahren von der Annahme aus, dass nicht alle Rechner eines Unternehmens gleichzeitig kommunizieren bzw. in Betrieb sind. Es reicht also aus, ein entsprechend verringertes Kontingent an Adressen vorzuhalten, um die erforderliche Kommunikationsleistung zu erreichen. Diese dynamische Adressvergabe wird durch das *Dynamic Host Configuration Protocol* (DHCP) realisiert (siehe Abbildung 3–8).

HINWEIS

Nähere Angaben zur Weiterentwicklung des IP auf Basis der Version 6 (*IP Next Generation*) enthält das Kapitel 10.

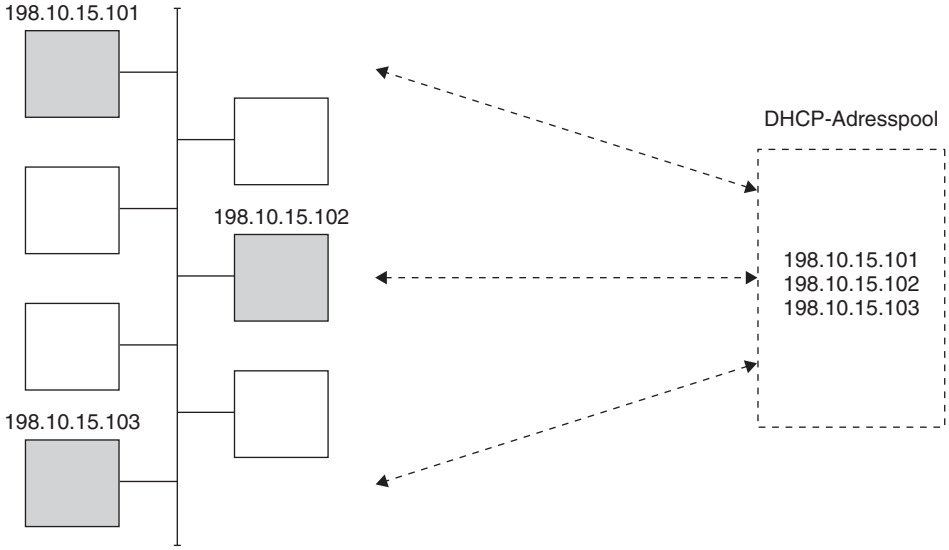

Abb. 3–8 *Prinzip des DHCP*

Einsatzweck

Der Einsatz von DHCP kann für das interne Netzwerk eines Unternehmens oder einer Organisation vor allem administrative Vorteile haben. Für die Realisierung einer umfassenden IP-Kommunikation müssen in mittleren bis großen Netzwerken oft mehrere hundert bis tausend IP-Adressen verwaltet werden. Dies bedeutet eine Vielzahl von Aktivitäten, die von mindestens einem Mitarbeiter in täglich anfallender Routine ausgeführt werden müssen. Dies sind im Einzelnen:

- Vergabe neuer IP-Adressen (Netzwerkerweiterung)
- Löschen von IP-Adressen (Ressourcenabbau)
- Anpassung bei Änderungen der Infrastruktur
- Netzwerkkonfiguration der IP-Clients

Sofern sich der Vorgang der Netzwerkanbindung weitgehend automatisieren lässt, kann ein Großteil der beschriebenen Aktivitäten entfallen. Dabei unterstützt DHCP drei Mechanismen zur IP-Adresszuordnung:

- *Automatischer Mechanismus*
  Sorgt für eine permanente IP-Adresszuordnung.
- *Dynamischer Mechanismus*
  Stellt einem IP-Host lediglich für einen begrenzten Zeitraum eine IP-Adresse zur Verfügung.
- *Manueller Mechanismus*
  Erfordert die manuelle Zuordnung durch den Systemverwalter, wobei DHCP dabei lediglich für den Transport der IP-Adresse benötigt wird.

Die Wahl des jeweiligen Verfahrens (oder von Kombinationen) wird nicht zuletzt durch Vorgaben oder Festlegungen in der Netzwerk- und Systemverwaltung bestimmt. Allerdings besitzt der *dynamische Mechanismus* eine besondere Bedeutung, da nur in dieser Variante durch die Wiederverwendung von temporär benutzten IP-Adressen eine optimierte Ressourcenverwaltung möglich ist.

Merkmale und Format

Das Format einer DHCP-Nachricht basiert auf Nachrichten des Bootstrap-Protokolls (BootP; siehe vorhergehender Abschnitt). Der kommunikationsbereite Rechner wird eingeschaltet und versendet im Netzwerk *BootP Requests*. Diese *Requests* werden vom zuständigen DHCP-Server registriert und mit der Zusendung von IP-Konfigurationsdaten beantwortet. Um nicht in jedem IP-Subnetz einen eigenen DHCP-Server installieren zu müssen, werden sogenannte *BootP Relay Agents* eingerichtet, die eine Weiterleitung der *BootP Requests* vornehmen. Auf der Ebene der Router werden zu diesem Zweck sogenannte *Helper-Adressen* konfiguriert, die eine Weiterleitung der sonst nicht übertragenen DHCP-Pakete ermöglichen.

DHCP wird dazu verwendet, IP-Hosts (keine Router) mit den zur Kommunikation erforderlichen Konfigurationsparametern zu versorgen (eigene IP-Adresse, Standard-Gateway usw.). Nachdem diese Parameter per DHCP empfangen wurden, ist ein IP-Host (z. B. Rechner) in der Lage, mit anderen Hosts Datenpakete auszutauschen. Es werden nicht immer sämtliche Parameter zur Host-Initialisierung benötigt; die tatsächlich erforderlichen Parameter werden vielmehr zwischen DHCP-Client und -Server ausgehandelt. Somit werden beim Einsatz von DHCP folgende Ziele verfolgt:

- Kontrolle der Konfigurationsparameter und lokaler Ressourcen
- keine manuelle Host-Konfiguration
- Vermeidung von DHCP-Serverinstallationen auf jedem einzelnen Netzwerk durch Einrichtung von BootP/DHCP Relay Agents
- DHCP-Clients müssen in der Lage sein, Antworten mehrerer DHCP-Server verarbeiten zu können.
- Koexistenz mit statisch konfigurierten IP-Hosts
- DHCP muss mit BootP Relay Agents zusammenarbeiten können.
- Bereits existierende BootP-Clients müssen durch DHCP ebenfalls bedient werden können.
- keine doppelte Vergabe von IP-Adressen

Protokollspezifikationen

Aus der Sicht eines Clients (IP-Hosts) stellt DHCP eine Erweiterung des Boot-Protokolls dar. Daher können auch bestehende BootP-Clients mit DHCP-Servern kommunizieren, ohne explizit dafür konfiguriert worden zu sein. Abbildung 3–9 zeigt die Struktur einer DHCP-Nachricht. Der Aufbau von Frames bzw. Datenpaketen

wird normalerweise vertikal zu jeweils 32 Bit skizziert (jedes Byte wird dabei auch *Oktett* genannt).

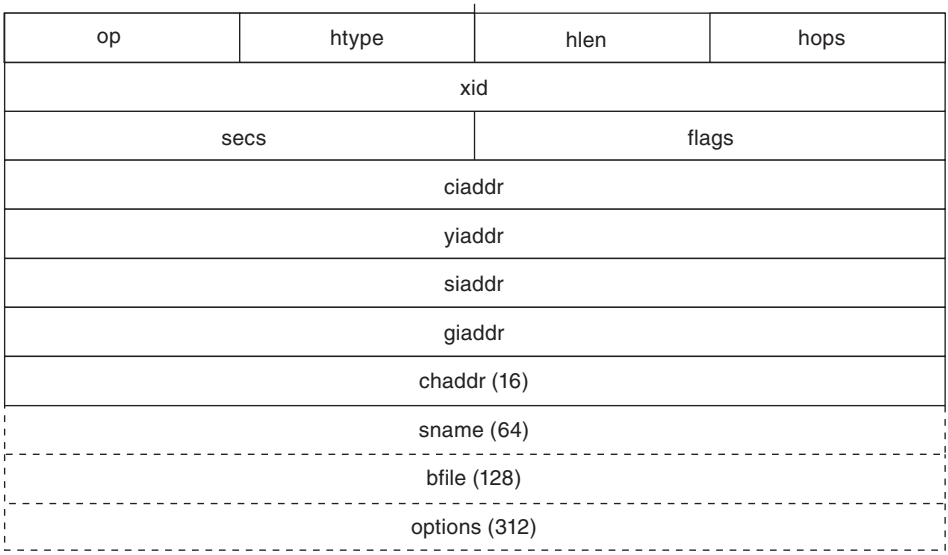

Abb. 3–9 *Aufbau und Struktur einer DHCP-Nachricht*

Die einzelnen Felder der DHCP-Nachricht haben folgende Bedeutung (die Klammerwerte stellen die Feldgröße in Bit dar):

- *op (8)*
  Angabe zur Art der Meldung (*message operation code/message type*; 1 = bootrequest, 2 = bootreply)
- *htype (8)*
  Typ der Hardwareadresse (siehe RFC 1340 *Assigned Numbers*)
- *hlen (8)*
  Länge der Hardwareadresse
- *hops (8)*
  Bei Clients wird hier der Wert 0 eingetragen. Wenn über *Bootp Relay Agents* gebootet wird, so wird hier die Anzahl der Netzübergänge (Hops) eingetragen.
- *xid (32)*
  Eine nach Zufallsprinzip generierte Transaktionsidentifikation sorgt für eindeutige Zuordnungen von *Request* und *Reply* innerhalb der Client-Server-Kommunikation.
- *secs (16)*
  Hier wird vom Client die Zeit (in Sekunden) eingetragen, die seit seinem ersten Boot-Versuch verstrichen ist.

- *flags (16)*
  Dieses Feld erhält vom Client im ersten Bit den Wert 1. Dieses Bit (*Broadcast Flag*) ist für die Auswertung durch DHCP-Server und *BootP Relay Agents* von Bedeutung. Alle weiteren Bits dieses Feldes erhalten den Wert 0.
- *ciaddr (32)*
  IP-Adresse des Clients, die in einem DHCPREQUEST eingetragen wird, um zuvor übernommene Konfigurationsparameter zu verifizieren
- *yiaddr (32)*
  Your (Client) IP-Address. Wenn der Client seine IP-Adresse nicht kennt, wird diese vom DHCP-Server hier eingetragen.
- *siaddr (32)*
  IP-Adresse des nächsten DHCP-Servers, der in einen *bootreply* als DHCPOFFER, DHCPACK oder DHCPNAK einträgt
- *giaddr (32)*
  IP-Adresse des *BootP Relay Agent*, sofern über ihn gebootet werden soll
- *chaddr (128)*
  Hardwareadresse des Clients
- *sname (512)*
  Dieser (optionale) Eintrag enthält den DHCP-Servernamen und wird mit einem NULL-String terminiert.
- *file (1024)*
  Hier wird der NULL-terminierte Bootfile-Name eingetragen. Der *bootrequest* (DHCPDISCOVER) enthält zunächst einen generischen Namen. In den *Reply* wird der (betriebssystemabhängige) vollqualifizierte Pfadname des Bootfiles eingetragen.
- *options (512)*
  Angabe von Optionen (z.B. Herstellerangaben)

Der folgenden Aufstellung sind jeweils die einzelnen DHCP-Nachrichtentypen zu entnehmen:

Nachrichtentyp	Beschreibung
DHCPDISCOVER	Client-Broadcast zur Lokalisierung verfügbarer Server
DHCPOFFER	Antwort des DHCP-Servers auf ein DHCPDISCOVER mit der Angabe von Konfigurationsparametern
DHCPREQUEST	Client-Broadcast an DHCP-Server zur Anforderung angebotener Parameter von einem dedizierten Server bei gleichzeitiger Ablehnung der angebotenen Parameter aller anderen Server
DHCPACK	Server schickt Client-Konfigurationsparameter einschließlich der festgelegten IP-Adresse.

→

Nachrichtentyp	Beschreibung
DHCPNAK	Server lehnt Konfigurationsparameter-Anforderung (auch IP-Adresse) des Clients ab.
DHCPDECLINE	Client informiert Server, dass Konfigurationsparameter bzw. IP-Adresse ungültig sind.
DHCPRELEASE	Client informiert Server, dass er nun die IP-Adresse nicht mehr benötigt.

Funktionsweise

Anhand eines Beispiels soll das DHCP-Verfahren erläutert werden. Es wird beschrieben, wie eine Netzwerkadresse in sechs Phasen dem anfragenden DHCP-Client zugeordnet wird.

- *Phase 1*
  Der Client versendet innerhalb seines Subnetzes (Subnet-Broadcast) eine DHCP-DISCOVER-Nachricht (siehe Abb. 3–10). Diese enthält ggf. Optionen für einen IP-Adressvorschlag und einen Zuordnungszeitraum (im Originaltext wird dieser *Lease* genannt). BootP Relay Agents reichen diese Nachricht nicht nur an DHCP-Server innerhalb, sondern auch außerhalb des lokalen Subnetzes weiter.

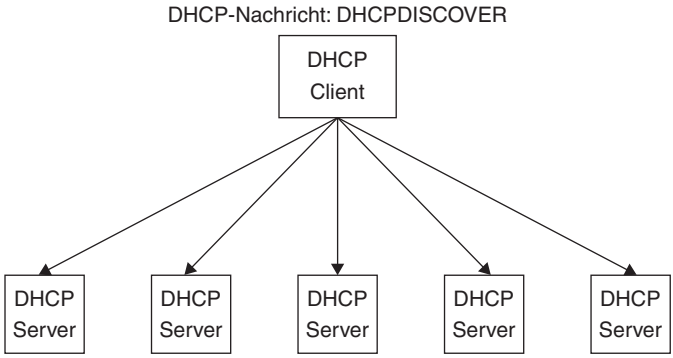

Abb. 3–10 *IP-Adresszuordnung Phase 1*

- *Phase 2*
  Jeder verfügbare DHCP-Server antwortet mit einer DHCPOFFER-Nachricht, die entweder direkt oder per *Subnet-Broadcast* an den Client übertragen wird (siehe Abb. 3–11). Die Antwort enthält eine verfügbare Netzwerkadresse und weitere Konfigurationsparameter in den DHCP-Optionen.

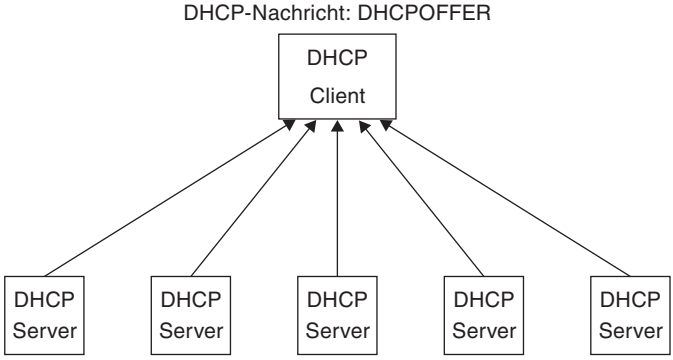

Abb. 3–11 *IP-Adresszuordnung Phase 2*

- *Phase 3*

  Der Client erhält eine oder mehrere DHCPOFFER-Nachrichten von einem oder mehreren Servern. Er darf allerdings auf mehrere Antworten warten, bevor er sich an einen konkreten Server wendet, von dem er dann die Konfigurationsparameter gemäß DHCPOFFER-Nachricht anfordert. Nun verschickt der Client eine DHCPREQUEST-Nachricht per *Broadcast*, die den *server identifier* als Option beinhalten muss. Dies ist erforderlich, damit der vom Client ausgewählte Server exakt identifiziert werden kann. Weitere Optionen mit Angaben zu Konfigurationswerten können ebenfalls eingetragen werden. Diese DHCPREQUEST-Nachricht wird an BootP Relay Agents weitergeleitet. Um sicherzustellen, dass die Relay Agents die Nachricht an dieselben Server weiterleiten, die auch die DHCPDISCOVER-Nachricht erhalten haben, muss die DHCPREQUEST-Nachricht denselben Wert im secs-Feld erhalten und dieselbe Broadcast-Adresse verwenden. Wenn der Client keine DHCPOFFER-Nachricht erhält, läuft ein Timer ab und sendet die DHCPDISCOVER-Nachricht erneut (siehe Abb. 3–12).

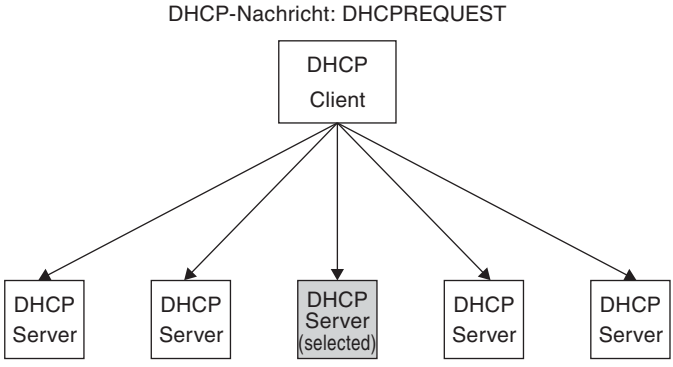

Abb. 3–12 *IP-Adresszuordnung Phase 3*

- *Phase 4*
  Die Server empfangen vom Client den DHCPREQUEST-Broadcast. Diejenigen Server, die vom Client nicht selektiert worden sind (Eintrag *server identifier*), betrachten die DHCPREQUEST-Nachricht als Ablehnung. Der selektierte Server verpflichtet sich nun, dem Client Ressourcen bereitzustellen, und antwortet mit einer DHCPACK-Nachricht mit Angabe der Konfigurationsparameter des anfragenden Clients. Die Kombination aus *chaddr* (*Client Hardware Address*) und einer IP-Adresse bildet eine eindeutige Identifikation innerhalb des Zuordnungszeitraums für jeden DHCP-Nachrichtenverkehr. In das Feld *yiaddr* (*your IP-Address*) der DHCPACK-Nachricht wird die zugeordnete IP-Adresse eingetragen (siehe Abb. 3–13).

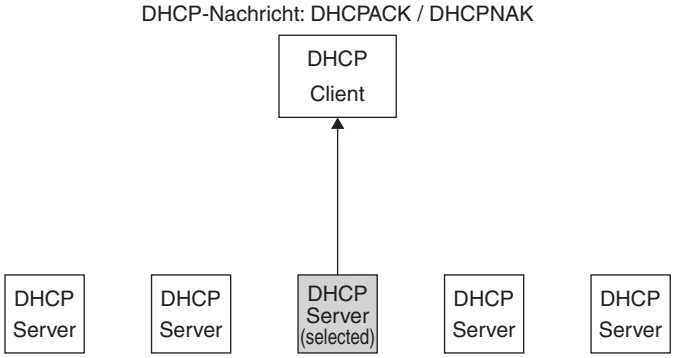

Abb. 3–13 *IP-Adresszuordnung Phase 4*

HINWEIS

Sollte der selektierte Server jedoch nicht in der Lage sein, den DHCPREQUEST zu beantworten (wenn beispielsweise die angeforderte Netzwerkadresse bereits vergeben wurde), wird der Server mit einer DHCPNAK-Nachricht antworten.

- *Phase 5*
  In Phase 5 erhält der Client die DHCPACK-Nachricht samt Konfigurationsparametern, führt einen abschließenden Test der Parameter durch (z.B. ARP für die zugeordnete IP-Adresse) und notiert die Zuordnungsdauer und die spezifizierte Identifikation. Die Konfiguration des Clients ist damit abgeschlossen. Stellt der Client jedoch fehlerhafte Parameter in der DHCPACK-Nachricht fest, übermittelt er dem Server eine DHCPDECLINE-Nachricht und wiederholt den Konfigurationsprozess. Der Client sollte mindestens 10 Sekunden warten, bevor er diesen Prozess startet, um unnötigen Netzwerkverkehr zu vermeiden. Der Client wiederholt den Konfigurationsprozess ebenfalls, wenn er eine DHCPNAK-Nachricht erhält. Sollte er in einem festgelegten Zeitintervall weder DHCPACK- noch DHCPNAK-Nachricht empfangen, so wiederholt er die DHCPREQUEST-Nachricht. Wenn der Client nach zehn Wiederholungen

der DHCPREQUEST-Nachricht weder eine DHCPACK- noch eine DHCP-NAK-Nachricht erhalten hat, erfolgt eine vollständige Initialisierung des Prozesses. Außerdem wird der Anwender über diesen Status informiert.

- Phase 6
  Wenn der Client seine Adresszuordnung beenden will, sendet er dem Server eine DHCPRELEASE-Nachricht. Der Client identifiziert die aufzugebende Zuordnung durch den Eintrag seiner IP-Adresse ins Feld *ciaddr* (*Client IP-Address*) und seiner Hardwareadresse ins Feld *chaddr* (*Client Hardware Address*).

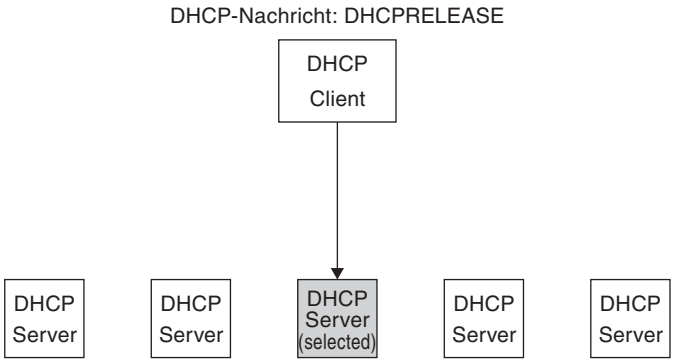

Abb. 3–14 *IP-Adresszuordnung Phase 6*

3.3.3 DHCP im Windows-Netzwerk

Bei DHCP erkundigt sich ein Endgerät (Client) beim Start im Netzwerk nach einer IP-Adresse. Sofern verfügbar, findet das Endgerät einen DHCP-Server, der dann eine IP-Adresse zuweist, mit der sich das Endgerät im Netzwerk bekannt machen kann. Der Server merkt sich die Knotenadresse (MAC-Adresse) des Endgeräts und die zugewiesene IP-Adresse.

Wie in der Praxis in einem Windows-Netzwerk ein DHCP-Server eingerichtet werden kann, ist Inhalt der nachfolgenden Erläuterungen. Dabei orientieren sich diese Angaben an dem Betriebssystem Windows Server 2008 (R2).

HINWEIS

Ein DHCP-Server, der unter Windows Server 2008 eingerichtet werden soll, muss auf jeden Fall Mitglied einer Windows-Domäne sein. Darüber hinaus muss im Netzwerk ein Nameserver (DNS) verfügbar sein; Näheres dazu enthält das Kapitel 5.

Zur erstmaligen Einrichtung eines DHCP-Servers muss im Startmenü innerhalb des Menüs *Verwaltung* der *Server-Manager* aufgerufen werden. In der Oberfläche des entsprechenden Dienstprogramms werden dann die verschiedenen Optionen ange-

zeigt. Dabei ist es wichtig zu wissen, dass es auf einem Windows-System *Features* und *Rollen* gibt, wobei die Funktion des DHCP-Servers zu Letzeren zählt.

Nach dem Anklicken des Eintrags *Rollen* im linken Bereich muss dann rechts der Punkt *Rollen hinzufügen* angewählt werden, so wie in Abbildung 3–15 dargestellt.

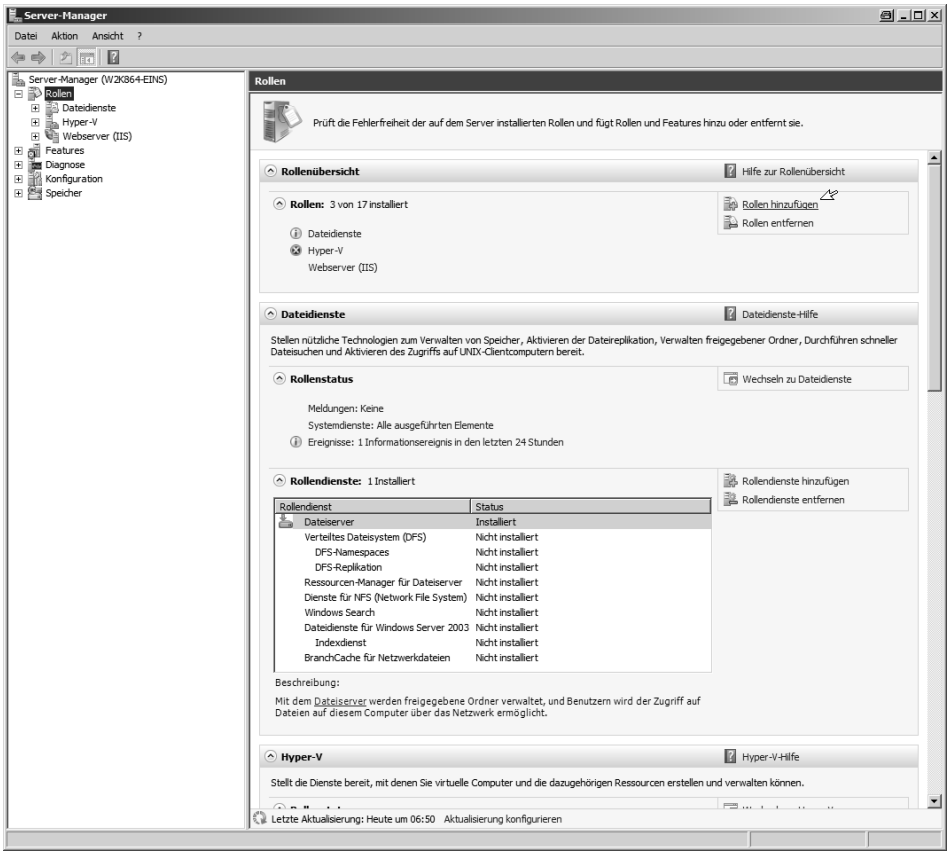

Abb. 3–15 *Hinzufügen einer Rolle im Server-Manager*

Nachfolgend erscheint ein Assistent, dessen erste Seite mit der Schaltfläche *Weiter* bestätigt werden muss. Im nachfolgenden Auswahlfenster muss dann die Rolle *DHCP-Server* angeklickt und dies mit *Weiter* bestätigt werden.

Es erscheint eine Informationsseite, auf der die wesentlichen Grundlagen von DHCP und die Funktionen eines DHCP-Servers nachgelesen werden können.

Im nächsten Fenster des Assistenten muss der Netzwerkcontroller ausgewählt werden, über den die DHCP-Dienste bereitgestellt werden. In der Regel wird hier die lokale Netzwerkeinstellung angezeigt, die dann mit *Weiter* übernommen werden kann.

Im nächsten Konfigurationsfenster muss der Name der übergeordneten Windows-Domäne angegeben werden, der der Server angehört, und es müssen die Namen der verfügbaren Nameserver (DNS-Server) angegeben werden, wobei die Adresse des alternativen DNS-Servers notfalls auch leer gelassen werden kann (sofern es nur einen DNS-Server gibt). In Abbildung 3–16 sind entsprechende Einträge beispielhaft dargestellt.

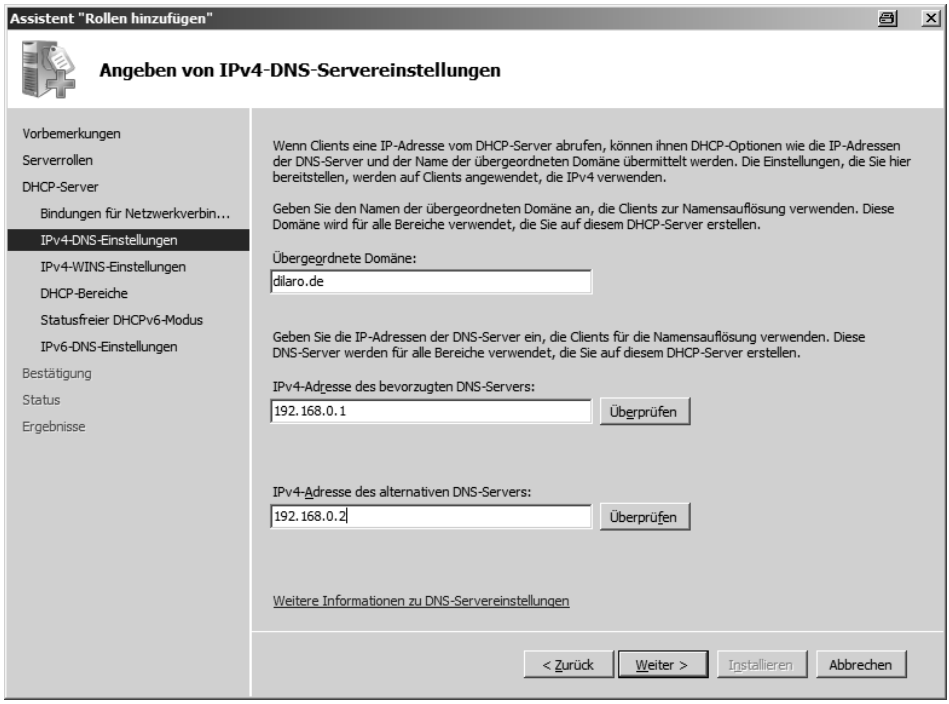

Abb. 3–16 *Notwendige Vorgaben für den DHCP-Server*

Nach Bestätigung der Konfigurationsvorgaben mit *Weiter* können im nächsten Fenster die Einstellungen für den WINS-Dienst (*Windows Internet Naming Service*) eingetragen werden. Dies ist dann interessant und wichtig, wenn es beispielsweise spezielle Anwendungen gibt, die diese proprietäre Form der Namensauflösung der Firma Microsoft benötigen.

Als *DHCP-Bereiche* werden die verfügbaren Adressbereiche definiert, die der Server den Clients für die Adressvergabe zur Verfügung stellt. Dazu muss die Schaltfläche *Hinzufügen* angeklickt und dann ein Name für den Bereich und die Anfangs- und Endadresse angegeben werden. Im unteren Bereich kann die Subnetzmaske und das Standard-Gateway festgelegt werden (siehe Abb. 3–17).

Abb. 3–17 *Festlegung eines DHCP-Bereichs*

Innerhalb eines DHCP-Bereichs stehen Optionen zur Verfügung (Adresspool, Leases, Reservierungen, Bereichsoptionen), mit denen diverse Einstellungen an der Konfiguration vorgenommen werden können. Beispielsweise können im Bereich *Reservierungen* Adressen definiert werden, die nicht dynamisch vergeben werden, sondern für bestimmte Geräte reserviert werden sollen (Server usw.). Nach Bestätigung mit *OK* erscheint der Eintrag in der Aufstellung der verfügbaren DHCP-Bereiche. Es können an dieser Stelle weitere Bereiche definiert werden, um dies anschließend mit *Weiter* zu bestätigen.

HINWEIS

Die Festlegungen der Adressbereiche und die Konfigurationsvorgaben für den DHCP-Server können natürlich auch jederzeit nachträglich geändert werden.

Die Einstellungen auf der nächsten Konfigurationsseite hängen vom Netzwerk ab, in dem sich der DHCP-Server befindet; wurde dort bereits IPv6 aktiviert, muss an dieser Stelle die entsprechende Konfiguration angewählt werden, wobei sich diese wiederum aus der Konfiguration der eingesetzten Router ergibt. Ist keine IPv6-Unterstützung notwendig, muss die zweite Option (*Statusfreien DHCPv6-Modus für diesen Server deaktivieren*) angewählt werden.

HINWEIS

Nähere Informationen zur Version 6 von IP enthält das Kapitel 10.

Abschließend folgt eine Zusammenstellung der Vorgaben für die Einrichtung des DHCP-Servers, so wie in Abbildung 3–18 beispielhaft dargestellt.

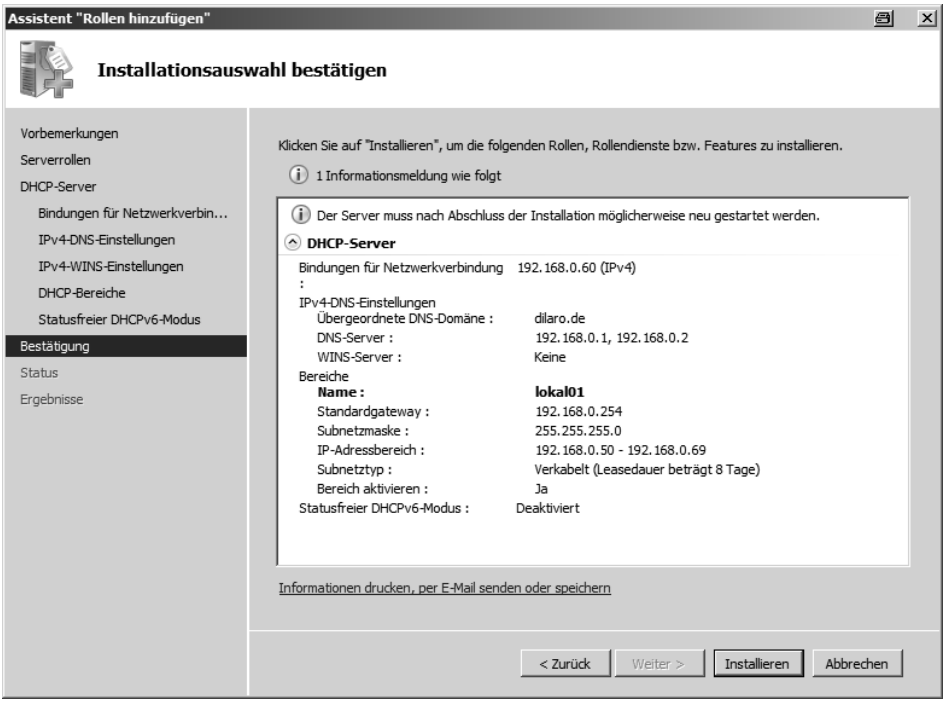

Abb. 3–18 *Zusammenfassung der Konfigurationsvorgaben eines DHCP-Servers*

Mit Einsatz der Schaltfläche *Installieren* erfolgt die abschließende Einrichtung des DHCP-Servers, der anschließend sofort verfügbar ist.

Wurde ein DHCP-Server einmal eingerichtet, kann dieser jederzeit nachträglich über den Server-Manager umkonfiguriert werden. Innerhalb des Server-Managers gibt es dafür einen speziellen Eintrag, der sämtliche Konfigurationsmöglichkeiten zur Verfügung stellt (siehe Abb. 3–19).

HINWEIS

Sobald ein DHCP-Server verfügbar ist, können Clients im entsprechenden Netzwerksegment darauf zugreifen, indem dort der DHCP-Client (in der Regel ein Dienst) und in den Netzwerkeinstellungen die Verwendung von DHCP aktiviert wird.

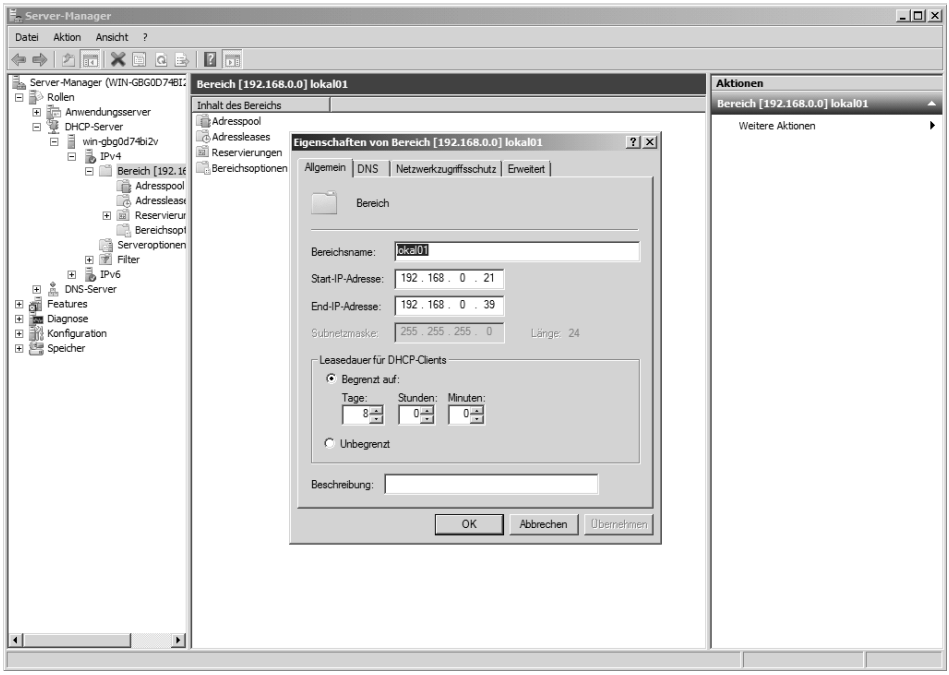

Abb. 3–19 *Nachträgliche Konfiguration eines DHCP-Servers im Server-Manager*