



Unternehmen für Bildung.

IT - Berufe

Software

Modul:

**J – Netzwerk &
Protokolle Teil 1**

Version:

1.0

Januar 2019

Berufsfortbildungswerk GmbH (bfw)

Geschäftsbereich Ost

Berufsbildungsstätte Leipzig
Hohmannstraße 6
04129 Leipzig

Modul: **J – Netzwerke & Protokolle**

Version, Datum: 1.0 Januar 2019

Berufsfortbildungswerk GmbH (bfw)
Geschäftsbereich Ost

Berufsbildungsstätte Leipzig
Hohmannstraße 6 04129 Leipzig

Alle Rechte vorbehalten. Kein Teil des Werkes darf in irgendeiner Form (Druck, Fotokopie oder einem anderen Verfahren) ohne Genehmigung des Herausgebers reproduziert oder unter Verwendung elektronischer Systeme verarbeitet, vervielfältigt oder verbreitet werden.

Die Modulunterlage und alle ihre Bestandteile wurden mit großer Sorgfalt erstellt und geprüft. Dennoch können Fehler nicht vollkommen ausgeschlossen werden. Der Herausgeber und die Autoren können für fehlerhafte Angaben und deren Folgen weder eine juristische Verantwortung noch irgendeine Haftung übernehmen.

Literaturhinweis

Titel: **Computer-Netzwerke – Grundlagen, Funktionsweise, Anwendung**

ISBN: ISBN 978-3-8362-4323-0

Verlag: Rheinwerkverlag

Inhalt

3.1 Physikalische Adresse (MAC-Adresse) Tag 1 - 5	7
3.2 Ethernet-Pakete (Ethernet-Frames).....	8
3.3 Zusammenführung von MAC- und IP-Adresse	9
3.3.1 Address Resolution Protocol (ARP), IPv4	9
3.3.2 Neighbor Discovery Protocol (NDP), IPv6	10
3.4 IP-Adressen	12
3.5 IPv4-Adressen.....	13
3.5.1 Netzwerkklassen im IPv4.....	13
3.5.2 Netz- und Subnetzmaske, Unterteilung von Netzen.....	14
3.5.3 Berechnungen	17
3.5.4 Private Adressen des IPv4	18
3.5.5 Zeroconf – konfigurationsfreie Vernetzung von Rechnern.....	19
3.5.6 Localnet und Localhost	20
3.5.7 Weitere reservierte Adressen	20
3.6 IPv6-Adressen.....	21
3.6.1 Adresstypen des IPv6	22
3.6.2 IPv6-Loopback-Adresse	25
3.6.3 Unspezifizierte Adresse.....	25
3.6.4 IPv4- in IPv6-Adressen und umgekehrt.....	25
3.6.5 Tunnel-Adressen	26
3.6.6 Kryptografisch erzeugte Adressen (CGA).....	27
3.6.7 Lokale Adressen	27
3.6.8 Übersicht der Präfixe von IPv6-Adressen.....	28
3.6.9 Adresswahl und –benutzung.....	28
3.7 Internetprotokoll.....	29
3.7.1 Der IPv4-Header	30
3.7.2 Der IPv6-Header	32
3.8 Prüfungsfragen.....	33
3.8.1 Berechnungen	33
3.8.2 IP-Adressen	33
4.1 MAC-Adressen.....	33
4.1.1 Ermitteln der MAC-Adresse	33
4.1.2 Ändern der MAC-Adresse	35
4.1.3 Manuelles Setzen und Ändern von MAC-Adressen mittels »arp«.....	35
4.1.4 ARP-Spoofing erkennen	35

4.2 IP-Adressen setzen	36
4.2.1 Netzwerkkonfiguration von PCs	37
4.2.2 IP-Adresskonfiguration von weiteren Netzwerkgeräten	43
4.2.3 Zentrale IP-Adressverwaltung mit dem DHCP-Server	45
4.2.4 Zeroconf	50
4.3 Verwendung von Rechnernamen Tag 6 - 10	51
4.3.1 Der Urtyp: Adressauflösung in der »hosts«-Datei	51
4.3.2 Der Domain Name Server (DNS) und seine Konfiguration	52
4.3.3 Einstellungen beim Client	60
4.4 Überprüfung der Erreichbarkeit und Namensauflösung von Hosts	61
4.4.1 Prüfung der Erreichbarkeit und Namensauflösung mit »ping« bzw. »ping6«	61
4.4.2 Werkzeuge für Nameserver-Abfragen (»nslookup«, »host«, »dig«)	63
4.4.3 Mitschnitte von DNS-Abfragen mit Netzwerkdiagnoseprogrammen	64
4.5 Zentrale Netzwerkgeräte auf Sicherungs- und Vermittlungsebene	66
4.5.1 Bridges – Verbinden von Netzwerkteilen	66
4.5.2 Hubs – die Sammelschiene für TP-Netze	66
4.6 Switches – Verbindungsknoten ohne Kollisionen	67
4.6.1 Funktionalität	67
4.6.2 Schleifen – Attentat oder Redundanz?	68
4.6.3 Verbindungen zwischen Switches	70
(Link Aggregation, Port Trunking, Channel Bundling)	70
4.6.4 Virtuelle Netze (VLAN)	71
4.6.5 Switch und Sicherheit	73
4.6.6 Geräteauswahl	75
4.6.7 Anzeigen und Anschlüsse am Switch	76
4.6.8 Konfiguration eines Switchs allgemein	77
4.6.9 Spanning Tree am Switch aktivieren	77
4.6.10 VLAN-Konfiguration von Switches	78
4.6.11 Konfiguration von Rechnern für tagged VLANs	79
4.7 Routing – Netzwerkgrenzen überschreiten	82
4.7.1 Gemeinsame Nutzung einer IP-Adresse mit PAT	85
4.7.2 Festlegen des Standard-Gateways	85
4.7.3 Routing-Tabelle abfragen (»netstat«)	85
4.7.4 Routenverfolgung mit »traceroute«	86
4.7.5 Route manuell hinzufügen (»route add«)	86
4.7.6 Route löschen (»route«)	88

4.8 Multicast-Routing.....	89
4.9 Praxisübungen.....	90
4.9.1 Glasfasern.....	90
4.9.2 TP-Verkabelung.....	90
4.9.3 Switches	90
4.9.4 MAC- und IP-Adressen	90
4.9.5 Namensauflösung	90
4.9.6 Routing.....	90
4.9.7 Sicherheit im lokalen Netz	91
5. Steuer- und Fehlercodes mit ICMP Tag 11 - 20.....	92
und ICMPv6 übertragen.....	92
5.1 ICMP-Pakete (IPv4).....	92
5.2 ICMPv6-Pakete	93
6. Datentransport mit TCP und UDP	96
6.1 Transmission Control Protocol (TCP)	96
6.1.1 Das TCP-Paket	96
6.1.2 TCP: Verbindungsaufbau.....	99
6.1.3 TCP: Transportkontrolle	100
6.1.4 TCP: Verbindungsabbau	101
6.2 User Datagram Protocol (UDP)	102
6.2.1 UDP: Der UDP-Datagram-Header.....	102
6.3 Nutzung von Services mittels Ports und Sockets	103
6.3.1 Sockets und deren Schreibweise	105
6.3.2 Übersicht über die Port-Nummern	105
6.3.3 Ports und Sicherheit.....	108
6.4 Die Firewall.....	110
6.4.1 Integration der Firewall in das Netzwerk.....	111
6.4.2 Regeln definieren	113
6.5 Der Proxyserver.....	116
6.5.1 Lokaler Proxyserver.....	117
6.5.2 Proxyserver als eigenständiger Netzwerkteilnehmer	117
6.5.3 Squid, ein Proxyserver.....	118
6.6 Port and Address Translation (PAT), ...	118
Network Address Translation (NAT)	118
6.7 Praxis Tag 21 - 25	120
6.7.1 Verbindungsaufbau zu einem Dienst mit geänderter Port-Nummer	121

6.7.2 Durchführen von Portscans zum Austesten von Sicherheitsproblemen	121
6.7.3 Schließen von Ports.....	122
6.8 Prüfungsfragen.....	123
6.8.1 TCP-Protokoll.....	123
6.8.2 Ports und Sockets.....	123
6.8.3 Firewall.....	124

3.1 Physikalische Adresse (MAC-Adresse)

Mit der (physikalischen) MAC-Adresse (Media Access Control) können Sie ein am Netzwerk angeschlossenes Gerät weltweit eindeutig identifizieren. Egal, ob Netzwerkkarte, Printserver oder Webkamera, jedes Gerät verfügt über eine unverwechselbare MAC-Adresse. Sie entspricht der Fahrgestellnummer eines Kraftwagens und wird bei der Herstellung des Geräts fest »eingepägt«. Sie unterscheidet sich damit von den logischen Adressen des OSI-Layers 3 (z. B. IP-Adressen). Die MAC-Adresse wird in hexadezimaler Form angegeben und weist eine Länge von 6 Byte auf. Anhand der ersten drei Bytes der MAC-Adresse können Sie den Hersteller identifizieren.

Aufbau von MAC-Adressen

LL:LL:LL:XX:XX:XX

LL: Herstellercode

XX: Identifikationsteil

Eine Trennung der Bytes kann auch durch »-« erfolgen.

Die hier gezeigte Schreibweise ist auch als *kanonische Schreibweise* bekannt (Norm IEEE 802.3). Im Gegensatz dazu gibt es auch die *Bit-reverse-Darstellung*, sie ist in RFC 2469 näher beschrieben.

Herstellercode	Hersteller
00-05-5D	D-Link Systems Inc.
00-09-5B	Netgear Inc.
00-E0-4C	Realtek Semiconductor Corp.
00-E0-4F	Cisco Systems Inc.
00-E0-64	Samsung Electronics

Tabelle 3.1 Auswahl von Herstellercodes in den MAC-Adressen

Sie können unter <http://standards.ieee.org/regauth/oui/index.shtml> den Herstellercode ermitteln oder die komplette aktuelle Liste herunterladen.

Broadcast-MAC-Adresse

Für das Rundsenden (Broadcast) an alle Teilnehmer innerhalb des erreichbaren Netzwerksegments wird die Adresse

ff:ff:ff:ff:ff:ff

verwendet. Sie wird vom Betriebssystem ausgegeben und ist geräteunabhängig.

Die Netzwerkprotokolle verwenden die MAC-Adresse nur innerhalb des gleichen Netzsegments.

Sicherheitshinweis

Die MAC-Adresse wird stets über das jeweilige Betriebssystem bzw. Microcontrollerprogramm weitergereicht. Damit besteht die Möglichkeit, beliebige MAC-Adressen zu verwenden oder die tatsächliche Adresse zu verändern. Für die eindeutig sichere Identifizierung von Geräten ist die MAC-Adresse daher nur sehr bedingt geeignet!

Sie können die MAC-Adresse für die Geräteverwaltung verwenden. Im Zusammenhang mit Datenbank Anwendungen weisen Sie für PCs und Thin-Clients nicht nur die IP-Adressen, sondern auch Boot-Images oder weitere Zugangsmöglichkeiten zu. Allerdings müssen Sie den jeweiligen Gerätedatensatz nach einem Tausch der Netzwerkkarte ändern.

3.2 Ethernet-Pakete (Ethernet-Frames)

Ihre Daten transportieren Sie immer mithilfe der *Ethernet-Pakete*. Darin gehen Ihre in Einzelpakete zerlegten Daten vom Sender zum Empfänger, und das unabhängig vom verwendeten Netzwerkprotokoll.

Im Grunde würde für den Datentransport dieser einfache Paketmechanismus bereits ausreichen. Leider ist die Reichweite auf ein einziges, gemeinsames *Netzwerksegment* beschränkt. Sobald also Komponenten wie Switch, Bridge oder Router zwischen Ihrem Rechner und dem Zielgerät liegen, funktioniert mangels Adressierbarkeit die Kommunikation nicht. Für den segmentübergreifenden (und damit weltweiten) Datenverkehr benötigen Sie die Hilfe der höheren Netzwerkprotokolle. Diese enthalten die notwendigen Mechanismen.

Die *Ethernet-Frames* haben einen relativ einfachen Aufbau. Die Pakete haben eine Standardlänge von 1518 Byte (Tabelle 3.2) bzw. 1522 Byte, wenn Sie um das VLAN-Tag erweitert sind (Tabelle 3.3).

7 Byte	1 Byte	6 Byte	6 Byte	2 Byte	bis 1500 Byte	max. 46 Byte	4 Byte
Präambel	SFD	MAC-Adresse Ziel	MAC-Adresse Quelle	Typ	Nutzdaten	PAD	CRC (FCS)

Tabelle 3.2 Standard-Ethernet-Frame nach IEEE 802.3

7 Byte	1 Byte	6 Byte	6 Byte	4 Byte	2 Byte	bis 1500 Byte	max. 42 Byte	4 Byte
Präambel	SFD	MAC-Adresse Ziel	MAC-Adresse Quelle	VLAN-Tag	Typ	Nutzdaten	PAD	FCS

Tabelle 3.3 Um das VLAN-Tag erweiterter Ethernet-Frame nach IEEE 802.1Q

Das VLAN-Tag kennzeichnet virtuelle Netzwerke (VLANs). Deren Mechanismen und Einsatzmöglichkeiten erkläre ich in Abschnitt 4.6.4, »Virtuelle Netze (VLAN)«. Meist haben Sie es heute mit den um das VLAN-Tag erweiterten Ethernet-Frames zu tun.

Bestandteile des Ethernet-Frames mit VLAN-Tag

- **Präambel:** 7 Byte mit wechselnden Bits (»10101010 ...«), zum Synchronisieren der Endgeräte, noch aus Kompatibilitätsgründen vorhanden
- **SFD:** Start-Frame-Delimiter, Bitfolge 10101011, zum Synchronisieren und Anzeigen des Beginns des Frames, noch aus Kompatibilitätsgründen vorhanden
- **MAC-Adresse Ziel**
- **MAC-Adresse Absender (Quelle)**
- **VLAN-Tag:** Kennzeichnung für virtuelle Netzwerke
- **Typ:** Angabe über das von den Nutzdaten angewendete Netzwerkprotokoll. Einige wichtige Vertreter hiervon sind: 0x800 IPv4, 0x86DD IPv6, 0x0806 Address Resolution Protocol (ARP), 0x0842 Wake on LAN (WOL).
- **Nutzdaten:** maximal 1500 Byte
- **Pad:** dient zum Auffüllen des Ethernet-Frames auf die Mindestgröße von 64 Byte, wenn diese mit den Nutzdaten nicht erreicht wird

3.3 Zusammenführung von MAC- und IP-Adresse

Sie wollen Ihre Datenpakete weltweit transportieren. Dazu benötigen Sie die Hilfe der Netzwerkprotokolle. Sie müssen die MAC-Adresse des Rechners mit seiner IP-Adresse zusammenführen. Hierbei helfen Ihnen die im Folgenden dargestellten Protokolle.

3.3.1 Address Resolution Protocol (ARP), IPv4

Mit dem *Address Resolution Protocol (ARP)* ermitteln Sie in IPv4-Netzen die MAC-Adresse zu einer IP-Adresse. Sie können damit auch einer MAC-Adresse eine IP-Adresse manuell zuweisen.

Normalerweise arbeitet das ARP-Protokoll ohne direkten Benutzereingriff unauffällig im Hintergrund. Daten sollen von A nach B geschickt werden, und Sie geben jeweils die IP-Adresse auf dem OSI-Layer 3 an. Das ARP-Protokoll fragt alle erreichbaren Netzwerkteilnehmer, ob sie über die gesuchte IP-Adresse verfügen. Der Zielrechner antwortet entsprechend und gibt seine MAC-Adresse dem Fragesteller bekannt. Die Kommunikation zwischen den beiden wird nunmehr aufgebaut. Das Beispiel in Abbildung 3.1 zeigt Ihnen den Ablauf detailliert. Der linke Rechner sendet die ARP-Anfrage mit der eigenen IP- und MAC-Adresse (IP(S), MAC(S)), die IP-Adresse des Empfängers (IP(E)) und die Broadcast-Adresse ff:ff:ff:ff:ff:ff. Die erreichbaren Rechner prüfen, ob ihnen die angegebene Adresse gehört. Der angesprochene Zielrechner sendet daraufhin ein Paket mit seiner MAC-Adressangabe zurück (MAC(E)). Die anderen Rechner verwerfen das Paket.

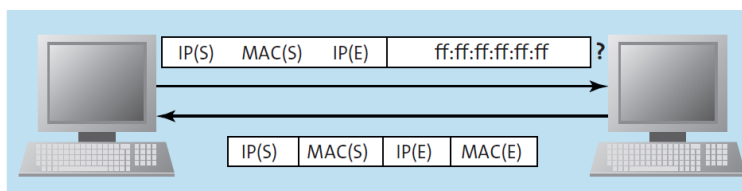


Abbildung 3.1 Schema einer ARP-Abfrage

Protokoll einer ARP-Anfrage

Rechner1 stellt die Anfrage:

```
00:1e:33:0d:e9:f9 > ff:ff:ff:ff:ff:ff, ARP,
length 42: arp who-has 192.168.0.102 tell 192.168.0.103
```

Rechner1 erhält als Antwort von Rechner2:

```
00:11:6b:62:93:2e > 00:1e:33:0d:e9:f9, ARP,
length 60: arp reply 192.168.0.102 is-at 00:11:6b:62:93:2e
```

Nach diesem Austausch steht der weiteren Datenübertragung nichts mehr im Wege.

Beide Rechner speichern normalerweise die erfolgreiche ARP-Anfrage im *ARP-Cache*. Wie lange die Informationen dort verbleiben, ist unterschiedlich. Gebräuchlich sind Zeiten zwischen 5 und 10 Minuten.

Inhalt der ARP-Caches der beteiligten Rechner

Cache von Rechner1:

```
192.168.0.102 ether 00:11:6b:62:93:2e C eth0
```

Cache von Rechner2:

```
? (192.168.0.103) at 00:1e:33:0d:e9:f9 on re0 expires in 1078 seconds
[ethernet]
```

Die verschiedenartige Darstellung der Cache-Inhalte resultiert aus der Verwendung verschiedener Betriebssysteme.

Ist die IP-Adresse unbekannt, können Sie mittels des *Reverse Address Resolution Protocol*s (*RARP*) einen zentralen Rechner kontaktieren und nach der IP-Adresse fragen. Dieses Netzwerkprotokoll ist aber kaum mehr von Bedeutung.

Proxy-ARP und Sicherheit: Im seltenen Fall, dass ein Router zwei Netze mit gleichem IPAdressbereich verbindet, antwortet der Router anstelle des Zielrechners. Listet man den ARP-Cache auf, taucht die IP-Adresse des Routers anstelle der des Zielrechners auf. **Proxy-ARP oder *ARP-Spoofing*:** Wenn Sie den ARP-Cache des abfragenden Rechners auflisten und finden zu ein und derselben MAC-Adresse verschiedene IP-Adressen, befindet sich entweder ein Proxy-ARP-Server im Netz oder es liegt ARP-Spoofing vor. In diesem Fall hat ein Angreifer vor, den Netzwerkverkehr zu belauschen oder zu manipulieren!

Rechner versenden einen ARP-Broadcast mit der eigenen IP-Adresse als Absender und Ziel mit dem Zweck, die neue, eigene MAC-Adresse allen erreichbaren Rechnern des eigenen Netzwerksegments bekannt zu geben (*Gratuitous ARP*). Das ist beim Laden oder Netzwerkstart eines Rechners ganz normal. Die benachrichtigten Rechner ergänzen oder ändern daraufhin die entsprechenden Einträge des ARP-Caches.

Wenn Sie zwei oder mehrere Rechner redundant betreiben, wird damit allen anderen Teilnehmergeräten der Umstieg auf die Reserve, was das Netzwerk angeht, bekannt gemacht.

3.3.2 Neighbor Discovery Protocol (NDP), IPv6

Was Sie hier im Buch über das ARP-Protokoll für IPv4-Adressen gelesen haben, würde grundsätzlich auch mit IPv6-Adressen funktionieren. In der Praxis kommt aber stattdessen das *Neighbor Discovery Protocol (NDP)* zum Einsatz. Die Grundlagen hierfür finden Sie in den RFCs 3122 und 4861 hinterlegt, Updates in den RFCs 5942, 6980, 7048, 7527 und 7559. Auch dieses Protokoll arbeitet unbemerkt vom Benutzer, und normalerweise ist keinerlei Eingriff nötig.

Die Funktionen, Begriffe und Eigenschaften des NDP für IPv6

- ▶ **Neighbor Solicitation:** Anfrage nach der MAC-Adresse
- ▶ **Neighbor Advertisement:** Antwort auf eine Anfrage nach der MAC-Adresse (Neighbor Solicitation)
- ▶ **Knoten:** Netzwerkteilnehmer
- ▶ **Finden** von Routern, die Pakete in weitere Netzwerke weiterreichen
- ▶ **Duplicate IP Address Detection:** Zwei Knoten (Netzwerkteilnehmer) können niemals die gleiche IP-Adresse bekommen.
- ▶ Kombination von **ARP** und **ICMP Router Discovery:** Auffinden von Routern im gleichen Netzwerk

- ▶ **Neighbor Unreachability Detection:** automatische Feststellung nicht erreichbarer Knoten im eigenen Netzwerk
- ▶ **Autokonfiguration** von IPv6-Adressen
- ▶ **Wegfall** der Subnetzmasken
- ▶ neue **Sicherheitsfunktionen**

Bei der Fehlersuche hilft Ihnen Tabelle 3.4 weiter. NDP verwendet mehrere verschiedene ICMPv6-Nachrichten. Allgemeines über ICMP-Nachrichten finden Sie in Kapitel 5, »Steuer- und Fehlercodes mit ICMP und ICMPv6 übertragen«.

Typ	Name	Zweck
133	Router Solicitation	Netzwerkteilnehmer, die nicht als Router fungieren, können von den Routern im gleichen Netzwerk ein Router Advertisement anfordern. Der oder die erreichbaren Router reagieren sofort und übersenden die gewünschte Nachricht.
134	Router Advertisement	Router verschicken diese Nachrichten in festen Intervallen oder nach Anforderungen des Typs 133 sofort. Sie geben damit ihre Anwesenheit im Netz bekannt.
135	Neighbor Solicitation	Rechner überprüfen damit die Erreichbarkeit des Nachbarn oder ermitteln die MAC-Adresse von diesem.
136	Neighbor Advertisement	Es handelt sich um die Antwort auf die Anfrage des Typs 135, der angefragte Rechner ist erreichbar.
137	Redirect Message	Router informieren damit andere Netzwerkteilnehmer, wenn es einen anderen, besseren Router zum Erreichen des gewünschten Zieles gibt.

Tabelle 3.4 Nachrichtentypen des NDP

Betrachten Sie den Ablauf einer Adressanfrage mit Antwort.

Protokoll einer Neighbor Solicitation mit Antwort

Anfrage von Rechner1:

```
16:15:19.726208 IP6 fe80::211:6bff:fe62:932e > fe80::21e:33ff:fe0d:e9f9:
ICMP6, neighbor solicitation, who has fe80::21e:33ff:fe0d:e9f9, length 32
```

Antwort von Rechner2:

```
16:15:19.726249 IP6 fe80::21e:33ff:fe0d:e9f9 > fe80::211:6bff:fe62:932e:
ICMP6, neighbor advertisement, tgt is fe80::21e:33ff:fe0d:e9f9, length 24
```

NDP bringt seine Informationen in verschiedenen Caches unter. Sie finden daher die Informationen besser geordnet.

Übersicht über die Caches des NDPs

- ▶ **Destination Cache:** enthält Adressen, an die erfolgreich Daten gesendet wurden, und die Angabe für den nächsten Hop, den die Pakete benutzen sollen
- ▶ **Default Router List:** enthält für jeden Netzwerkanschluss die erreichbaren Router. Die Einträge enthalten Verfallsvorgaben.
- ▶ **Prefix List:** beinhaltet Präfixe, die im gleichen Netzwerk gültig sind, mit Verfallsvorgaben. Ausnahme: Angaben des eigenen Rechners (Link-local)
- ▶ **Neighbor Cache:** Liste mit Netzwerkteilnehmern, mit denen innerhalb einer festgelegten Zeitspanne erfolgreich kommuniziert wurde. Es werden die Unicast-IP-Adresse, die MAC-Adresse, der Zustand, das »Alter« des Eintrags und der Netzwerkanschluss festgehalten.

Aus dem *Neighbor Cache* können Sie den Zustandswert eines Eintrags auslesen. Sie können damit nicht nur protokollbedingte Netzwerkprobleme eingrenzen. Die Bedeutung dieser Angaben finden Sie in Tabelle 3.5.

Zustand	Bedeutung
Delay	Ablauf der <i>Reachable Time</i> , Versand eines Datenpakets innerhalb der <i>Delay First Probe Time</i> . Kommt innerhalb dieser Zeit keine Quittung, ändert sich der Zustand auf <i>Probe</i> , und eine Neighbor Solicitation wird abgesetzt.
Probe	Zustand während des Wartens auf eine positive Quittung der Neighbor Solicitation. Der Cache-Eintrag verfällt, wenn die vorkonfigurierte Anzahl der Versuche ohne positives Ergebnis durchgeführt wurde.
Stale	Es ist unbekannt, ob der angegebene Nachbarrechner erreichbar ist. Der Zeitpunkt der letzten positiven Quittung liegt länger als die <i>Reachable Time</i> zurück. Der Zustand bleibt bis zum Versenden des nächsten Datenpakets an den hier eingetragenen Rechner bestehen.
Incomplete	der Zustand nach dem Absenden der Neighbor Solicitation und vor dem Eintreffen des Neighbor Advertisement
Reachable	Der Rechner ist erreichbar, da innerhalb der <i>Reachable Time</i> ein Neighbor Advertisement von diesem erhalten wurde.

Tabelle 3.5 Einträge für den Zustandswert im Neighbor Cache des NDP

3.4 IP-Adressen

In der OSI-Schicht 3 begegnen Ihnen *logische Adressen*. Damit haben Sie die Möglichkeit, Verbindungen mit Partnern außerhalb des eigenen Netzwerksegments aufzunehmen. Die logischen Adressen dürfen Sie durchaus mit dem amtlichen Kfz-Kennzeichen vergleichen; die MAC-Adresse entspräche der Fahrgestellnummer.

Die logischen Adressen kennzeichnen auch die IP-Pakete. Sie tauchen hier als Quell und Zieladressen auf, unabhängig von der Hardware-Adresse der Ethernet-Frames.

Die Adressierung nach dem IP-Protokoll hat sich im Laufe der Jahre gegen herstellerspezifische Lösungen (Novell, DEC, Microsoft usw.) durchgesetzt. Es war und ist für die Anwender einfacher, einen offenen, allgemeingültigen Standard zu verwenden, der zudem von Geräten aller Größenordnungen beherrscht wird. Sie werden – besonders hier in diesem Kapitel – mit IP-Adressen nach dem IP-Protokoll Version 4, aber auch Version 6 konfrontiert. Das hat ganz einfach damit zu tun, dass sich der Vorrat an freien Adressen des IPv4 langsam dem Ende zuneigt (4.294.967.296 Adressen). Beim IPv6 treffen Sie nicht nur auf einen schier unerschöpflichen Adressvorrat (ca. 34×1038 Adressen),

sondern auch auf einige interessante Neuerungen, die Sie nur hier finden.

Weltweit vergibt die *Internet Assigned Numbers Authority (IANA)* Adressblöcke an die regionalen Vergabestellen *Regional Internet Registry (RIR)* (Tabelle 3.6). Diese wiederum versorgen die *Local Internet Registry (LIR)* mit freien Adressen. Die nächsterreichbare LIR ist Ihr Internet-Service-Provider. Von ihm erhalten Sie entweder Ihre feste oder temporär gültige IP-Adresse.

Region	RIR
Afrika	AfriNIC (African Network Information Centre)
Asien, Pazifik	APNIC (Asia Pacific Network Information Centre)
Europa, Naher Osten, Zentralasien	RIPE NCC (Réseaux IP Européens Network Coordination Centre)
Lateinamerika, Karibik	LACNIC (Regional Latin-American and Caribbean IP Address Registry)
Nordamerika	ARIN (American Registry for Internet Numbers)

Tabelle 3.6 Die »Regional Internet Registries« (RIR)

Die *Internet-Service-Provider (ISPs)* sind in der Regel Mitglied in der RIR. In einem *privaten Netzwerk* vergeben Sie selbst die IP-Adressen.

Einzigkeit von IP-Adressen

Jede IP-Adresse darf innerhalb eines Netzwerks nur ein einziges Mal vorkommen.

3.5 IPv4-Adressen

Die IPv4-Adressen bestehen aus 32 Bit, die zu vier Blöcken (Oktetts) gruppiert werden (4 Byte). Die Werte werden als Dezimalzahlen geschrieben, mit einem zulässigen Bereich von 0 bis 255.

Aufbau von IPv4-Adressen

NNN.NNN.NNN.NNN

Eine IPv4-Adresse besitzt einen Netzwerk- und einen Teilnehmeranteil (Host). Welches der Bytes allerdings zu welchem Teil gehört, hängt wiederum von der Netzwerkklass ab. Sollen Ihre Rechner alle miteinander kommunizieren, müssen deren IP-Adressen denselben Netzwerkteil aufweisen.

Die IPv4-Adressen sind seit 1981 in RFC **791** definiert. In den RFCs 2474, 3168, 3260, 4301 und 6040 finden Sie einige Weiterentwicklungen und Verweise auf weitere beteiligte RFCs.

Mit IPv4 werden Sie noch lange arbeiten. Vor allem nicht öffentliche Bestandsnetze werden es sicher bis auf Weiteres anwenden. Viele Netzwerkgeräte (z. B. Printserver) können nicht auf die neue IP-Version aktualisiert werden. Daher sollten Sie also die theoretischen Grundlagen dieser Version kennen.

3.5.1 Netzwerkklassen im IPv4

Anhand der IP-Adresse erkennen Sie normalerweise nicht mehr, wie weit sich der Netzwerkanteil erstreckt. Hierbei hilft Ihnen die (Sub-)Netzmaske.

Zusammen mit der (Sub-)Netzmaske, die im nächsten Abschnitt erklärt wird, können Sie Netze mit einer verschiedenen hohen Anzahl von Teilnehmern definieren. Allerdings gilt dies nur für private Netze. Wenn Sie Netze betreuen, die »draußen« im Internet existieren, müssen Sie unbedingt die Vorgaben Ihres Providers (= LIR) einhalten, weil es sonst zu großen Störungen kommt. Konkret bedeutet dies, dass Sie Ihre Einteilungen nur »unterhalb«, also »rechts« vom Netzwerkanteil, vornehmen dürfen. Früher konnten Sie einer IP-Adresse bereits ansehen, welcher Netzklasse sie angehörte (Tabelle 3.7). Das erste Byte zeigte dies an. Jedoch wurde diese Einteilung seit Längerem aufgehoben, um der Adressknappheit etwas abzuhelpfen.

Klasse	Bereich	1. Byte binär	Nutzbare Adressen	Netzmaske
A	0.0.0.0 – 127.255.255.255	00000000	16777214	255.0.0.0
B	128.0.0.0 – 191.255.255.255	10000000	65534	255.255.0.0
C	192.0.0.0 – 223.255.255.255	11000000	254	255.255.255.0

Tabelle 3.7 Einteilung von Netzwerkklassen

Beachten Sie bitte einige Anmerkungen zur Einteilung in Netzwerkklassen (Tabelle 3.7):

- Die Zahl der mathematisch möglichen Adressen wird stets um zwei vermindert. Sie benötigen diese für die Netz- und Broadcast-Adresse (z. B. 192.168.0.0 und 192.168.0.255).
- Während Sie die Bereichsangaben als mehr oder weniger historisch betrachten müssen, ist die Einteilung hinsichtlich der nutzbaren Adressen und die Angabe der Subnetzmaske in der Tabelle weiterhin gültig.
- Die in der Tabelle nicht aufgeführte Klasse D dient zu Multicast-Zwecken (bei Datenstreams, z. B. Video). Hier klinken sich Rechner passiv ein. Die Klasse E dient zu Test- und Forschungszwecken. Sie werden in der Praxis in der Regel mit beiden Klassen kaum in Berührung kommen.

Zur Verdeutlichung der Aufteilung einer IP-Adresse betrachten Sie folgende Angaben:

Anteile von Netz (N)- und Host-Anteilen (H) nach Netzwerkklassen

Klasse A: NNN.HHH.HHH.HHH

Klasse B: NNN.NNN.HHH.HHH

Klasse C: NNN.NNN.NNN.HHH

3.5.2 Netz- und Subnetzmaske, Unterteilung von Netzen

Eine Netz- oder Subnetzmaske besteht wie die IPv4-Adresse aus vier Bytes. Mit ihr unterteilen Sie IP-Adressen in den Netzwerk- und den Host-Anteil.

Die Unterteilung eines Netzes in Subnetze bringt Ihnen verschiedene Vorteile:

- Mit dadurch verkleinerten Broadcast-Domänen verringern Sie die Netzlast durch Rundsendungen (Broadcasts).
- Sie können organisatorische Gegebenheiten durch die aufgeteilten Adressräume abbilden.

Sie können (in der Regel innerhalb privater Netze) durch die Netzmaske aber auch z. B. zwei Klasse-C-Netze nach alter Denk- und Sprachweise miteinander verbinden.

Netzmaske und Subnetzmaske

- ▶ Umfasst die Maske alle möglichen Adressen eines Netzes, handelt es sich um eine **Netzmaske** (siehe [Tabelle 3.8](#)).
- ▶ Unterteilt sie jedoch den Adressraum, spricht man von der **Subnetzmaske**.
- ▶ Technisch bestehen zwischen der Netzmaske und der Subnetzmaske keine Unterschiede.
- ▶ Der Netzanteil wird stets mit 1, die Host-Anteile mit 0 belegt.
- ▶ Netz- und Subnetzmasken bestehen somit stets aus **führend gesetzten Bits**. Es treten keine »O-Lücken« auf.
- ▶ Weitere Normen: RFCs 950, 4632, 6918

Im Rahmen des *Classless Inter-Domain Routings (CIDR)* wurde auch eine neue Schreibweise für die Netzmaske eingeführt. Anstelle der byteweisen Angabe (z. B. 255.255.0.0) werden hier die führenden gesetzten Bits zusammengezählt und per Schrägstrich an die IP-Adresse angehängt.

Die Angabe 192. 168. 0. 36/24 beinhaltet die Netzmaske 255. 255. 255. 0. Binär wird die Netzmaske in diesem Fall so dargestellt:

11111111. 11111111. 11111111. 00000000

Zählen Sie die gesetzten Bits (oder rechnen Sie einfach 3 Oktets mal 8 Bits), erhalten Sie hier die 24.

Für Ihre tägliche Arbeit sind die Angaben in Tabelle 3.8 und Tabelle 3.9 sicher nützlich. Sie finden in ihnen eine Gegenüberstellung der binären, klassischen und der CIDRSchreibweise. In Tabelle 3.9 ist die Anzahl der Adressen bzw. Hosts je Netzmaske angegeben.

(Sub-)Netzmaske		
Binär	Dezimal	CIDR-Notation
11111111.00000000.00000000.00000000	255.0.0.0	/8
11111111.10000000.00000000.00000000	255.128.0.0	/9
11111111.11000000.00000000.00000000	255.192.0.0	/10
11111111.11100000.00000000.00000000	255.224.0.0	/11
11111111.11110000.00000000.00000000	255.240.0.0	/12
11111111.11111000.00000000.00000000	255.248.0.0	/13
11111111.11111100.00000000.00000000	255.252.0.0	/14
11111111.11111110.00000000.00000000	255.254.0.0	/15
11111111.11111111.00000000.00000000	255.255.0.0	/16
11111111.11111111.10000000.00000000	255.255.128.0	/17
11111111.11111111.11000000.00000000	255.255.192.0	/18
11111111.11111111.11100000.00000000	255.255.224.0	/19
11111111.11111111.11110000.00000000	255.255.240.0	/20
11111111.11111111.11111000.00000000	255.255.248.0	/21
11111111.11111111.11111100.00000000	255.255.252.0	/22
11111111.11111111.11111110.00000000	255.255.254.0	/23
11111111.11111111.11111111.00000000	255.255.255.0	/24
11111111.11111111.11111111.10000000	255.255.255.128	/25
11111111.11111111.11111111.11000000	255.255.255.192	/26
11111111.11111111.11111111.11100000	255.255.255.224	/27
11111111.11111111.11111111.11110000	255.255.255.240	/28
11111111.11111111.11111111.11111000	255.255.255.248	/29
11111111.11111111.11111111.11111100	255.255.255.252	/30
11111111.11111111.11111111.11111110	255.255.255.254	/31
11111111.11111111.11111111.11111111	255.255.255.255	/32

Tabelle 3.8 Subnetzmasken

Die binäre Darstellung benötigen Sie bei der Berechnung von Netzadressen und Subnetzmasken.

CIDR-Notation	Adressen	Nutzbare Host-Adressen	CIDR-Notation	Adressen	Nutzbare Host-Adressen
/8	16.777.216	16.777.214	/21	8 × 256	2046
/9	128 × 65.536	8.388.606	/22	4 × 256	1022
/10	64 × 65.536	4.194.302	/23	2 × 256	510
/11	32 × 65.536	2.097.150	/24	256	254
/12	16 × 65.536	1.048.574	/25	128 × 1	126
/13	8 × 65.536	524.286	/26	64 × 1	62
/14	4 × 65.536	262.142	/27	32 × 1	30
/15	2 × 65.536	131.070	/28	16 × 1	14
/16	65.536	65.534	/29	8 × 1	6
/17	128 × 256	32.766	/30	4 × 1	2
/18	64 × 256	16.382	/31	2 × 1	0
/19	32 × 256	8.190	/32	1 × 1	1
/20	16 × 256	4.094	–	–	–

Tabelle 3.9 Subnetzmaske und Adressräume

3.5.3 Berechnungen

Sie können einen (zugewiesenen) Adressbereich in Teilsegmente unterteilen, allerdings in dem in Tabelle 3.9 gezeigten Mengenraster. Die Unterteilung zeige ich Ihnen anhand eines Beispiels.

Ihr Netz soll in zwei Teile segmentiert werden. In einem werden sich künftig 19 Teilnehmer aufhalten, im anderen 80. Sie verwenden das Netzwerk 192. 168. 1. 0. Sie wollen unter anderem wissen, in welches der Teilnetze das Gerät mit der IP-Adresse 192. 168. 1. 25 gehört. Außerdem interessiert Sie natürlich der IP-Adressbereich, der vergeben werden kann. Bei den Berechnungen verwenden Sie die *logische Addition (AND)*.

Sie hat folgende Regeln (Wahrheitstabelle):

Logische Addition (AND)

$0 + 0 = 0$
 $0 + 1 = 0$
 $1 + 0 = 0$
 $1 + 1 = 1$

Wenn bei zwei Binärzahlen also jeweils die 1 aufeinandertrifft, bekommen Sie wiederum eine 1 als Ergebnis.

Sehen Sie zunächst den manuellen Lösungsweg zur Frage, wohin 192. 168. 1. 25 gehört:

Manuelle Berechnungen

- Entnehmen Sie aus der Tabelle die passende Netzmaske, hier: /27, ausgeschrieben 255.255.255.224.
- Schreiben Sie die IP-Adresse und Netzmaske binär untereinander, und führen Sie die logische Addition durch:

11000000.10101000.00000001.00011001 192.168.1.25
 11111111.11111111.11111111.11100000 255.255.255.224
 11000000.10101000.00000001.00000000 192.168.1.0

Die Netzadresse lautet also 192.168.1.0, in der 192.168.1.25 Mitglied ist.

Sie können natürlich auch ein Rechenprogramm wie *ipcalc* für die IP-Adressen und Netzmasken benutzen. Die Ausgaben einer Berechnung für die beiden Teilnetze sehen Sie in Abbildung 3.2. Dabei soll der Rechner mit der IP-Adresse 192.168.1.25 zum ersten, kleineren Netz gehören.

Um nun herauszubekommen, ab welcher Adresse das zweite Teilnetz beginnt, führen Sie die Berechnung für die Netzadresse 192.168.1.0 zweimal aus: einmal mit der zutreffenden Netzmaske 255.255.255.224 und für das zweite Teilnetz mit 255.255.255.128.

Sie können dann die höchste vorkommende Adresse ablesen (Broadcast). Deren Wert erhöhen Sie um 1 und nehmen das Ergebnis als Startwert für die dritte Berechnung. Zusammen mit der richtigen Netzmaske (/25) erhalten Sie alle notwendigen Angaben für das zweite Teilnetz mit den 80 Teilnehmern.

Im Ergebnis erhalten Sie jeweils die Netzadresse, die niedrigste und höchste für Netzwerkteilnehmer verwendbare Adresse (HostMin und HostMax), deren Anzahl (Hosts/Net) und die Broadcast-Adresse.

```

harald@ZE4:~$ ipcalc 192.168.1.0/27
Address: 192.168.1.0      11000000.10101000.00000001.000 00000
Netmask: 255.255.255.224 = 27 11111111.11111111.11111111.111 00000
Wildcard: 0.0.0.31      00000000.00000000.00000000.000 11111
=>
Network: 192.168.1.0/27  11000000.10101000.00000001.000 00000
HostMin: 192.168.1.1    11000000.10101000.00000001.000 00001
HostMax: 192.168.1.30   11000000.10101000.00000001.000 11110
Broadcast: 192.168.1.31 11000000.10101000.00000001.000 11111
Hosts/Net: 30           Class C, Private Internet

harald@ZE4:~$ # 2. Berechnung, um die Startadresse für das 2. Teilnetz zu erhalten
harald@ZE4:~$ ipcalc 192.168.1.0/25
Address: 192.168.1.0      11000000.10101000.00000001.0 0000000
Netmask: 255.255.255.128 = 25 11111111.11111111.11111111.1 0000000
Wildcard: 0.0.0.127      00000000.00000000.00000000.0 1111111
=>
Network: 192.168.1.0/25  11000000.10101000.00000001.0 0000000
HostMin: 192.168.1.1    11000000.10101000.00000001.0 0000001
HostMax: 192.168.1.126  11000000.10101000.00000001.0 1111110
Broadcast: 192.168.1.127 11000000.10101000.00000001.0 1111111
Hosts/Net: 126           Class C, Private Internet

harald@ZE4:~$ # 3. Berechnung für das zweite Teilnetz
harald@ZE4:~$ ipcalc 192.168.1.128/25
Address: 192.168.1.128    11000000.10101000.00000001.1 0000000
Netmask: 255.255.255.128 = 25 11111111.11111111.11111111.1 0000000
Wildcard: 0.0.0.127      00000000.00000000.00000000.0 1111111
=>
Network: 192.168.1.128/25 11000000.10101000.00000001.1 0000000
HostMin: 192.168.1.129  11000000.10101000.00000001.1 0000001
HostMax: 192.168.1.254  11000000.10101000.00000001.1 1111110
Broadcast: 192.168.1.255 11000000.10101000.00000001.1 1111111
Hosts/Net: 126           Class C, Private Internet

```

Abbildung 3.2 Berechnungen für die beiden Teilnetze

3.5.4 Private Adressen des IPv4

Für den Aufbau Ihrer privaten, d. h. nicht mit dem Internet direkt verbundenen Netze benutzen Sie die durch RFC 1918 und das Update-RFC 6761 definierten Adressbereiche (Tabelle 3.10). Diese Adressen werden auch nicht in das öffentliche Internet weitergeleitet.

Adressbereich	Anzahl der IP-Adressen
10.0.0.0 – 10.255.255.255	ein Netz mit 16.777.216 Adressen, Netzmaske 255.0.0.0 10.0.0.0/8
172.16.0.0 – 172.31.255.255	16 Netze mit 65.536 Adressen, Netzmaske 255.255.0.0 172.16.0.0/16–172.31.0.0/16

Tabelle 3.10 Adressbereiche privater Netzwerke

Adressbereich	Anzahl der IP-Adressen
192.168.0.0 – 192.168.255.255	256 Netze mit 256 Adressen, Netzmaske 255.255.255.0 192.168.0.0/24–192.168.255.0/24

Tabelle 3.10 Adressbereiche privater Netzwerke (Forts.)

3.5.5 Zeroconf – konfigurationsfreie Vernetzung von Rechnern

Neben den drei klassischen IPv4-Privatnetzen stehen Ihnen auch noch die Adressen von 169.254.1.0 bis 169.254.254.255 zur Verfügung. Auch sie werden nicht in das Internet durchgeroutet. In diesem reservierten Bereich, genannt *Zeroconf*, können Sie Rechner ohne weitere Konfiguration vernetzen. Sie müssen weder manuell noch per DHCP eine IP-Adresse zuweisen. Die Nutzung von DHCP widerspricht sogar dem Wesen von Zeroconf und darf deshalb nicht angewendet werden. Der Adressbereich ist für das *Automatic Private IP Addressing (APIPA)* nach RFC 3927 reserviert, das auch als Zeroconf oder *Auto-IP* bekannt ist. Die Netzmaske lautet 255.255.0.0, die ersten und letzten 256 Adressen dürfen nicht verwendet werden. Somit verbleiben Ihnen immerhin 65.024 Adressen zur Nutzung.

Die automatische Adresskonfiguration greift dabei auch auf die Sicherungsschicht (Layer 2) des OSI-Schichtenmodells zu. Wenn Sie Ihren Rechner einschalten, berechnet er zunächst aus seiner MAC-Adresse zusammen mit einem Zufallsgenerator seine IP-Adresse. Weil eine IP-Adresse in einem Netz nur einmal vorhanden sein darf, prüft Ihr Rechner, ob die Adresse nicht schon von einem anderen Gerät benutzt wird. Er bildet dazu ein ARP-Paket (ARP-Probe), in dem er die Absenderadresse mit 0.0.0.0 und als Empfängeradresse die eigene angibt. Ihr Rechner sendet das Paket insgesamt dreimal alle 1 bis 2 Sekunden aus. Erhält Ihr Rechner innerhalb eines Zeitraums von 2 Sekunden nach dem Absetzen der letzten ARP-Probe ein ARP-Paket, bei dem die errechnete IP-Adresse mit der Absenderadresse übereinstimmt, muss er nochmals eine neue Adresse berechnen. Befinden sich zum gleichen Zeitpunkt besonders viele Rechner am Netz, die ihre Adressen berechnen, müssen Sie mit solchen Adresskonflikten rechnen. Damit das Netzwerk nicht mit zu vielen ARP-Paketen überfrachtet wird, werden die sich gerade konfigurierenden Rechner nach zehn Fehlversuchen nur noch einen Versuch je Minute starten. Für Sie und die anderen Benutzer können also durchaus fühlbare Wartezeiten auftreten.

Gehört Ihrem Rechner »seine« Adresse, macht er diese allen anderen per doppeltem ARP-Announcement im Abstand von 2 Sekunden bekannt. Hierbei verwendet er als Absender- und Empfänger-IP diese eben berechnete Adresse. Im Zeroconf-Netz herrscht hinsichtlich der IP-Adresse trotzdem eine trügerische Ruhe für Ihren Rechner. Er muss jetzt ständig auf Adresskonflikte achten, die durch andere Rechner verursacht werden. Diesen Fall erkennt er, wenn ARP-Pakete von anderen Teilnehmern eintreffen, die »seine« Adresse als Absenderadresse benutzen.

Ihr Rechner reagiert darauf auf verschiedene Weise. Hat er offene TCP-Verbindungen und noch keine kollidierenden ARP-Pakete empfangen, also nur ein ARP-Probe, sendet er ein klarstellendes ARP-Announcement. In allen anderen Fällen wird er sich eine neue Adresse berechnen und damit eine »Schlacht um die IP-Adresse« mit viel Netzlast durch Endlosschleifen vermeiden.

Zeroconf bietet eine eigene Namensauflösung und Diensterkennung an, was in Abschnitt 4.2.4, »Zeroconf«, gezeigt wird.

Zeroconf finden Sie bei allen gängigen Betriebssystemen. Für Apple-Rechner ist es unter *bonjour* bekannt, Linux und Unix-Derivate verwenden hauptsächlich den *avahi-Daemon*. Microsoft nennt das Verfahren *Automatic Private IP Addressing (APIPA)*.

Fakten zu Zeroconf

- ▶ automatische Adresszuweisung ohne DHCP-Server
- ▶ Namensauflösung ohne DNS-Server
- ▶ automatische Erkennung von Netzwerkdiensten
- ▶ Sicherheit: Die automatische Konfiguration greift tief in das jeweilige Betriebssystem ein. Vom Einsatz in sicherheitsrelevanten Umgebungen wird deshalb abgeraten.
- ▶ Norm: RFC 3927

3.5.6 Localnet und Localhost

Moderne Betriebssysteme benötigen ein »internes Netzwerk«, um verschiedene Aufgaben zu erledigen. Im Bereich Unix/Linux greifen unter anderem die Druckdienste und das X11- bzw. Xorg-Grafiksystem darauf zurück. Für diese *Loopback-Adressen* wurde der Adressbereich von 127.0.0.0 bis 127.255.255.255 reserviert. Für jede Netzwerkkarte Ihres Rechners wird diese (zusätzliche) Adresse mehr oder weniger von selbst vergeben. Datenpakete an eine Adresse aus diesem Bereich werden nicht in das Netzwerk geleitet, sondern wieder zurück an den Rechner.

Sie können für viele kritische Anwendungen die Sicherheit erheblich erhöhen, wenn diese ausschließlich auf dieser *Loopback-Adresse* »lauschen«. Natürlich ist das nur sinnvoll, wenn diese Anwendung ausschließlich für diesen einzigen Rechner ihre Dienste anbieten soll, z. B. ein eigener, rechnerlokaler Nameserver oder eine Datenbank. Angriffe auf diesen Rechner über das lokale Netzwerk sind damit ausgeschlossen.

Auch für den Test von Serveranwendungen greifen Entwickler gerne auf den localhost zurück. Sie vermeiden Netzlast und natürlich auch das Ausnutzen von Sicherheitslücken in den noch unfertigen Anwendungen durch Dritte.

Auszug der Ausgabe von »ifconfig« für die superprivate Adresse

```
lo
Link encap:Lokale Schleife
inet Adresse:127.0.0.1 Maske:255.0.0.0
inet6-Adresse: ::1/128 Gültigkeitsbereich:Maschine
UP LOOPBACK RUNNING MTU:16436 Metrik:1
RX packets:433140 errors:0 dropped:0 overruns:0 frame:0
TX packets:433140 errors:0 dropped:0 overruns:0 carrier:0
Kollisionen:0 Sendewarteschlangenlänge:0
RX bytes:50267351 (47.9 MiB) TX bytes:50267351 (47.9 MiB)
```

Auf diesem Linux-Rechner wird das Loopback-Gerät als `lo` bezeichnet. Vergleichen Sie die Angaben zu den gesendeten (TX) und empfangenen (RX) Daten. Sie zeigen stets die gleichen Werte. Die RFCs 2606, 6598, 6761 und 6890 beschreiben den Einsatz dieses besonderen Adressbereichs.

3.5.7 Weitere reservierte Adressen

Sie finden in Tabelle 3.11 weitere reservierte Adressbereiche (RFC 5735). Die Anwendungen sind sehr unterschiedlich. So finden Sie IP-Bereiche, die Sie für Anleitungen und Dokumentationen verwenden können, ohne damit Betriebs- und Sicherheitsgeheimnisse auszuplaudern.

Adressbereich	Netzmaske	Zweck	RFC
0.0.0.0– 0.255.255.255	255.0.0.0	aktuelles Netz (nur als Quelladresse zulässig)	3232
192.0.0.0– 192.0.0.255	255.255.255.0	noch reserviert, aber Ver- gabe vorgesehen	
192.0.2.0– 192.0.2.255	255.255.255.0	Test-Net-1, Dokumenta- tion und Beispielcode	5737
192.88.99.0– 192.88.99.255	255.255.255.0	Weiterleitungspräfix für 6to4-Anycast	7526

Tabelle 3.11 Weitere reservierte IPv4-Adressbereiche

Adressbereich	Netzmaske	Zweck	RFC
198.18.0.0– 198.19.255.255	255.254.0.0	Benchmark-Tests	2544, 6201, 6815
198.51.100.0– 198.51.100.255	255.255.255.0	Test-Net-2, Dokumenta- tion und Beispielcode	5737
203.0.113.0– 203.0.113.255	255.255.255.0	Test-Net-3, Dokumenta- tion und Beispielcode	5737
224.0.0.0– 239.255.255.255	weitere Unterteilungen	Multicasts	5771
240.0.0.0– 255.255.255.255	weitere Unterteilungen	reserviert	1112 Sect. 4, 2236, 3376
255.255.255.255	255.255.255.255	Broadcast	919 und 922

Tabelle 3.11 Weitere reservierte IPv4-Adressbereiche (Forts.)

3.6 IPv6-Adressen

Sie werden mehr und mehr mit dem Nachfolger des IPv4 zu tun haben. IPv6 wird IPv4 ablösen (müssen). Die maximal 4.294.967.296 Adressen des IPv4 sind bald komplett vergeben. Eine Adressknappheit bedeutet aber eine Stagnation beim weiteren Ausbau des Internets. Schon lange wurde deshalb am Nachfolgeprotokoll gearbeitet. Die Netzgemeinde profitiert bei IPv6 nicht nur vom erweiterten Adressraum. Das IPv6 besitzt auch neue Leistungsmerkmale, die unter IPv4 nicht oder nur mit externen Erweiterungen verfügbar waren.

Kenndaten des IPv6

- ▶ Anzahl der Adressen: 2^{128}
- ▶ zustandslose automatische Konfiguration von Adressen
- ▶ integrierte Verschlüsselung von IP-Paketen (*IPsec*)
- ▶ Verwirklichung des Ende-zu-Ende-Prinzips
(Wegfall von Verfahren wie *Network Address Translation [NAT]*)
- ▶ IP-Adressen für mobile Geräte
- ▶ Vereinfachung von Techniken wie *Quality of Service* und *Multicast*
- ▶ Wichtige Normen: RFC 2460 (Updates in RFC 5095, 5722, 5871, 6437, 6564, 6935, 6946, 7045, 7112) und RFC 4291 (Updates in RFC 5952, 6052, 7136, 7346, 7371)

IPv6-Adressen bieten Ihnen einen anderen Anblick. Durch ihre Länge von 128 Bit und ihre hexadezimale Schreibweise sind sie schwerer lesbar als die gewohnten IPv4-Adressen. Rechner haben allerdings weniger Schwierigkeiten mit der Umwandlung von Hexadezimalzahlen in Binärzahlen als mit der von Dezimalzahlen in Binärzahlen.

Eine IPv6-Adresse wird wegen ihrer Länge anders dargestellt. Sie wird in acht durch »:«

getrennte Blöcke zu 16 Bit unterteilt, und die Werte werden in hexadezimaler Schreibweise angegeben.

Schreibweisen von IPv6-Adressen

- ▶ **Grundform:** fe80:0000:0000:0000:0223:54ff:fe5b:869d
- ▶ **Nullen:** Sie dürfen führende Nullen innerhalb eines Blocks auslassen.
- ▶ **Blöcke mit dem Wert 0:** Sie dürfen aufeinanderfolgende Blöcke, deren Wert 0 beträgt, auslassen und durch einen Doppelpunkt ersetzen. Sie dürfen das innerhalb einer Adresse allerdings nur einmal durchführen.
- ▶ **Beispiel nach dem Entfernen führender Nullen und von zusammenhängenden Blöcken mit 0:** fe80::223:54ff:fe5b:869d
- ▶ **URL:** IPv6-Adressen werden in eckige Klammern eingeschlossen:
http://[fe80::223:54ff:fe5b:869d]/
- ▶ **Port-Nummern:** Sie werden außerhalb der Klammerung angefügt:
http://[fe80::223:54ff:fe5b:869d]:8080/

(Sub-)Netzmasken in der klassischen Form sind bei IPv6 nicht gebräuchlich, vielmehr benutzen Sie hier die CIDR-Notation. Die Größe eines zu vergebenden Netzwerks muss einer Zweierpotenz entsprechen; ein einzelner Host trägt die /128. Abbildung 3.3 zeigt Ihnen die Gliederung einer IPv6-Adresse.

Byte	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
=Bits (Länge)	8	16	24	32	40	48	56	64	72	80	88	96	104	112	120	128
Adresse (Beispiel)	fe	80	00	00	00	00	00	00	02	23	54	ff	fe	5b	86	9d

Abbildung 3.3 Gliederung einer IPv6-Adresse

Sie geben ein Netz in der Form fe80:0000:0000:0000:0223/80 an. Die Netzadresse lautet damit fe80:0000:0000:0000:0223:0000:0000:0000, die Adressen für die Hosts gehen von fe80:0000:0000:0000:0223:0000:0000:0001 bis fe80:0000:0000:0000:0223:ffff:ffff:ffff, was 281.474.976.710.655 Adressen ergibt.

Ihr Internet-Provider bezieht normalerweise die ersten 48 Bits des Netzes von seiner *Regional Internet Registry (RIR)*. Diesen Bereich teilt er weiter in Teilnetze auf. Die Teilnetz-ID ist normalerweise 16 Bit lang. Sie selbst bekommen ein Teilnetz /64. Nun kennen Sie bereits die ersten beiden Teile Ihrer Adressen, das Standort-Präfix und die Teilnetz-ID. Der dritte Teil besteht aus dem 64 Bit langen *Interface Identifier*. Dabei wird entweder die MAC-Adresse der Netzwerkkarte zur Berechnung mit herangezogen, oder Sie vergeben diesen Teil der IPv6-Adresse selbst (Abbildung 3.4, RFC 4291).

XX:XX:XX:	XX:	XX:XX:XX:XX
Standortpräfix	Teilnetz-ID	Schnittstellen-ID

Abbildung 3.4 Unterteilung von IPv6-Adressen

Bestandteile von IPv6-Adressen

- ▶ **Präfix:** kennzeichnet den Adressraum, wird in CIDR-Schreibweise angegeben
- ▶ **Subnet-ID:** Organisationen unterteilen damit die ihnen zugewiesenen Bereiche. Sie erhalten ein /48-Präfix und verwenden 16 Bit für die Kennzeichnung ihrer Subnetze.
- ▶ **Interface-ID:** kennzeichnet die Schnittstelle (Schnittstellen-ID)

3.6.1 Adresstypen des IPv6

Sie treffen beim IPv6 auf drei Arten von Adressen. Die Ihnen vom IPv4 her bekannten Broadcast-Adressen sind hier ungebräuchlich und wurden teilweise durch die *Multicast-Adressen* ersetzt.

Grundregeln zur IPv6-Adressvergabe

- ▶ Für jede Netzwerkschnittstelle (Interface) müssen Sie mindestens eine Unicast-Adresse vergeben.
- ▶ Sie können für jede Netzwerkschnittstelle mehrere IPv6-Adressen zuweisen.
- ▶ Sie können für jede Netzwerkschnittstelle gleichzeitig Unicast-, Multicast- und Anycast-Adressen verwenden.
- ▶ Sie können eine Unicast-Adresse mehreren Netzwerkschnittstellen eines Rechners zuweisen, um darüber eine Lastverteilung zu ermöglichen.
- ▶ Sie können einen Knoten über jede der Unicast-Adressen ansprechen.

Die drei Adresstypen *Unicast*, *Multicast* und *Anycast* erfüllen alle bestimmte Zwecke und haben ihre Besonderheiten, auf die Sie achten müssen.

Unicast-Adressen

- ▶ Sie dienen zur eindeutigen Identifizierung eines Interfaces eines Knotens (Netzwerkteilnehmers).
- ▶ **Globale Unicast-Adressen** sind weltweit gültig und werden über das Internet geroutet. Ihr Präfix lautet 2000::/3.
- ▶ **Link-local Unicast-Adressen** gelten nur im lokalen Netz und werden nicht in das Internet weitergeroutet. Ihr Präfix lautet FE80::/10.
- ▶ Jede Netzwerkschnittstelle benötigt eine eigene *Link-local* Adresse, um im privaten Netzwerk den Rechner (Knoten) gegenüber anderen Knoten zu identifizieren. Dies ist für das Funktionieren der automatischen Adresskonfiguration und des *Neighbor Discovery Protocol (NDP)* notwendig.

Wenn Sie eine Unicast-Adresse mehreren Netzwerkschnittstellen zuordnen, haben Sie eine *Anycast-Adresse* geschaffen.

Anycast-Adressen

- ▶ Sie verfügen über das gleiche Präfix wie Unicast-Adressen.
- ▶ Sie werden aus dem Unicast-Adressbereich entnommen.
- ▶ Anycast-Adressen dienen zur Lastverteilung und zur Bildung von Redundanzen.
- ▶ Bei einer Router-Gruppe mit gemeinsamer Anycast-Adresse werden Pakete an den nächsten erreichbaren Router gesendet.
- ▶ Der Absender hat keine Möglichkeit, die Empfangsschnittstelle auszuwählen. Das Zielgerät wird durch das Routing-Protokoll bestimmt.
- ▶ Beim Interface Identifier der Adresse sind alle Bits auf 0 gesetzt.
- ▶ Reservierte Subnetz-Anycast-Adressen verwenden eine Anycast-ID mit 7 Bit Länge. Diese vermindert die Bitzahl beim Interface Identifier entsprechend. Der Interface-Identifier-Anteil beim EUI-64-Format ist damit nur 57 Bit lang.
- ▶ Norm: RFC 2526

Die Anycast-IDs von 00-7D und 7F sind reserviert, 7E ist dem *Mobile IPv6 Home Agent Anycast* vorbehalten.

Multicast-Adressen benötigen Sie, um gleichzeitig viele Rechner anzusprechen, z. B. beim Internetradio, NTP-Server (Zeitserver).

Multicast-Adressen

- ▶ Mit der Multicast-Adresse erreichen Sie eine Gruppe von Netzwerkschnittstellen.
- ▶ Ein Paket an eine Multicast-Adresse erreicht alle Netzwerkschnittstellen mit dieser Adresse und wird von diesen verarbeitet.
- ▶ Eine Netzwerkschnittstelle kann mehrere Multicast-Adressen haben.
- ▶ Multicast-Adressen erkennen Sie am Präfix FF00::/8.
- ▶ Multicast-Adressen enthalten einen Group Identifier anstelle des Interface Identifiers (Tabelle 3.12).

8 Bit	4 Bit	4 Bit	8 Bit	8 Bit	64 Bit	32 Bit
FF	Flags ORPT	Scope	reserviert	plen	IID	Group Identifier

Tabelle 3.12 Zusammensetzung der Multicast-Adresse nach RFC 4489

Die Werte für die Flags (Tabelle 3.13) und den Scope (Tabelle 3.14) zeigen Ihnen weitere Eigenschaften der Multicast-Adresse auf.

Flag	Einträge
0	reserviert, muss mit 0 belegt werden
R	Rendezvous-Point nach RFC 3956 und 7371
P	dynamisch zugewiesene Präfix-Information gemäß RFC 3306 (Updates in RFC 3956, 4489 und 7371)
T	0 = fest von IANA zugewiesen, 1 = temporäre Multicast-Adresse

Tabelle 3.13 Flags in Multicast-Adressen

Im Scope-Feld finden Sie diejenigen Angaben, die die Reichweite der Multicast-Adresse beschränken.

Wert(e)	Zuweisung
0, 3, F	reserviert
6, 7, 9, A, B, C, D	nicht zugewiesen
1	Interface-local Scope

Tabelle 3.14 Einträge im Scope-Feld

Wert(e)	Zuweisung
2	Link-local Scope
4	Admin-local Scope
5	Site-local Scope
8	Organisation-local Scope
E	Global Scope

Tabelle 3.14 Einträge im Scope-Feld (Forts.)

Die nachfolgenden acht Bits sind reserviert, Sie müssen sie mit 0 belegen. Den Bereich plen müssen Sie mit FF belegen (nach RFC 4489).

Das IID-Feld ersetzt das 64-Bit-Präfix gemäß RFC 3306 und den Updates durch die RFCs

3956, 4489, 7371. Sie halten damit die einzelnen Knoten auseinander. Der *Group Identifier* kennzeichnet eine Multicast-Anwendung. Ferner stellt er sicher, dass diese nur einmal auf dem Host vorhanden ist.

Beispiele für bekannte Multicast-Adressen

FF01:0:0:0:0:0:0:1 All-Nodes-Adresse, Interface-local, Scope

FF01:0:0:0:0:0:0:2 All-Router-Adresse, Interface-local, Scope

FF02:0:0:0:0:0:0:1 All-Nodes-Adresse, Link-local, Scope

FF02:0:0:0:0:0:0:2 All-Router-Adresse, Link-local, Scope

3.6.2 IPv6-Loopback-Adresse

Die Ihnen schon vom IPv4 bekannte Loopback-Adresse wird auch im IPv6 verwendet. Sie lautet 0:0:0:0:0:0:0:1, in Kurzform ::1. Auch hier müssen Sie beachten, dass Sie diese Adresse niemals Netzwerkschnittstellen zuweisen dürfen und können.

Loopback-Adresse bei einem Linux-Rechner mit IPv4 und IPv6

```
lo
Link encap:Lokale Schleife
inet Adresse:127.0.0.1 Maske:255.0.0.0
inet6-Adresse: ::1/128 Gültigkeitsbereich:Maschine
.....
```

Beachten Sie die CIDR-Schreibweise (/128), mit der ein einzelner Host gekennzeichnet wird. Manche Systeme, wie FreeBSD, vergeben zusätzlich noch eine Link-local Unicast-Adresse.

Loopback-Adresse bei einem FreeBSD-System mit IPv4 und IPv6

```
lo0: flags=8049<UP,LOOPBACK,RUNNING,MULTICAST> metric 0 mtu 16384
options=3<RXCSUM,TXCSUM>|
inet 127.0.0.1 netmask 0xff000000
inet6 ::1 prefixlen 128
inet6 fe80::1 %lo0 prefixlen 64 scopeid 0x5
.....
```

3.6.3 Unspezifizierte Adresse

Diese Adresse verwendet Ihr Rechner unter anderem während des Ladevorgangs als Absenderadresse. Empfängt er in dieser Phase eine Anfrage zur Adresskonfiguration von einem anderen Knoten, zeigt er damit an, dass er (noch) nicht über eine gültige IP-Adresse verfügt.

Die *All-Zero-Adresse* sollten Sie nie in Zieladressen verwenden. Die Adresse wird mit 0:0:0:0:0:0:0:0 ausgeschrieben oder als :: abgekürzt.

3.6.4 IPv4- in IPv6-Adressen und umgekehrt

Der Wechsel von IPv4 nach IPv6 geschieht allmählich, sodass Sie mit beiden Protokollfamilien parallel arbeiten werden.

Sie werden beim Stöbern in älterer Fachliteratur auf die *IPv4-kompatible IPv6-Adresse* stoßen. Diese speziellen Unicast-Adressen transportierten IPv6-Pakete über ein IPv4-

Netz. Die letzten 32 Bits enthielten die IPv4-Adresse. Dieser Adresstyp wird nicht mehr verwendet und auch nicht mehr unterstützt (RFC 4291, abgelöst durch RFC 6052).

Mit der Verwendung einer *IPv4-mapped IPv6-Adresse* stellen Sie die Adresse eines IPv4-Hosts als IPv6-Adresse dar. Nach RFC 6052 wird empfohlen, den IPv4-Adressteil an den Schluss zu hängen (/96). Damit können Sie den IPv4-Teil vor allem auch in der gewohnten Schreibweise verwenden. Von rechts nach links gesehen (Tabelle 3.15), füllen Sie nach dem IPv4-Teil die nächsten 16 Bits mit FFFF auf. Damit wird eine Länge von 48 Bit erreicht. Die weiteren Stellen werden mit 0 Bits aufgefüllt. Sie erhalten damit ein Präfix 0:0:0:0:0:FFF::/96.

80 Bit	16 Bit	32 Bit
0:0:0:0:0	FFFF	IPv4-Adresse

Tabelle 3.15 IPv4-mapped IPv6-Adresse

Schreibweisen von IPv4-mapped IPv6-Adressen

IPv4-Adresse: 192.168.1.200

IPv6-Adresse: 0:0:0:0:0:FFF::C0A8:01C8

Kurzschreibweisen: ::FFFF:C0A8:01C8, ::FFFF:192.168.1.200

3.6.5 Tunnel-Adressen

Während der Übergangsphase reisen Ihre Datenpakete durch IPv4- und IPv6-Netze. Dabei müssen sich die Adressen anpassen. Ihnen begegnen dabei verschiedene Tunnel-Adressen.

Sie benötigen *6to4-Adressen*, wenn Sie Pakete Ihrer IPv6-Knoten über ein IPv4-Netzwerk zu anderen IPv6-Knoten leiten wollen und dabei nicht auf einen statisch konfigurierten Tunnel zurückgreifen können. Hier wird die IPv4-Adresse nicht am Schluss angehängt, sondern in den führenden Stellen angegeben. Soll die Reise über das öffentliche Internet gehen, dürfen Sie dabei keine privaten IP-Adressen verwenden.

6to4-Adressen tragen das Präfix 2002. Die eingebettete IPv4-Adresse wird hexadezimal dargestellt (Tabelle 3.16).

16 Bit	32 Bit	16 Bit	64 Bit
2002	IPv4-Adresse	Teilnetz-ID	Interface-ID
Präfix	48 Bit		

Tabelle 3.16 Aufbau von 6to4-Adressen

Beispiel einer 6to4-Adresse

IPv4-Adresse: 85.88.3.146

IPv6-Adresse: 2002:5558:0392:0001:0:0:0:8

Sie haben einen Rechner, der mit seiner privaten IPv4-Adresse über einen Router mit dem Internet verbunden ist. Der Router führt die übliche *Network Address Translation (NAT)* durch. Ihr Rechner soll sich in ein entferntes IPv6-Netz einfügen. Um diese Verbindung zu realisieren, müssen Sie den *Teredo-Mechanismus* benutzen. Dieser tunnelt das IPv6 in UDP über Teredo-Relais. Der Teredo-Dienst besteht aus diesen Relais, Clients und Servern.

Teredo stammt aus dem Hause Microsoft. In der Unix-Welt ist der Dienst unter dem Begriff *Miredo* bekannt. Der Dienst umgeht die üblicherweise vorhandene IPv4-Firewall, sodass Sie die Anwendung als Sicherheitsproblem betrachten müssen.

Die Teredo-Adressen (Tabelle 3.17) besitzen das Präfix 2001:0000:/32. Die Pakete werden über den UDP-Port 3544 geleitet. (Erklärungen zu Ports finden Sie in Abschnitt 6.3, »Nutzung von Services mittels Ports und Sockets«.) Die genauen Definitionen finden Sie in RFC 4380 (Updates in den RFCs 5991 und 6081) und Informationen zur Sicherheit von Teredo in RFC 5991.

32 Bit	32 Bit	16 Bit	16 Bit	32 Bit
2001:0000	IPv4-Adresse	Flags	Port	IPv4-Adresse
Präfix	Server			Client

Tabelle 3.17 Aufbau von Teredo-Adressen

Wenn Sie den Wunsch haben, über ein IPv4-Netzwerk zwei IPv6-Hosts ohne Zuhilfenahme von Routern kommunizieren zu lassen, hilft Ihnen dabei das *Intra-Site Automatic Tunnel Addressing Protocol (ISATAP)*.

In RFC 5214 ist festgeschrieben, dass die Adressen im modifizierten EUI-64-Format (nach RFC 4291) gehalten sind. Updates hierzu finden Sie in den RFCs 5952, 6052, 7136, 7346 und 7371. Damit ergibt sich ein Aufbau nach Tabelle 3.18.

64 Bit	32 Bit	32 Bit
Präfix	private IPv4-Adresse 0000:5EFE	IPv4-Adresse
	öffentliche IPv4-Adresse 0200:5EFE	

Tabelle 3.18 Aufbau einer ISATAP-Adresse

Die ersten 64 Bits einer ISATAP-Adresse entsprechen jenen einer gewöhnlichen Unicast-Adresse. Von den folgenden 32 Bits dienen 24 Bits dem *IEEE Organizationally Unique Identifier (OUI)*. Davon sehen Sie in den ersten 16 Bits wiederum, ob es sich um eine private IPv4-Adresse (0000) oder um eine öffentliche (0200) handelt. Mit der Kennung FE zeigt die Adresse an, dass sie eine IPV4-Adresse enthält.

3.6.6 Kryptografisch erzeugte Adressen (CGA)

Durch die Anwendung von kryptografisch erzeugten Adressen (Secure Neighbor Discovery [SEND]) können Sie die Sicherheit bei der Neighbor Discovery (ND) erhöhen. Sie können damit sicherstellen, dass der Absender von NDP-Paketen auch der »rechtmäßige« Inhaber der IP-Adresse ist.

Die Interface-ID wird hierbei durch zusätzliche Parameter und Hash-Funktionen berechnet. In den RFCs 3972, 4581 und 4982 finden Sie Erläuterungen zu den eingesetzten Verfahren. Insbesondere verhüten Sie durch den Einsatz dieser so berechneten Adressen Spoofing-Attacken gegen die Neighbor- und Router-Solicitation und das Router-Advertisement. Auch gegen DOS-Attacken der *Duplicate Address Detection (DAD)* und Massenantworten sind Sie damit geschützt.

3.6.7 Lokale Adressen

Auch im IPv6 treffen Sie auf lokale Adressen. Diese werden keinesfalls in das Internet weitergeleitet, sondern sie dienen zum Datentransport in Ihrem privaten Netzwerk.

Die *Site-local Unicast-Adressen* sind nach RFC 3879 obsolet. Stattdessen können Sie die *Unique-local Unicast-Adressen*, wie das RFC 4193 sie beschreibt, frei innerhalb Ihres lokalen Netzes verwenden. Sie erkennen diese Adressen am Präfix FD00::/8 bzw. FC00/8. Ihren Aufbau finden Sie in Tabelle 3.19 beschrieben.

Bereich	8 Bit	40 Bit	16 Bit	64 Bit
privat administriert	FD00	Global ID	Subnet-ID	Interface-ID
reserviert für Registrierung	FC00 Präfix			

Tabelle 3.19 Der Aufbau von Unique-local Unicast-Adressen

Das Präfix FD00 wenden Sie bei privater, nicht öffentlicher Administration des lokalen Netzes an. FC00 ist für die in der Zukunft einmal mögliche Registrierung eigener privater Adressbereiche vorgesehen.

3.6.8 Übersicht der Präfixe von IPv6-Adressen

Zusammenfassend finden Sie in Tabelle 3.20 eine Übersicht über die wichtigsten Präfixe im IPv6.

Präfix	Zugewiesen für
FF00/8	Multicast-Adressen
FD00/8	Unique-local Unicast-Adressen, privat verwaltet
FC00/8	Unique-local Unicast-Adressen, reserviert für eine später mögliche Registrierung
FE80 /10	Link-local Unicast-Adressen
2000::/3	Global Unicast-Adressen
2001::/32	Teredo-Adressen
2001:db8::/32	nur für Dokumentationszwecke; werden nicht im Internet weitergeleitet
2002:::/16	6to4
0:0:0:0:0:FFFF::/96	IPv4-mapped IPv6-Adresse
0:0:0:0:0:0:0:0::	All-Zero-Adresse
0:0:0:0:0:0:0:1::1/128	Localhost, Loopback-Adresse

Tabelle 3.20 IPv6-Adresspräfixe

3.6.9 Adresswahl und –benutzung

Bei den alten IPv4-Adressen reicht eine einzige für die Netzwerkkarte aus, damit Sie mit Ihrem Rechner im Netz »mitmischen« können. Unter IPv6 hingegen benutzt Ihr Rechner (teilweise für jede Netzwerkschnittstelle) die Link-local Adresse, die eventuell zugewiesenen Uni- und Anycast-Adressen und die Loopback-Adresse. Ferner kennt er die All-Nodes-Multicast-Adresse und die Solicited-Node-Multicast-Adressen. Fungiert er zusätzlich als Router, gehören die Subnet-Router-Anycast-Adresse(n), die Anycast-Adressen und die All-Router-Multicast-Adressen dazu. Welche Adresse Ihr Rechner in welcher Situation benutzt, wird in RFC 6724 definiert. RFC 5220 hierzu ist informeller Natur.

Der Wegfall von *Site-local Unicast-Adressen* (nach RFC 3879) und die dafür neu hinzugekommenen

Unique-local Unicast-Adressen (RFC 4193) bewirken einige Änderungen des Wie und Wann der Anwendung der jeweiligen Adressen.

Als Netzwerkadministrator werden Sie damit in der Regel nicht behelligt, wohl aber als Programmentwickler. Halten Sie sich über die entsprechenden RFCs über www.ietf.org unbedingt auf dem Laufenden!

Einige wichtige Regeln zur Adressbenutzung

- Ein kleinerer Scope-Wert hat Vorrang: Damit werden lokale Adressen vor anderen bevorzugt. Pakete mit Ziel im eigenen, lokalen Netz werden mit diesen lokal gültigen Adressen transportiert.
- Benutzung des möglichst gleichen Scope-Werts oder Link-Typs
- Mobile IP: Home-Adressen gehen vor Care-of-Adressen.

3.7 Internetprotokoll

Das Internetprotokoll (IP) stellt für Sie nur die grundlegenden Transportmechanismen bereit, die für die Sendung von Datagrammen über Netzwerkgrenzen hinweg benötigt werden. Das Internetprotokoll arbeitet verbindungslos. Es wird, anders als bei verbindungsorientierten

Verfahren, keine Verbindung im technischen Sinne aufgebaut, gesichert und abgebaut. Vielmehr sendet Ihr Rechner die Datagramme auf gut Glück in das Netz, in der Hoffnung, diese mögen ihr Ziel erreichen. Damit haben Sie auch schon eine besondere Stärke des Internetprotokolls kennengelernt. Sie und Ihr Rechner müssen sich nicht um Verbindungswege und dergleichen kümmern. Und Sie werden auch kein »Besetztsymbol« erhalten. Wenn viele Rechner gleichzeitig Daten für ein Ziel senden, geht es einfach langsamer (oder, wenn es einfach zu viel wird, gar nicht mehr).

Es ist durchaus normal, dass die Datagramme nicht in der Reihenfolge ihrer Aussendung am Ziel eintreffen bzw. nicht oder unvollständig ankommen. Das IP-Protokoll verfügt für solche Situationen über keinerlei Mechanismen. Diese sind vielmehr eine OSI-Schicht darüber angesiedelt (im TCP-Protokoll, siehe Kapitel 6, »Datentransport mit TCP und UDP«).

Ihre IP-Pakete reisen mittels der Ethernet-Frames durch das Netz. Die IP-Header stehen vor den TCP- bzw. UDP-Headern, die die Nutzdaten selbst aufnehmen. Details über die Ethernet-Frames finden Sie in Abschnitt 3.2, »Ethernet-Pakete (Ethernet-Frames)«.

Anhand des folgenden Gedankenmodells sehen Sie die Zusammenhänge klarer: Eine Fabrik versendet einen Traktor. Für den Versand wird er in eine Kiste (TCP- oder UDP- »Verpackung«) verpackt. Diese Umverpackung ist aber keine geeignete Transporthülle, also wird die Kiste in einen Container gestellt. Der Container enthält ein Dokument mit der Absender- und Zielangabe (IP-Header). Der Container wird auf einen Eisenbahnwaggon verladen. Der Waggon hat unabhängig von den Angaben auf dem Container die Angabe eines Abgang- und Zielbahnhofs (Ethernet-Frame, Quell- und Ziel-MAC-Adresse). Der Waggon gelangt zum Zielbahnhof, einem Güterverkehrszentrum (Router). Der Container wird vom Waggon abgeladen und anhand des Frachtdokuments auf das nächste Verkehrsmittel geladen, das die Zielrichtung ansteuert. Auf diesem Weg wird der Container sicher noch öfter zwischen den verschiedenen Verkehrsträgern (Ethernet-Frames) umgeladen, bis er schließlich beim Empfänger landet. Dort wird er wieder vom Waggon genommen (Herauslösung aus dem Ethernet-Frame). Der Container wird geöffnet und die Kiste herausgeholt (Herausnahme aus der Transportverpackung, IP-Header). Die Kiste (TCP- oder UDP-Paket) wird geöffnet und schließlich der Traktor (Nutzlast) von ihr befreit.

Dieses Beispiel ist sehr ausführlich, Sie erkennen darin aber den kompletten Grundzug des Datentransports über Netzwerke. Oft wird auf den notwendigen Zusammenhang von Ethernet-Frame und IP-Paket nicht eingegangen (Abbildung 3.5). Wenn Sie nun betrachten, was sich damit alles an »Overhead« zusammenballt, verstehen Sie, was es mit dem Begriff *Netzlast* auf sich hat. Selbst für ein paar Byte brauchen Sie diesen ganzen Aufwand.

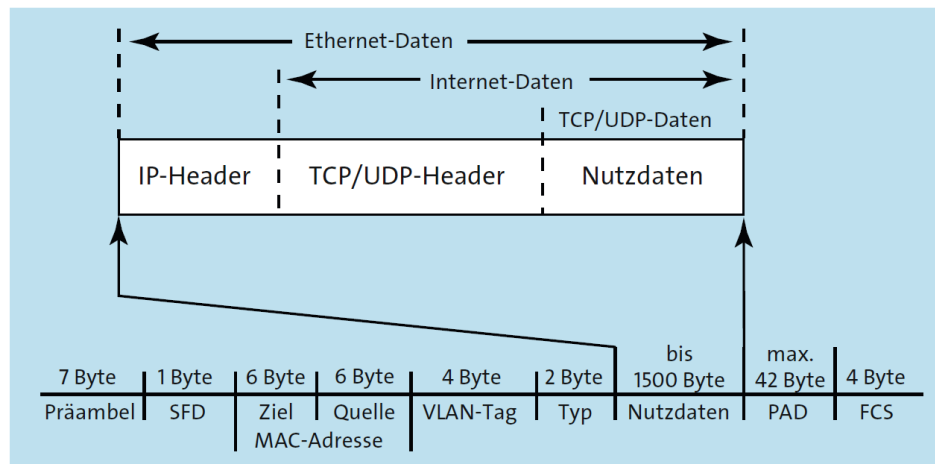


Abbildung 3.5 IP-Paket als Nutzlast des Ethernet-Frames

3.7.1 Der IPv4-Header

Im Header (Abbildung 3.6) finden Sie die Informationen für den Transport von Datagrammen. Je Zeile beträgt die Datenmenge 32 Bit. Die Bedeutung der einzelnen Felder des Headers finden Sie in Tabelle 3.21. Weitere Informationen finden Sie in den RFCs 3168 (Updates 4301, 6040) und 3260.

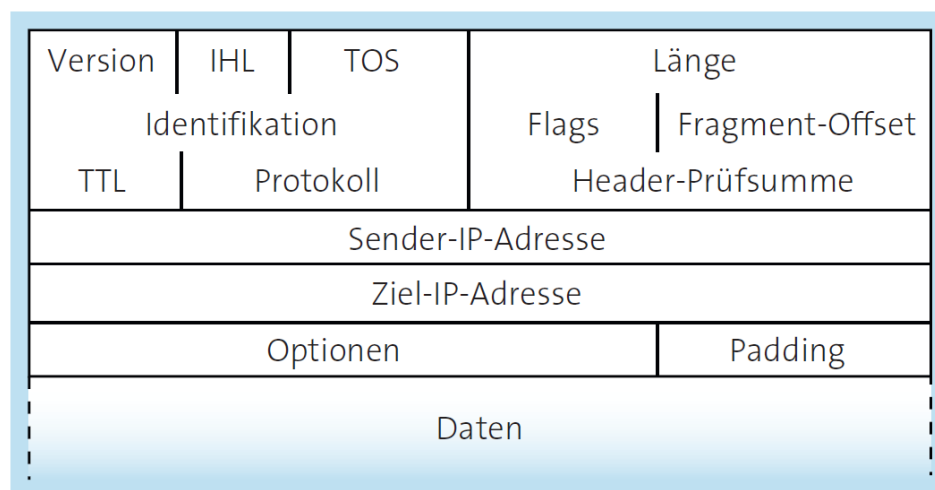


Abbildung 3.6 IPv4-Header

Feld	Länge (Bits)	Inhalt
Version	4	IP-Protokollversion (IPv4)
IHL	4	<i>Internet Header Length</i> , Länge des IP-Headers
TOS	8	<i>Type of Service</i> , Angabe zu Priorität und Eigenschaften des Pakets. Dieses Feld wird heute auch für die Angabe des QoS (Quality of Service) verwendet (RFC 3168, 4301 und 6040), wobei nur die ersten 6 Bits benutzt werden: 0 für <i>Best Effort</i> , 40 für <i>Expedited Flow</i> (VoIP-Datenstrom) und 46 für <i>Expedited Forwarding</i> (VoIP-Datenstrom).
Länge	16	Total Length, maximal 64 kByte
Identifikation	16	laufende Nummerierung der Pakete, dient zum Bilden der richtigen Reihenfolge beim Empfänger
Flags	3	Bit 0: 0 (fest) Bit 1: 0: Fragmentierung erlaubt 1: Fragmentierung verboten Bit 2: 0: letztes Fragment, 1: weitere Fragmente folgen Diese Anweisung betrifft Router. Ist die Fragmentierung nicht erlaubt und das Paket größer als die <i>Maximum Transport Unit</i> (MTU), verfällt das Paket. Im IPv4 beträgt die Standardgröße für die Nutzlast 1500 Byte.

Tabelle 3.21 Inhalt des IPv4-Headers

Feld	Länge (Bits)	Inhalt
Fragment-Offset	13	Positionsangabe für Fragmente
TTL	8	<i>Time to live</i> : Lebensdauer eines Pakets in Sekunden. Der Standardwert beträgt 64. Bei jedem Router, den das Paket durchläuft, vermindert sich der Wert um (mindestens) 1. Router verwerfen ein Paket mit der TTL 0. Dieser Mechanismus verhindert »unzustellbaren Datenmüll« und Nachrichten, die endlos im Internet kreisen.
Protokoll	8	Angabe des <i>Upper Layer Protocols</i> , des eine OSI-Schicht höher liegenden Protokolls Die Werte sind gemäß RFC 3232 in einer Datenbank hinterlegt. Beispiele: 6 = TCP, 17 = UDP, 1 = ICMP
Header-Prüfsumme	16	Prüfsumme (gilt ausschließlich für den Header, nicht für die folgende Nutzlast)
Sender-IP-Adresse	32	IP-Adresse des Absenders
Ziel-IP-Adresse	32	IP-Adresse des Empfängers
Optionen	2	Angaben zu Routing- und Diagnosezwecken
Padding	*	eventuell notwendige Füllbits zum Erreichen der vorgeschriebenen Bitzahl

Tabelle 3.21 Inhalt des IPv4-Headers (Forts.)

3.7.2 Der IPv6-Header

Der IPv6-Header (Abbildung 3.7) unterscheidet sich deutlich vom älteren IPv4-Format. So hat er eine feste Größe von 320 Bit. Weitere Informationen finden in dem erweiterten Kopfdatenbereich ihren Platz, der sich zwischen dem Header und dem Nutzdatenbereich befindet.

Tabelle 3.22 erläutert Ihnen die einzelnen Felder des IPv6-Headers. Einzelheiten zur IPv6-Adresse sind in den RFCs **2460**, 5095, 5722, 5871, 6437, 6564, 6935, 6946, 7045 und 7112 niedergeschrieben.

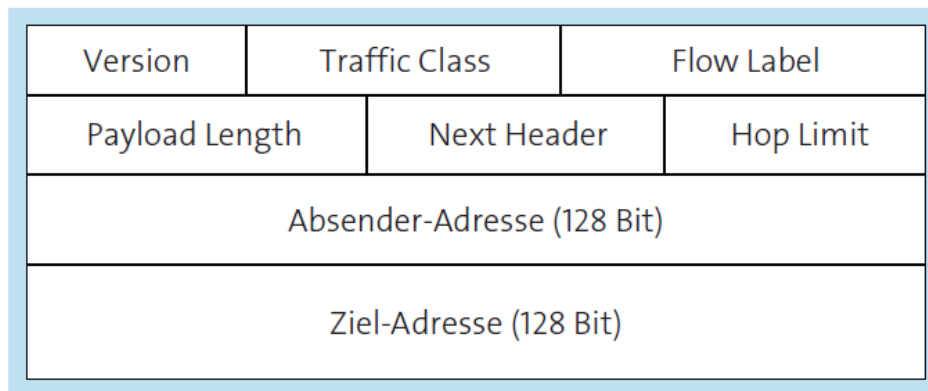


Abbildung 3.7 IPv6-Header

Feld	Länge (Bits)	Inhalt
Version	4	IP-Protokollversion (6)
Traffic Class	8	<i>Quality of Service</i> (QoS), Kennzeichnung der Priorität
Flow Label	20	Markieren von Paketen gleicher Verwendung und Behandlung (QoS). Zufallswerte, möglicher Bereich von 00001 bis FFFFF. Pakete ohne Eintrag durch Absender führen alle Bits mit 0. Pakete desselben Flows müssen stets die gleiche Absender- und Empfängeradresse tragen, sonst wird das Flow-Label nicht ausgewertet. Weitere Informationen finden Sie in RFC 6437.
Payload Length	16	Länge der Daten nach dem IPv6-Header (maximal 64 KB; Ausnahme: Jumbogramm nach RFC 2675)
Next Header	8	Angabe des Folgeprotokolls (6 für TCP, 17 für UDP). Die Datenbank finden Sie unter http://www.iana.org/assignments/protocol-numbers/protocol-numbers.xml .
Hop-Limit	8	Anzahl der maximalen Router-Sprünge (Hops). Wird der Wert überschritten, wird das Paket verworfen, und der Absender erhält eine ICPMv6-Nachricht. Bei jedem Hop vermindert sich der Wert um 1.
Absender-adresse	128	Angabe zwingend
Zieladresse	128	kann auch nur die Adresse des nächsten Hops enthalten

Tabelle 3.22 IPv6-Header

3.8 Prüfungsfragen

Die Auflösungen finden Sie in Anhang B, »Auflösungen zu den Prüfungsfragen«.

3.8.1 Berechnungen

Ein Host hat die Adressangabe 192.168.5.65/23.

1. Über wie viele Hosts kann dieses Netz maximal verfügen?
2. Wie lautet die Netzwerkadresse dieses Netzes?
3. Wie lautet die erste und die letzte IP-Adresse für Hosts in diesem Netz?

3.8.2 IP-Adressen

Warum gelingt der direkte Zugriff auf die Adresse
fe80::218:e7ff:fe16:7130/64
aus dem Internet nicht?

4.1 MAC-Adressen

In Ihrer täglichen Netzwerkpraxis werden Sie vor allem bei der Einbindung neuer Geräte in Ihr LAN mit der MAC-Adresse zu tun haben. Sie werden die neuen Geräte anhand dieser Adresse identifizieren und die IP-Adresszuweisung vornehmen.

4.1.1 Ermitteln der MAC-Adresse

Die MAC-Adresse eines Netzwerkgeräts finden Sie im einfachsten Fall auf einem Aufkleber neben der Anschlussbuchse (Abbildung 4.1). Sie finden die Information vielleicht auch in das Gehäuse des *Netzwerk-Chips* fest eingraviert. Eventuell ist sie auch auf der Baugruppe aufgedruckt oder mit einem kleinen Aufkleber angebracht.



Abbildung 4.1 Aufkleber mit MAC-Adresse an einem internen Printserver

Wenn Sie immer noch nicht fündig geworden sind, müssen Sie die MAC-Adresse über das Betriebssystem selbst ermitteln. Handelt es sich dabei um einen PC, melden Sie sich dazu am betreffenden System mit der *Systembetreuererkennung* (root bzw. Administrator) an. Die notwendigen Befehle zur Abfrage finden Sie in Tabelle 4.1.

Betriebssystem	Kommando(s)
Linux	ifconfig -a ip link show <Gerät>
FreeBSD	ifconfig -a
Windows Vista, 7, 8, 8.1, 10	ipconfig /all

Tabelle 4.1 Ermitteln der »eigenen« MAC-Adresse

Beispiel: Ermitteln der eigenen MAC-Adresse

```
RE2# ifconfig -a
.....
re0: flags=8843<UP,BROADCAST,RUNNING,SIMPLEX,MULTICAST>
.....
ether 00:11:6b:62:93:2e
.....
media: Ethernet autoselect (1000baseT <full-duplex>)
status: active
```

Kontrollieren Sie immer nach, ob Sie auch das richtige Netzwerkgerät auslesen, da ein Rechner meist über mehrere davon verfügt (z. B. lo).

Können Sie die MAC-Adresse eines Geräts nicht mit der Abfrage nach Tabelle 4.1 vornehmen, verbinden Sie es mit einem PC per Cross-over-Kabel bzw. kreuzen bei Glasfaseranschlüssen beide Adern. Damit liegen beide Geräte in einem einzigen Netzwerksegment. Anschließend verwenden Sie in Abhängigkeit des von Ihnen benutzten Betriebssystems eines der Kommandos aus Tabelle 4.2.

Betriebssystem	Kommando(s)
Linux	arp -a ip neigh
FreeBSD	arp -a
Windows Vista, 7, 8, 8.1, 10	arp -a

Tabelle 4.2 Ermitteln einer MAC-Adresse von einem anderen System aus

Beispiel: Ermitteln einer MAC-Adresse von einem anderen System aus

```
RE1:~# ip neigh
192.168.0.102 dev eth0 lladdr 00:11:6b:62:93:2e REACHABLE

RE1:~# arp
Address      HWtype  HWaddress      Flags Mask    Iface
192.168.0.102 ether    00:11:6b:62:93:2e C             eth0
```

Nicht in jedem Fall kommen Sie mit diesen Standardmitteln weiter. Mancher Printserver z. B. lässt sich das erste Mal nur mit einer Spezialsoftware konfigurieren. Dabei müssten Sie »nur« eine IP-Adresse zuweisen, wofür Sie ja die MAC-Adressangabe benötigen. Vielleicht haben Sie einen managbaren Switch? Schließen Sie das Gerät, das anonym bleiben soll, an einem freien Port an. Im Verwaltungsmenü des Switchs können Sie

die MAC-Adresse anschließend auslesen. Meist finden Sie diese unter Menüs wie Dynamic Forwarding Table oder MAC-Table.

4.1.2 Ändern der MAC-Adresse

Sie können das Ändern von MAC-Adressen im Rahmen von Hochverfügbarkeitsszenarien sinnvoll nutzen. Fällt ein Server aus, leiten Sie damit ohne große Umkonfiguration der Systeme den Netzwerkverkehr auf die Redundanz um. Auch bei der Konfiguration von Reservegeräten kann diese Vorgehensweise für Sie nützlich sein. Die notwendigen Kommandos finden Sie in Tabelle 4.3.

Betriebssystem	Kommando
Linux	<code>ifconfig <Gerät> hw ether <Mac-Adresse-neu></code>
FreeBSD	<code>ifconfig <Gerät> link <Mac-Adresse-neu></code>
Windows Vista, 7, 8, 8.1, 10	Verändern des Registry-Eintrags oder, falls angeboten, im Eintrag bei der Systemsteuerung

Tabelle 4.3 Verändern der MAC-Adresse

Beispiel: Ändern der MAC-Adresse einer Reserve-Netzwerkkarte unter FreeBSD

```
RE2# ifconfig rl0
rl0: .....
     ether 00:02:44:50:14:47
     .....
```

```
RE2# ifconfig rl0 link 00:02:44:50:13:45
```

```
RE2# ifconfig rl0
rl0: .....
     ether 00:02:44:50:13:45
     .....
```

4.1.3 Manuelles Setzen und Ändern von MAC-Adressen mittels »arp«

Mit dem arp-Kommando können Sie auch Einträge im ARP-Cache setzen und löschen. Es ist sowohl in der Unix-Welt als auch auf Windows-Rechnern vorhanden. Die Befehlsoptionen sind alle ähnlich.

Mit `arp -s <IP-Adresse> <Mac-Adresse>` werden Einträge gesetzt, mit `arp -d <IP-Adresse>` gelöscht.

4.1.4 ARP-Spoofing erkennen

Das (böswillige) Ändern der MAC-Adresse ist auch unter dem Begriff *ARP-Spoofing* bekannt. Angreifer bedienen sich dieser Methode, wenn sie eine »*Man-in-the-Middle*«-Attacke beginnen wollen.

Einen Angriff auf Ihr Netzwerk durch einen manipulierten ARP-Cache erkennen Sie daran, dass alle IP-Adressen auf eine einzige MAC-Adresse zeigen.

Beispiel: ARP-Cache nach Spoofing (»vergifteter« ARP-Cache)

```
RE1:~# arp -a
? (192.168.0.52) auf 00:1b:11:1f:9e:86 [ether] auf eth0
? (192.168.0.53) auf 00:1b:11:1f:9e:86 [ether] auf eth0
? (192.168.0.54) auf 00:1b:11:1f:9e:86 [ether] auf eth0
? (192.168.0.55) auf 00:1b:11:1f:9e:86 [ether] auf eth0
```

Wenn Sie auf Derartiges stoßen, müssen Sie sofort handeln. Ihre Systeme sind in größter Gefahr!

4.2 IP-Adressen setzen

Sie können IP-Adressen manuell direkt am Client oder automatisiert im Netz zuweisen. Für größere Netze sollten Sie vorher eine schlüssige Planung hinsichtlich der Adressen selbst vornehmen. Unterteilen Sie Ihr Netz nach Kriterien wie Filialen, Aufgabenbereichen, Infrastruktur oder auch Datenschutz. Sie können damit auch an Übersichtlichkeit gewinnen. Auch die Gerätenamen, die wiederum mit den IP-Adressen zusammengeführt werden, beziehen Sie in diese Planungen mit ein. Sie finden in Tabelle 4.4 Anhaltspunkte, die Ihnen vielleicht bei Ihrer Netzplanung helfen können.

Konfigurationsfrage	Problematik/Hinweise
IPv4- oder IPv6-Adressen	Kommen reine (ältere) IPv4-Geräte vor (Printserver, NAS, Router ...)? Das interne »Verwaltungsnetz« und das »Arbeitsnetz« können verschiedene IP-Adressversionen haben.
Feste IP-Adressen ohne DHCP-Server	Pro: keine zentrale Instanz, die ausfallen kann (DHCP-Server), hohe Ausfall- und Sabotagesicherheit Contra: Übersichtlichkeitsprobleme möglich, zum Teil umständliche Verwaltung, da jedes Gerät einzeln bearbeitet werden muss
Feste IP-Adresse über DHCP-Server	Pro: vereinfachte netzweite Geräteverwaltung, sogar mit Datenbankeinsatz möglich Contra: Bei einem Ausfall des DHCP-Servers ist kein Netzwerkbetrieb möglich.
IP-Adresse über DHCP-Server (wechselnde Adresse)	Pro: Endgerät ist nicht über IP-Adresse identifizierbar. Contra: nicht für Geräte geeignet, wie z. B. Printserver
Unterteilung in Teilnetze oder verschiedene Netze	Pro: Abbildung von betrieblichen Strukturen, Erhöhung des Datenschutzniveaus Contra: Geräte benötigen bei »Umzug« in ein anderes Teilnetz eine neue IP-Adresse; eine genauere Planung ist notwendig.

Tabelle 4.4 Hinweise zu Adresszuweisung und Netzkonfiguration

Neben der IP-Adresse müssen Sie auch noch die richtige Netzmaske setzen (siehe Abschnitt 3.5.2, »Netz- und Subnetzmaske, Unterteilung von Netzen«). Im Beispiel möchten Sie zwei IPv4-Teilnetze mit maximal 62 Teilnehmern anlegen:

Beispielaufgabe: Berechnung der Subnetzmaske mit »ipcalc«

Netz 1: 192.168.0.0 – 192.168.0.63

Netz 2: 192.168.0.64 – 192.168.0.127

Verwenden Sie dazu das Werkzeug *ipcalc*. Sie können das kleine Programm für viele Betriebssysteme bekommen. Es erspart Ihnen die Rechnerei mit den Binärzahlen.

Lösungsweg

- ▶ Ermittlung der Netzmaske:
 Eingabe: `ipcalc 192.168.0.0 – 192.168.0.63`
 Ausgabe: `deaggregate 192.168.0.0 – 192.168.0.63`
 192.168.0.0/26
 Die Netzmaske in CIDR-Schreibweise lautet /26.
- ▶ Die Berechnung des zweiten Adressblocks führt zum gleichen Ergebnis.
- ▶ Ausgabe der Subnetzmaske in Nicht-CIDR-Schreibweise:
 Eingabe: `ipcalc /26`
 Ausgabe: `Address: 255.255.255.192`
 11111111.11111111.11111111.11000000

In Abschnitt 3.5.3, »Berechnungen«, finden Sie weitere Berechnungen mit *ipcalc*.

4.2.1 Netzwerkkonfiguration von PCs

Für das Setzen von IP-Adressen und weiteren Einstellungen benutzen Sie meist die Werkzeuge der grafischen Benutzeroberfläche (kurz GUI für *Graphical User Interface*). Diese sind Ihnen sicher bekannt, deshalb gehe ich nur kurz darauf ein.

Windows 7, 8/8.1 und 10

In Windows 7 klicken Sie folgenden Weg: Systemsteuerung • Netzwerk und Internet • Netzwerk- und Freigabecenter • Verbindung mit einem Netzwerk herstellen (Abbildung 4.2 bis Abbildung 4.4). Der Rechner muss dabei an das Netzwerk angeschlossen sein – im Voraus konfigurieren ohne Netzwerkanschluss funktioniert hier leider nicht.

Sie können nun die Eigenschaften der LAN-Verbindung sehen. Wählen Sie die gewünschte Netzwerkkarte und anschließend das Internetprotokoll Version 4 oder Internetprotokoll Version 6. Anschließend klicken Sie auf Eigenschaften (Abbildung 4.3).

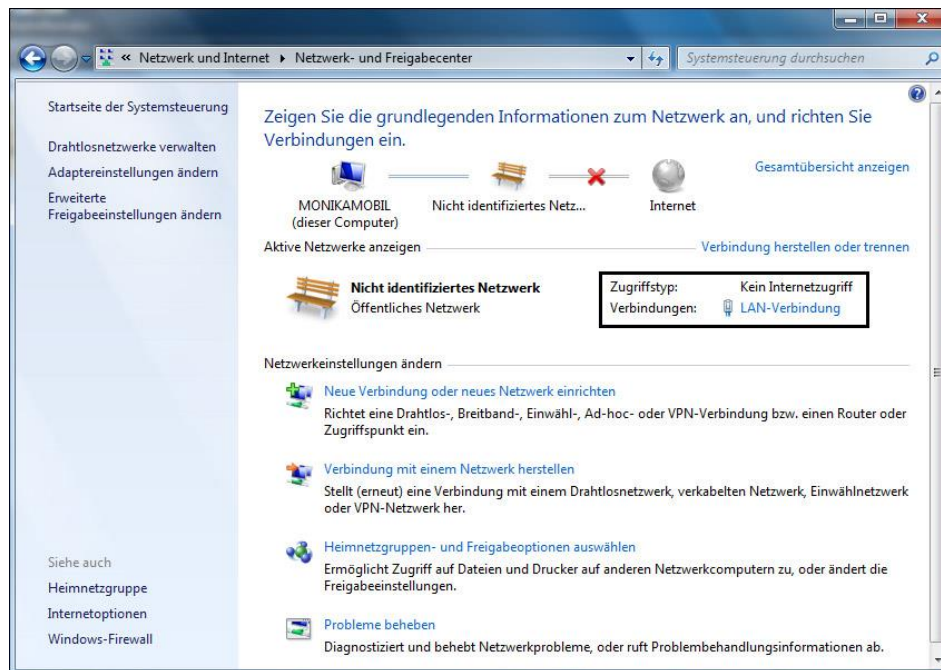


Abbildung 4.2 Netzwerk- und Freigabecenter, Menüpunkt »LAN-Verbindung« wählen

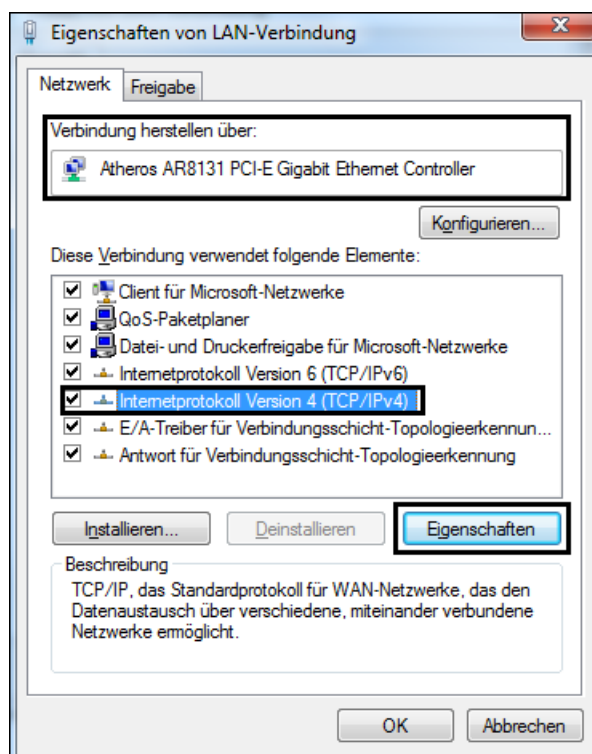


Abbildung 4.3 Eigenschaften von LAN-Verbindung (Windows 7)

Egal, für welche IP-Version Sie sich entschieden haben, hier bestimmen Sie nun, ob sich der Rechner die Adresse von einem DHCP-Server abholt oder eine von Ihnen festgelegte Adresse bekommt (Abbildung 4.4).

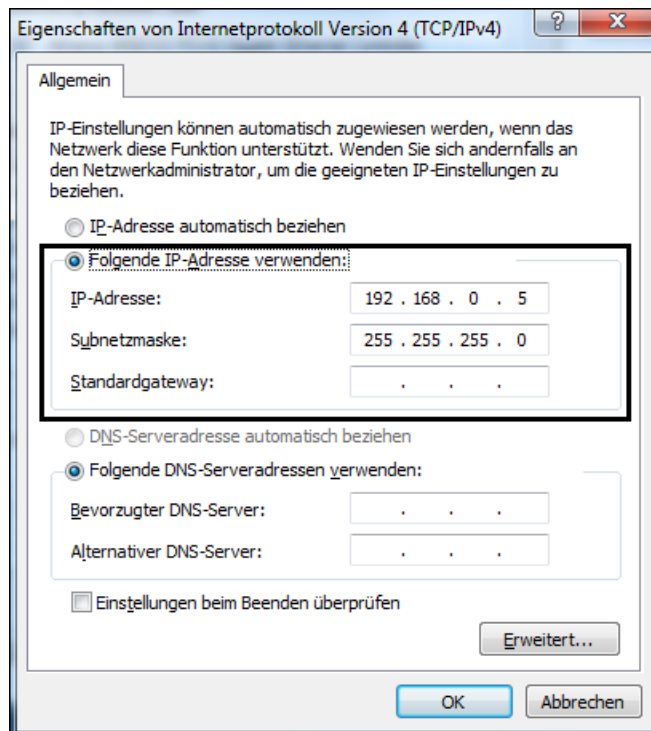


Abbildung 4.4 Eintragen der IP-Adresse oder DHCP-Zugriff

In Windows 8/8.1 und 10 erreichen Sie das Netzwerk- und Freigabecenter am einfachsten über die Suche. Unter Windows 8/8.1 öffnen Sie mit (Win) + (C) die Charmsbar und klicken auf die mit Suchen beschriftete Lupe, unter Windows 10 erreichen Sie das Suchfeld mit der (Win)-Taste.

Geben Sie dort »Netzwerk u« ein; daraufhin bietet Ihnen die Suche bereits das gesuchte Netzwerk- und Freigabecenter als Treffer an. Wenn Sie darauf klicken, öffnet sich ein Fenster wie in Abbildung 4.3. Die weiteren erforderlichen Schritte entsprechen den hier für Windows 7 ausgeführten.

Welchen Eintrag Sie in das Feld Standardgateway setzen müssen, erfahren Sie in Abschnitt 4.7.2, »Festlegen des Standard-Gateways«. Meist wird hier die Adresse des DSL-Routers eingetragen, zumindest bei kleinen Netzen. Windows bietet in dieser Maske auch den Nameserver-Eintrag an. Dies dient der Auflösung von Rechnernamen zu IP-Adressen. Was dies bedeutet, lesen Sie in Abschnitt 4.3.

Linux

Unter Linux bieten einige Benutzeroberflächen wie *Unity*, *Gnome* oder *KDE* direkten Konfigurationszugriff. Auch hier gelangen Sie über Systemmenüs zur Netzwerkkonfiguration, wobei Sie die Netzwerkschnittstelle auswählen können. Die Einträge für diese entsprechen meist denen von Abbildung 4.2 bis Abbildung 4.4.

Abseits des GUIs können Sie die Konfiguration direkt per Texteditor vornehmen. Damit können Sie bei Masseninstallationen sehr schnell diesen Teil der Konfiguration durchlaufen lassen, indem Sie nur noch per Skript oder Datenbank erzeugte Konfigurationsdateien am vorgesehenen Ort ablegen. Sind Sie viel mit dem Notebook auf Reisen und haben Sie in verschiedenen Netzen eine feste Adresse, brauchen Sie nur die Datei mit den jeweils gültigen Einstellungen zu kopieren und das Netzwerk neu zu starten.

Bei Debian-Linux ist die Datei *interfaces* im Verzeichnis */etc/network* abgelegt. Sie enthält für jede Netzwerkschnittstelle Einträge. Sie sehen im Beispiel Standardeinträge für die Loopback-Netzwerkschnittstelle, für eine Netzwerkkarte mit fester IP-Adresse und für eine, die die Adresse per DHCP bezieht. Die auto-Einträge bewirken, dass die jeweilige Netzwerkschnittstelle bereits beim Systemstart aktiviert wird. Wenn Sie diesen Zusatz weglassen, können Sie die Schnittstelle per Hand (oder auch über ein Shell-

Skript zu gegebener Zeit) aktivieren. Damit können Sie z. B. einen nur zeitweise »offenen
 « Wartungszugang freischalten oder Tests mit Netzwerkanwendungen durchführen, die nur auf diese Netzwerkschnittstelle zugreifen. Genauere Angaben zu den Feinheiten erhalten Sie, indem Sie die Manual-Seiten hierzu lesen (man interfaces).

Konfigurationsdatei »/etc/network/interfaces«

- ▶ Loopback-Schnittstelle


```
auto lo
iface lo inet loopback
```
- ▶ Netzwerkkarte mit fester IP-Adresse


```
auto eth0
iface eth0 inet static
address 192.168.0.34
netmask 255.255.255.0
gateway 192.168.0.1
```
- ▶ zweite Netzwerkkarte mit IP-Adresse über DHCP-Server


```
auto eth1
iface eth0-work inet dhcp
```

Nach jeder Änderung dieser Datei müssen Sie das Netzwerk neu starten. Unter der Benutzerkennung root geben Sie dazu die Anweisung

```
/etc/init.d/networking restart
```

in der Shell ein. Prüfen Sie anschließend, ob alle von Ihnen konfigurierten Netzwerkschnittstellen wie gewünscht arbeiten. Benutzen Sie dazu das Kommando `ifconfig` (Abbildung 4.5). Zusammen mit dem `ip`-Befehl können Sie damit viele Netzwerkkonfigurationsarbeiten vornehmen.

```
ZE4:~# ifconfig
eth0      Link encap:Ethernet  Hardware Adresse 00:23:54:5b:86:9d
          inet Adresse:192.168.0.34  Bcast:192.168.0.255  Maske:255.255.255.0
          inet6-Adresse: fe80::223:54ff:fe5b:869d/64  Gültigkeitsbereich:Verbindung
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metrik:1
          RX packets:125273 errors:0 dropped:0 overruns:0 frame:0
          TX packets:115321 errors:0 dropped:0 overruns:0 carrier:0
          Kollisionen:0 Sendewarteschlangenlänge:1000
          RX bytes:144516723 (137.8 MiB)  TX bytes:11594080 (11.0 MiB)
          Interrupt:251

eth1      Link encap:Ethernet  Hardware Adresse 00:18:e7:16:71:30
          inet Adresse:192.168.0.103  Bcast:255.255.255.255  Maske:255.255.255.0
          UP BROADCAST RUNNING MULTICAST  MTU:576  Metrik:1
          RX packets:121 errors:0 dropped:0 overruns:0 frame:0
          TX packets:223 errors:0 dropped:0 overruns:0 carrier:0
          Kollisionen:0 Sendewarteschlangenlänge:1000
          RX bytes:27456 (26.8 KiB)  TX bytes:32982 (32.2 KiB)
          Interrupt:17 Basisadresse:0x2c00

lo        Link encap:Lokale Schleife
          inet Adresse:127.0.0.1  Maske:255.0.0.0
          inet6-Adresse: ::1/128  Gültigkeitsbereich:Maschine
          UP LOOPBACK RUNNING  MTU:16436  Metrik:1
          RX packets:83771 errors:0 dropped:0 overruns:0 frame:0
          TX packets:83771 errors:0 dropped:0 overruns:0 carrier:0
          Kollisionen:0 Sendewarteschlangenlänge:0
          RX bytes:11951043 (11.3 MiB)  TX bytes:11951043 (11.3 MiB)

ZE4:~#
```

Abbildung 4.5 Ausgabe von »ifconfig«

Sehen Sie sich die Ausgabe in Abbildung 4.5 genau an. Beim Loopback-Gerät und eth0 sehen Sie auch jeweils die IPv6-Adresse. Diese Adressen werden automatisch konfiguriert,

weil bei der Installation des Systems die IPv6-Unterstützung aktiviert wurde. Die Konfiguration dazu ist bei *Debian Lenny* und *Debian Squeeze* unterschiedlich umgesetzt.

Wenn Sie hier eingreifen möchten, sollten Sie unbedingt auch das zutreffende Systemhandbuch zurate ziehen. Ferner sehen Sie in der Ausgabe die MAC-Adresse, die Menge bereits transportierter Pakete und weitere Konfigurationseinstellungen, die normalerweise aber nicht geändert werden müssen, z. B. die MTU-Größe.

Bei der IPv6-Adresse handelt es sich um eine Link-local Unicast-Adresse (Tabelle 3.20). Wenn Sie jetzt eine feste IPv6-Adresse (privat, Unique-local Unicast-Adresse) zusätzlich vergeben wollen, ergänzen Sie einfach den Eintrag in *interfaces* beim Eintrag zu *eth0*:

Ergänzung »/etc/network/interfaces«

```
iface eth0 inet6 static
address fd91:e4a5:5e03::5
netmask 48
gateway fd91:e4a5:5e03::1
```

Die IP-Konfiguration für Linux ist damit abgeschlossen.

FreeBSD

Unter FreeBSD nehmen Sie die Netzwerkeinstellungen in der zentralen Konfigurationsdatei */etc/rc.conf* vor.

Eine IPv4-Adresse benötigt folgende Einträge:

»/etc/rc.conf«, Eintrag einer IPv4-Adresse

```
ifconfig_re0="inet 192.168.0.25 netmask 255.255.255.0"
defaultrouter="192.168.0.1"
```

Eine Besonderheit unter FreeBSD sind die Jails. Sie können damit eigene, vom Rest des Systems abgeschottete Umgebungen einrichten. Damit diese Zugriff auf das Netzwerk erhalten, werden Aliase verwendet:

»/etc/rc.conf«, Aliase für Jails

```
# -- Aliase für Jails
ifconfig_re0_alias0="inet 192.168.0.65 netmask 255.255.255.255"
ifconfig_re0_alias1="inet 192.168.0.66 netmask 255.255.255.255"
ifconfig_re0_alias2="inet 192.168.0.67 netmask 255.255.255.255"
```

Den Adressbezug vom DHCP-Server fordern Sie mit dieser Zeile für eine Netzwerkkarte an:

»/etc/rc.conf«, Netzwerkkarte mit Adressbezug vom DHCP-Server

```
ifconfig_r10="DHCP"
```

Eine IPv6-Adresse tragen Sie wie folgt ein:

»/etc/rc.conf«, IPv6-Adresse, statisch

```
ipv6_enable="YES"
ipv6_network_interfaces="ae0"
ipv6_ifconfig_ae0="fd91:e4a5:5e03::dd04 prefixlen 64"
ipv6_defaultrouter="fd91:e4a5:5e03::dd01"
```

Nachdem Sie neue Einträge gesetzt oder bestehende geändert haben, werden sie nach dem Neustart des Netzwerks wirksam. Für IPv4-bezogene Einträge verwenden Sie:

```
/etc/rc.d/zentrale/netif restart
```

IPv6-Einstellungen geben Sie dem System durch den Aufruf

```
/etc/rc.d/network_ipv6 restart
```

bekannt. Anschließend kann es notwendig sein, auch

```
/etc/rc.d/routing restart
```

auszuführen. Wie unter Linux können Sie mit dem `ifconfig`-Kommando die aktuell geladenen Netzwerkeinstellungen abfragen. Dieses Kommando hilft Ihnen aber nicht nur bei der Abfrage von Netzwerkeinstellungen, sondern auch bei vielen Konfigurationsarbeiten bis hin zum WLAN.

OS X

Klicken Sie auf **Apple** • Systemeinstellungen • Netzwerk und hier in der linken Spalte auf Ethernet (Abbildung 4.6).

Hier wählen Sie im Punkt Konfiguration die gewünschte Variante aus: DHCP, DHCP mit manueller Adresse, BootP, Manuell, Aus oder PPPoE-Dienst erstellen.

Klicken Sie auf Weitere Optionen. Hier können Sie weitere Verfeinerungen vornehmen und auch eine IPv6-Adresse konfigurieren (Abbildung 4.7).

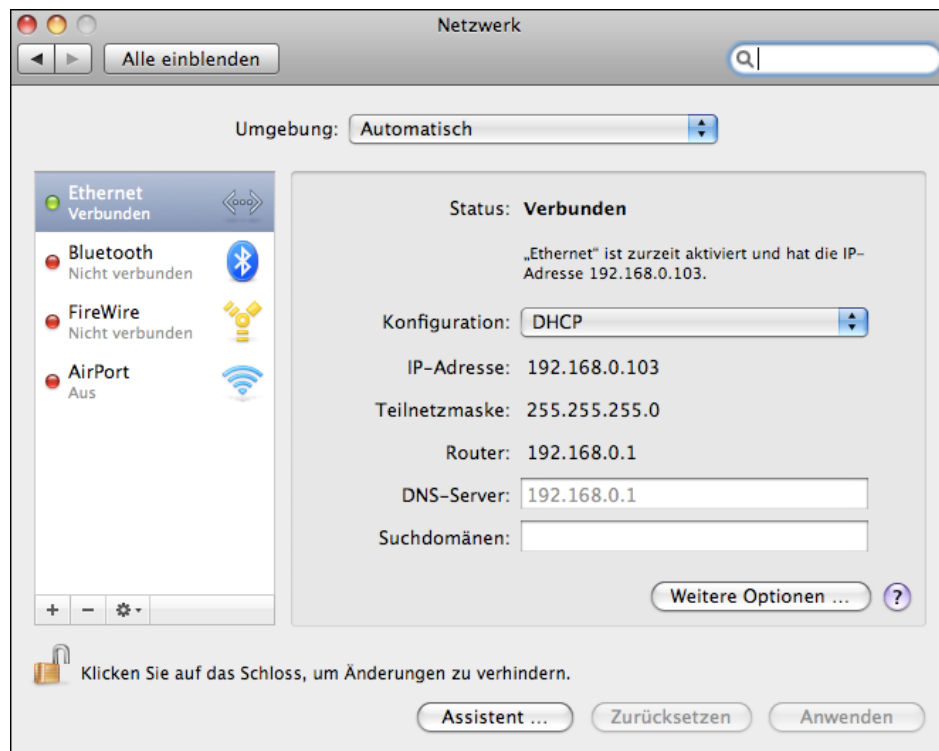


Abbildung 4.6 Auswahl der Netzwerkschnittstelle und Konfiguration

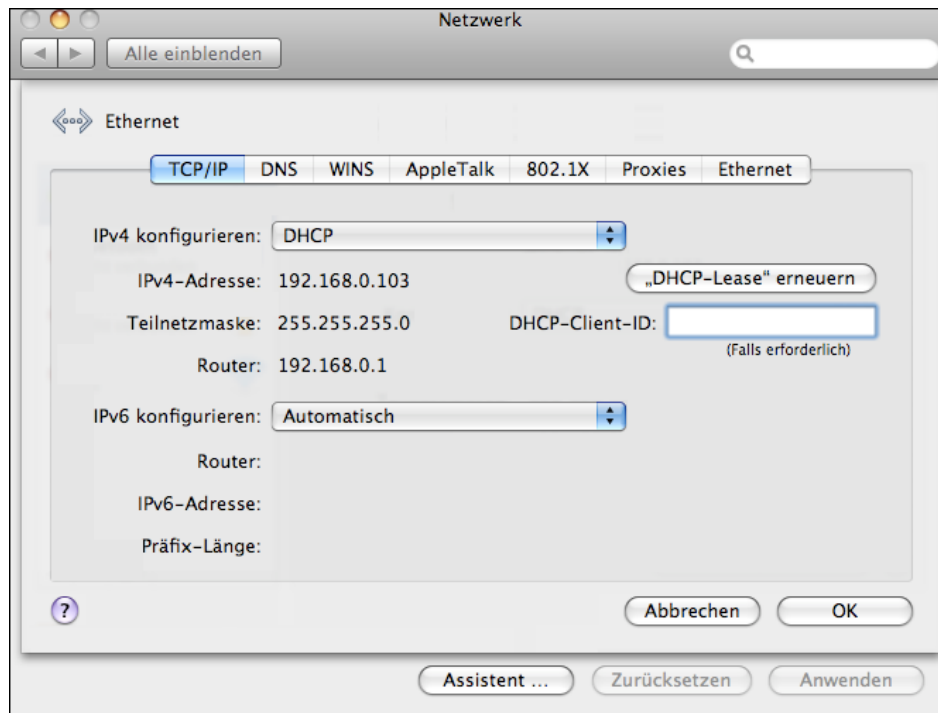


Abbildung 4.7 Feinkonfiguration der TCP/IP-Adressen

Die MAC-Adresse und andere physikalische Einstellungen sehen Sie, wenn Sie die Karteikarte Ethernet auswählen. Natürlich können Sie hier entsprechende Änderungen vornehmen (Abbildung 4.8).

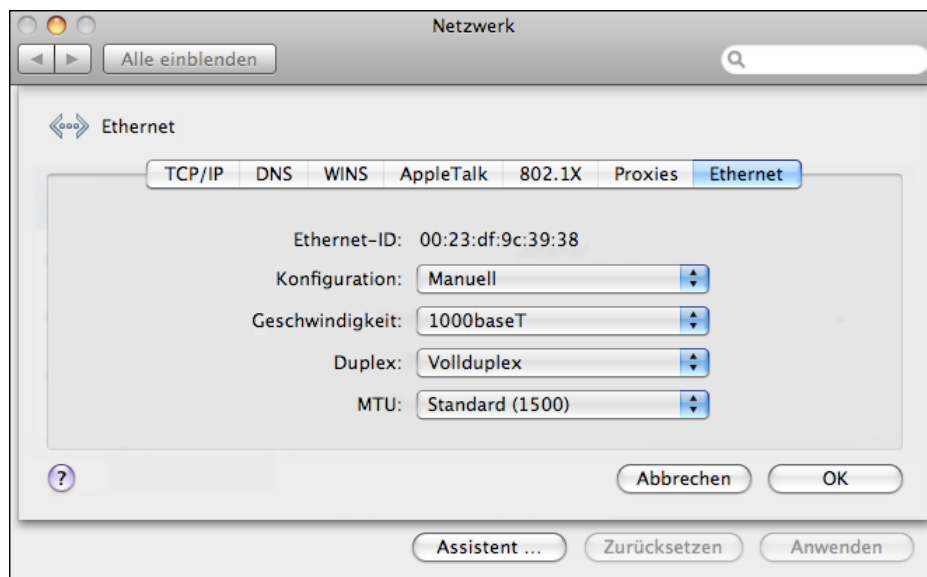


Abbildung 4.8 Einstellungen der Netzwerkschnittstelle

4.2.2 IP-Adresskonfiguration von weiteren Netzwerkgeräten

Im Netzwerk finden Sie weitere Geräte vor, denen Sie eine IP-Adresse zuweisen müssen: Konfigurationszugänge von managbaren Switches und Routern, Printservern, Kameras, NAS-Boxen bis hin zu Telefonanlagen und Heizungssteuerungen.

Speziell bei den Printservern begegnen Ihnen eingebaute und externe Modelle. Wenn der Drucker über ein Bediendisplay verfügt, wie es in Abbildung 4.9 gezeigt wird, können Sie die Adresse direkt hierüber setzen. Bei allen anderen Geräten (nicht nur bei den Printservern) hilft Ihnen nur ein Blick in die Installationsanleitung. Sie erfahren daraus die voreingestellte IP-Adresse oder erhalten einen Hinweis darauf, ob das Gerät per DHCP seine Adresse bezieht. Manchmal müssen Sie sogar ein spezielles Konfigurationsprogramm

unter Windows bemühen, damit Sie erstmals eine Adresse auf dem neuen Gerät setzen können.

Alle diese Geräte bieten Ihnen auch eine Weboberfläche zur weiteren Einstellung. In Abbildung 4.10 sehen Sie die Netzwerkkonfiguration eines Printservers, der beide IPProtokollversionen benutzen kann.



Abbildung 4.9 Netzwerkkonfiguration eines »Kyocera Mita FS-1370DN« über das Bedienfeld

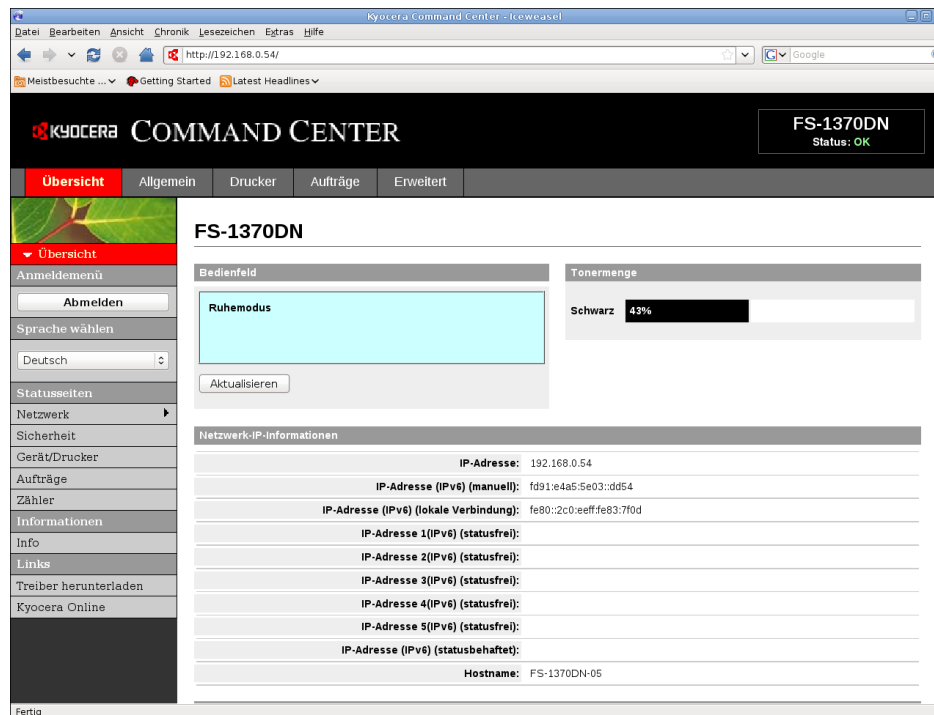


Abbildung 4.10 Konfiguration des Printservers eines »Kyocera Mita FS-1370DN« für IPv4 und IPv6

Ändern Sie im Rahmen dieser Erstkonfiguration in jedem Fall das Standardkennwort, das der Hersteller vergeben hat! »Offene« Netzwerkgeräte stellen ein großes Sicherheitsproblem dar und gefährden Ihr Netzwerk.

Sie werden bei einer Umstellung Ihres Netzwerks von IPv4 auf IPv6 auf die »Netzwerkkleingeräte« Rücksicht nehmen müssen. Die älteren und selbst viele der (2016) neu ausgelieferten Printserver und Switches beherrschen noch kein IPv6. Firmware-Updates hierfür werden Sie eher selten finden. Ich möchte keine Missverständnisse aufkommen lassen und bezüglich der Switches klarstellen, dass diese selbstverständlich IPv6-Pakete verarbeiten, aber der Konfigurations-Port ist in den meisten Fällen eben nur IPv4-tauglich.

Das Problem können Sie auf verschiedene Arten lösen:

Sie können die Server und PCs Ihres Netzwerks für den Dual-Stack-Betrieb konfigurieren. IPv4- und IPv6-Adressen werden gleichberechtigt behandelt.

Bezüglich der Printserver haben Sie die Möglichkeit, die alten Geräte durch stromsparende Kleinrechner wie den Raspberry Pi oder den Banana Pi zu ersetzen, wenn es räumlich möglich ist, mehrere Drucker zusammen unterzubringen. Ein Kleinrechner verfügt über mehrere USB-Ports. Daran schließen Sie mehrere Drucker an. Sowohl hinsichtlich der Anschaffungskosten als auch des Stromverbrauchs haben Sie keine Mehrkosten gegenüber den einzelnen Printservern zu erwarten. Ferner haben Sie auch die mit vielen Printserver-Modellen einhergehenden Sicherheitsprobleme aus Ihrem Netz verbannt. Sie haben damit einen schnellen, flexiblen IPv6-Druckserver geschaffen. Eine einfache Installation von Linux oder FreeBSD reicht vollkommen aus. Die Drucker werden auf diesem Gerät installiert bzw. die Warteschlangen darauf angelegt. Meist werden Sie den Druckerspools cups einrichten, der die PPD-Steuerdateien der Druckerhersteller direkt benutzt.

Falls Sie sowieso mit einem »Schattennetz« zur Verwaltung der Switches, Router und anderer Netzwerkkomponenten arbeiten, belassen Sie darauf einfach IPv4, solange die vorhandenen Geräte ihren Dienst verrichten.

4.2.3 Zentrale IP-Adressverwaltung mit dem DHCP-Server

Mit dem *Dynamic Host Configuration Protocol (DHCP)* können Sie über einen zentralen

Rechner (fast alle) Netzwerkgeräte konfigurieren.

An DHCP-Servern kommt niemand im täglichen Internetleben vorbei. Die meisten Internet-Service-Provider teilen Ihnen bei der Einwahl für die Zeit der Verbindung eine IP-Adresse aus einem Pool zu. Meist darf Ihr Rechner oder Router diese Adresse für maximal 24 Stunden behalten – dann vergibt der Dienstleister eine neue Adresse.

Auch für Ihre lokalen Netzwerke bietet DHCP den einen oder anderen Vorteil, aber auch die Nachteile sollen keinesfalls verschwiegen werden:

zentrale Netzwerkadministration, auch mit datenbankbasierter Inventarverwaltung

einfache Konfiguration der Clients: Sie müssen nur die Einstellung IP-Adresse automatisch beziehen setzen.

Zusammen mit dem *Bootstrap Protocol (BOOTP)* können Sie sogar plattenlose Rechner betreiben – hier müssen Sie am Client fast nichts mehr konfigurieren (Sie müssen nur die Bootreihenfolge im BIOS einstellen).

Fällt Ihr DHCP-Server aus, bekommen später hinzukommende Clients keine IP-Adresse.

Ein zusätzlicher, böswillig an das Netz angeschlossener DHCP-Server vergibt Adressen zum Zweck eines Angriffs auf das Netzwerk (*Rogue DHCP*).

Eine andere Angriffsmethode besteht darin, den DHCP-Server so lange um Adressvergaben zu bitten, bis der komplette Pool belegt ist. Clients bekommen dann keine Adresse mehr zugewiesen.

Wenn Sie den DHCP-Server so konfigurieren, dass eine feste Zuordnung von MAC zur IP-Adresse vorgenommen wird, erhöht dies nicht die Netzwerksicherheit. Böswillige Menschen ändern entweder die MAC-Adresse passend ab oder setzen gleich eine feste IP-Adresse.

Für jedes Subnetz müssen Sie einen eigenen DHCP-Server betreiben (da innerhalb einer Broadcast-Domäne befindlich). Soll der DHCP-Server über das Subnetz erreicht werden, müssen Sie den Router, der das Subnetz mit anderen Netzen verbindet, als DHCP-Relay konfigurieren. In diesem Fall leitet er die Anfragen und Antworten (als Unicast-Nachricht) über die Netzwerkgrenzen hinweg.

Sehen Sie sich den Leistungsumfang des DHCP-Protokolls an. Mit DHCP können Sie noch mehr tun, als IP-Adressen zuzuweisen:

Dynamic Host Configuration Protocol (DHCP)

Zuweisung von:

- ▶ IP-Adresse
- ▶ Netzwerkmaske (Subnetzmaske)
- ▶ Default-Gateway
- ▶ WINS-Server für Microsoft-Rechner
- ▶ DNS-Server (Nameserver)
- ▶ Zeit- und NTP-Server
- ▶ Proxykonfiguration über WPAD

Belegte Ports: UDP 67 (Server), UDP 68 (Client)

RFCs zum Nachlesen: 2131, 2132, 3315, 3396, 3442, 3942, 4361, 4833, 5494 und 6842

Manuelle Adresszuweisung: Für eine angegebene MAC-Adresse wird immer dieselbe IP-Adresse vergeben. Dies ist für alle Netzwerkgeräte von Vorteil, deren Dienste von anderen genutzt werden, wie z. B. Printserver.

Dynamische Zuordnung: Ein anfragender Client bekommt eine beliebige Adresse aus einem festgelegten Adressbereich für eine bestimmte Zeit zugewiesen. Die Zuweisung selbst nennt man *Lease*, die Zeitspanne der Gültigkeit *Leasetime*.

Im Vorgriff auf Kapitel 5, »Steuer- und Fehlercodes mit ICMP und ICMPv6 übertragen«, und Kapitel 6, »Datentransport mit TCP und UDP«, sehen Sie in Tabelle 4.5 speziell den Aufbau eines DHCP-Pakets dargestellt.

Byte	Feld
1	Anforderung (1) Antwort (2) (op)
1	Netzwerktyp (1 für Ethernet) (htype)
1	Länge Hardware-Adresse (6 Byte bei MAC-Adresse) (hlen)
1	Zahl der DHCP-Relay-Agenten (hops)
4	Transaktions-ID (xid)
2	abgelaufene Zeit seit Client-Start in Sekunden (secs)
2	Gültigkeit der IP-Adresse beim Client, Byte 2 reserviert (flags)
4	IP-Adresse des Clients, falls schon vorhanden (alte Adresse) (ciaddr)
4	zugeteilte IP-Adresse (yiaddr)
4	IP-Adresse des DHCP-Servers (siaddr)
4	IP-Adresse des DHCP-Relays (Router) (giaddr)
16	Client-MAC-Adresse (chaddr)
64	Host-Name des DHCP-Servers, Angabe optional (sname)
128	Boot-Image (Dateiname, wenn TFTPboot verwendet wird) (file)
312	Optionen (options)

Tabelle 4.5 Aufbau eines DHCP-Pakets

DHCP-Server finden Sie auch in DSL- und WLAN-Routern. Hier steht Ihnen eine komfortable Weboberfläche für Ihre Einträge zur Verfügung (Abbildung 4.11). Mit wenigen Angaben erhalten Sie für ein kleines Netz den DHCP-Komfort.

LAN IP Setup

☐ Enable UPnP

LAN TCP/IP Setup

IP Address

192

168

0

1

IP Subnet Mask

255

255

255

0

RIP Direction

None

RIP Version

Disabled

MTU Size

☐ Default (1500)
☒ Custom

1492

LAN IP Address Management

☐ Disable DHCP Server
☐ Use External DHCP Server
☒ Use Router as DHCP Server

DHCP Server Info

Starting IP Address

192

168

0

102

Ending IP Address

192

168

0

110

DNS Server Address
☒ Router is DNS Proxy
☐ Use These DNS Servers

Primary DNS

Secondary DNS

WINS Server

Lease Time (hours)

24

Reserved IP Table

	#	IP Address	Mac Address	Device Name

Abbildung 4.11 Webkonfigurationsoberfläche mit DHCP-Einrichtung eines DSL-Routers

Die Konfiguration eines DHCP-Servers finden Sie in der Datei *dhcpd.conf*. Bei Linux und FreeBSD finden Sie die Datei im Verzeichnis */etc* bzw. */usr/local/etc*. Die wichtigsten Punkte finden Sie hier mit Kommentaren (#) in der Muster-Konfiguration dargestellt:

Muster-Konfiguration der »dhcpd.conf«

```
# Gültigkeit der Zuweisung (Standard- und Maximalwert)
# in Sekunden
default-lease-time 600;
max-lease-time 7200;

# Der Server ist für das lokale Netz der alleinig
# zuständige:
authoritative;
# Angaben der Subnetzmaske, der Broadcast-Adresse, der
# Router, des Nameservers und des Domain-Namens
option subnet-mask 255.255.255.0;
option broadcast-address 192.168.5.255;
option domain-name-servers 192.168.5.1;
option domain-name "DOMAINNAME.TLD";
# Dynamisch zu vergebender Adressbereich
subnet 192.168.5.0 netmask 255.255.255.0 {
    range 192.168.5.50 192.168.5.99;
}
# Feste Adresszuweisung für MAC-Adresse
# Die Adresse liegt außerhalb des Bereichs
# dynamisch zu vergebender Adressen
host schulung1 {
```

```

hardware ethernet 01:02:03:1a:1b:1c;
fixed-address 192.168.5.100;
}

```

Sie finden weitere Optionen in der Dokumentation, die Sie unter Linux oder FreeBSD mit `man dhcpd.conf` abrufen können. Windows-Benutzer lesen die dem Programm beiliegende Information. Falls Sie eine Datenbanklösung für Ihre Geräteverwaltung erstellen, binden Sie die *dhcpd.conf* gleich mit ein. Ihr Skript sollte den Kopf dieser Datei inklusive der subnet-Anweisung aus einer eigenen Datei mit der Datenbankabfrage zusammenführen, die die festen Zuweisungen ausgibt. Das ist unkomplizierter zu programmieren als die Lösungen, die explizit Zeileneinträge ändern. Nach jeder Neuerstellung der *dhcpd.conf*-Datei sollten Sie (oder das Skript) den DHCP-Server neu starten. Für die Fehlersuche hilft Ihnen ein Packet-Sniffer wie `tcpdump`. Die (für das Buch kommentierte Ausgabe) finden Sie in Listing 4.1. Es zeigt eine funktionierende Adresszuweisung:

```

# Adressanforderung durch den Client
IP (tos 0x10, ttl 128, id 0, offset 0, flags [none],
  proto UDP (17),
  length 328) 0.0.0.0.68 > 255.255.255.255.67: BOOTP/DHCP,
Request from 00:18:e7:16:71:30,
  length 300, xid 0x1121db06, Flags [none]
Client-Ethernet-Address 00:18:e7:16:71:30 [|bootp]
# Antwort vom DHCP-Server
IP (tos 0x0, ttl 32,
  id 33152, offset 0,
  flags [DF],
  proto UDP (17),
  length 576) 192.168.0.1.67 > 192.168.0.103.68: BOOTP/DHCP,
  Reply, length 548, xid 0x1121db06, Flags [none]
Your-IP 192.168.0.103
Client-Ethernet-Address 00:18:e7:16:71:30
sname "M-^?" [|bootp]

```

Listing 4.1 Ausgabe von »tcpdump«

Speziell für die Fehlersuche können Sie `dhcpdump` benutzen. Dessen Ausgabe ist ausführlicher (Listing 4.2). Auch hier sehen Sie eine funktionierende Zuweisung:

```

ZE4:~# dhcpdump -i eth1
TIME: 2016-04-24 22:59:13.002
IP: 0.0.0.0 (0:18:e7:16:71:30) > 255.255.255.255 (ff:ff:ff:ff:ff:ff)
OP: 1 (BOOTPREQUEST)
HTYPE: 1 (Ethernet)
HLEN: 6
HOPS: 0
XID: 3bfee54c
SECS: 0
FLAGS: 0
CIADDR: 0.0.0.0
YIADDR: 0.0.0.0
SIADDR: 0.0.0.0
GIADDR: 0.0.0.0
CHADDR: 00:18:e7:16:71:30:00:00:00:00:00:00:00:00:00:00:00:00
SNAME: .
FNAME: .
OPTION: 53 ( 1) DHCP message type 3 (DHCPREQUEST)
OPTION: 50 ( 4) Request IP address 192.168.0.103
OPTION: 55 ( 12) Parameter Request List 1 (Subnet mask)
28 (Broadcast address)
2 (Time offset)

3 (Routers)
15 (Domainname)
6 (DNS server)
119 (Domain Search)
12 (Host name)
44 (NetBIOS name server)
47 (NetBIOS scope)

```

```

26 (Interface MTU)
121 (Classless Static Route)
-----
# Nachfolgend Antwort vom Server
-----
TIME: 2016-04-24 22:59:13.004
IP: 192.168.0.1 (0:18:4d:57:eb:a8) > 192.168.0.103 (0:18:e7:16:71:30)
OP: 2 (BOOTPREPLY)
HTYPE: 1 (Ethernet)
HLEN: 6
HOPS: 0
XID: 3bfee54c
SECS: 0
FLAGS: 0
CIADDR: 0.0.0.0
YIADDR: 192.168.0.103
SIADDR: 0.0.0.0
GIADDR: 0.0.0.0
CHADDR: 00:18:e7:16:71:30:00:00:00:00:00:00:00:00:00:00
SNAME: .
FNAME: .
OPTION: 53 ( 1) DHCP message type 5 (DHCPACK)
OPTION: 54 ( 4) Server identifier 192.168.0.1
OPTION: 1 ( 4) Subnet mask 255.255.255.0
OPTION: 51 ( 4) IP address leasetime 86400 (24h)
OPTION:
52 ( 1) Option overload 3 (file and sname field holds options)
OPTION: 28 ( 4) Broadcast address 255.255.255.255
OPTION: 3 ( 4) Routers 192.168.0.1
OPTION: 6 ( 4) DNS server 192.168.0.1
OPTION: 26 ( 2) Interface MTU 576

```

Listing 4.2 Ausgabe von »dhcpcdump«

Wenn Sie ein lokales Netz mittels DHCP verwalten, sollten Sie sich über aktuelle Sicherheitsprobleme auf dem Laufenden halten. Als zentraler Dienst ist DHCP auch Ziel von Schadsoftware, die die Störung oder Ausforschung Ihres Netzwerkverkehrs bezweckt.

4.2.4 Zeroconf

Für *Zeroconf* finden Sie keine einheitliche Einbindung in die verschiedenen Betriebssysteme:

Zeroconf-Implementierungen

- ▶ **Microsoft Windows:** Ist der Windows-Rechner für den Adressbezug über DHCP eingerichtet, versucht er mehrmals einen DHCP-Server zu erreichen. Schlägt dies fehl, weist er sich eine IP-Adresse aus dem für Zeroconf reservierten Bereich zu (*APIPA*).
- ▶ **Apple:** Von Apple wird die Implementierung *bonjour* genannt.
- ▶ **Linux, FreeBSD:** Hier finden Sie Zeroconf unter dem Namen *avahi*. Viele Distributionen installieren den dazugehörigen Daemon mit. Dessen Konfigurationsdatei *avahi-daemon.conf* finden Sie unter */etc/avahi*. Meist genügen die Voreinstellungen für den Betrieb. Genaue Ausführungen zu den Einträgen dieser Datei erhalten Sie mit `man avahi-daemon.conf`.

In der täglichen Netzwerkpraxis begegnet Ihnen Zeroconf, wenn etwas nicht funktioniert. Außer bei Apple-Systemen wird es meist als Notfalloption eingerichtet. Wenn Sie einen PC für DHCP-Adressbezug konfigurieren und eine Adresse aus dem Zeroconf-Bereich vorfinden, ist entweder der DHCP-Server nicht gestartet oder die Verbindung zu ihm ist gestört.

In der täglichen Arbeit können Sie Zeroconf nutzen, um ein Mini-Netzwerk mit zwei Rechnern über ein Crosskabel schnell in Betrieb zu nehmen. Aber auch hier sind Sie

manchmal mit festen IP-Adressen schneller ...

4.3 Verwendung von Rechnernamen

Bei Ihrer täglichen Arbeit sprechen Sie einen Rechner selten direkt mit der IP-Adresse an, sondern über seinen Namen. Sie tippen nur noch bei Konfigurationsarbeiten im Lokale Netzwerk IP-Adressen in den Webbrowser ein. Ihr Rechner oder ein Nameserver löst mehr oder weniger unauffällig im Hintergrund die Namen in die IP-Adressen auf.

4.3.1 Der Urtyp: Adressauflösung in der »hosts«-Datei

Bevor es den Nameserver gab, pflegten die Systembetreuer eine mehr oder weniger lange *hosts-Datei*. Sie finden darin eine Tabelle, die Namen und IP-Adressen gegenüberstellt

Sie finden die Datei bei Windows 7-Rechnern unter

```
%SystemRoot%\system32\drivers\etc\
```

und bei Linux und FreeBSD unter

```
/etc
```

abgelegt.

Für jede IP-Adresse dürfen Sie in die gleiche Zeile mehrere Namen eintragen. Meist finden Sie hier einen kompletten Namen (Name mit Domain und Top-Level-Domain) und anschließend den lokalen Kurznamen. Die Datei wird von oben nach unten gelesen. Beim ersten passenden Eintrag wird die dort gefundene IP-Adresse verwendet.

Die Einrichtung und Pflege einer lokalen *hosts*-Datei hat vor allem den Nachteil, dass Sie selbst die Einträge verwalten müssen. Die Verwendung dieser Datei bringt Ihnen aber auch große Vorteile:

Rechner fragen stets zuerst die lokale *hosts*-Datei vor einem Nameserver-Zugriff ab. Wenn Sie hier oft benötigte Adressen hinterlegen, können Sie auch bei einer Störung des DNS-Servers wie gewohnt mit den Namen weiterarbeiten.

Zusammen mit fest vergebenen lokalen Adressen erhalten Sie ein recht robustes, gegen Störungen und Angriffe gewappnetes Netz, da keine aktiven Dienste (DHCP, DNS) benötigt werden.

Testen von Webserveranwendungen: Leiten Sie auf den Localhost (127.0.0.1) um. Sie müssen später beim Echteinsatz am Code nichts mehr ändern.

Setzen privater »Websperren«. Durch die Umleitung auf die 127.0.0.1 kommen ausforschende Webseiten, Trojaner und Viren(-schleudern) nicht mehr zum Zug. Aber: Sollten Sie diese Einstellung auf den Arbeitsplätzen Ihrer Arbeitnehmer (oder Kollegen) vornehmen, achten Sie auf die aktuell gültige Rechtslage und auf Betriebsvereinbarungen! Listen mit obskuren Seiten schon im *hosts*-Format erhalten Sie im Internet.

In der Muster-*hosts*-Datei finden Sie lokale und nicht lokale Adressen sowie auf den Localhost umgeleitete Rechnernamen. Sie können daraus Ihr eigenes Muster ableiten, mit dem Sie Ihre lokalen Rechner ausstatten.

Muster für »/etc/hosts«**# Definition localhost, zwingend**

```
127.0.0.1 localhost
::1 localhost ip6-localhost ip6-loopback
```

```
fe00::0 ip6-localnet
ff00::0 ip6-mcastprefix
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters
ff02::3 ip6-allhosts
```

Rechner lokal und Internet

```
192.168.0.34 ze4.example.com ze4
192.168.0.25 zentrale.example.com zentrale
192.168.0.54 lp
fd91:e4a5:5e03::dd05 ze4-6.example.com ze4-6
fd91:e4a5:5e03::dd54 lp-6
fd91:e4a5:5e03::dd04 zentrale-6 .example.com zentrale-6
192.0.32.8 www.iana.org ia
```

Bewirkt Sperre nach außen

```
127.0.0.1 www.facebook.de
127.0.0.1 www.google-analytics.com
127.0.0.1 googleads.g.doubleclick.net
127.0.0.1 pagead.googleadsyndication.com
```

4.3.2 Der Domain Name Server (DNS) und seine Konfiguration

Für ein umfangreicheres Netz können Sie einen Nameserver (DNS) verwenden. Mit diesem schaffen Sie sich einen zentralen Konfigurationspunkt. Im Zusammenhang mit einem DHCP-Server und einer Datenbank für die Geräteverwaltung automatisieren Sie viele einzelne Arbeitsschritte.

Für Privatanutzer und kleine Firmennetze, die die Namen ihrer wenigen Rechner in der *hosts*-Datei auflösen, reicht die Weiterleitung der DNS-Anfragen zum Internet-Provider über den DSL-Router im Normalfall aus.

Sehen Sie sich die Zusammenfassung der Vor- und Nachteile einmal an:

Zentraler Konfigurationspunkt; Clients müssen nur einmal für den DNS-Zugriff eingestellt werden.

Ein eigener Nameserver verhindert, dass Sie bei einer erfolglosen Anfrage auf einer »Meinten Sie ...«-Seite des Internet-Providers landen.

Beim Ausfall des DNS-Servers des Internet-Providers können Sie trotzdem Anfragen an übergeordnete Nameserver stellen.

Aber: Der Nameserver ist ein beliebtes Ziel für Angriffe und Ausspähungen. Sie müssen das Programm deshalb stets in der aktuellen Version betreiben.

Anfall zusätzlichen Netzwerkverkehrs

Zunächst lesen Sie hier in Kurzform alles Wichtige:

Domain Name Service (DNS)

- ▶ Zusammenführung von IP-Adresse und Rechnername
- ▶ weltweit und lokal
- ▶ Suche mit Host-Namen oder IP-Adresse möglich

Zusatznutzen: Unterdrückung von Spam-Mails (unerwünschte Werbe-Mails) durch umgekehrte Namensauflösung

- ▶ Setzen von Websperren und -umleitungen (analog zur *hosts*-Datei)

Belegte Ports: UDP/TCP 53

RFCs zum Nachlesen: 1034 und 1035 (Updates in RFCs 1101, 1183, 1348, 1876, 1982, 1995, 1996, 2065, 2136, 2137, 2181, 2308, 2535, 2673, 2845, 3425, 3658, 4033, 4034, 4035, 4343, 4592, 5936, 5966, 6604 und 7766)

Aufbau des Namensraums: hierarchisch, Leseweise von rechts nach links, beginnend mit dem Zeichen ».«

Programm: Hauptsächlich wird *BIND* verwendet, Programmname: *named*.

Konfigurationsdateien: *named.conf*, *named.conf.options* und die Zonendateien *<zone>.db*

Weitere Begriffe:

Top-Level-Domain: zeigt Landeszugehörigkeit oder Verwendung an

Domain-Name: registrierter Eintrag, z. B. Firmenname

Subdomain: vom Domain-Inhaber vergebene Unterdomain

Fully Qualified Domain Name (FQDN): Bestandteil einer URL (URL ohne Angabe des Dienstes)

Beachten Sie den Aufbau des Namensraums. Sie geben in Ihren Webbrowser *www. example. org* ein. Ihr PC und die beteiligten Nameserver werten diese Eingabe umgekehrt aus: *. org. example. www*. Der Punkt steht als oberste Instanz der baumartigen Struktur (Wurzel- oder Root-Verzeichnis). Anschließend folgt die *Top-Level-Domain (org)*. Diese trifft eine Aussage über die Landeszugehörigkeit (*de, at, dk, ch ...*) oder deren Verwendung (*com* für kommerzielle Nutzung, *org* für Organisationen, *gov* für die US-Regierungsstellen). In letzter Zeit wurde der Bereich der Top-Level-Domains (TLD) um zusätzliche Möglichkeiten erweitert. Unter *www. iana. org/domains/root/db/* finden Sie die jeweils aktuell gültigen TLDs.

Unterhalb der TLD finden Sie die Second-Level-Domain. Sie wird hierzulande oft nur als *Domain-Name* bezeichnet, hier lautet sie *example*. Diesen Namen können Sie beim jeweils zuständigen Registrar (über Ihren Internetdienstleister) beantragen. Als Domain-Namen dürfen Sie alles verwenden, was nicht gegen Recht und Gesetz (Markenschutz) oder die »guten Sitten« verstößt. Das gezeigte Beispiel mit *example. org* ist für Dokumentations- und Anleitungszwecke nach RFC 6761 für die *IANA* reserviert. Abweichend davon können Sie bei bestimmten TLDs auf länderspezifische Ausnahmen stoßen. In einigen Ländern werden nur die Third-Level-Domains vergeben (z. B. in Großbritannien *example. co. uk*). Betrachten Sie hierzu das Schema in Abbildung 4.12.

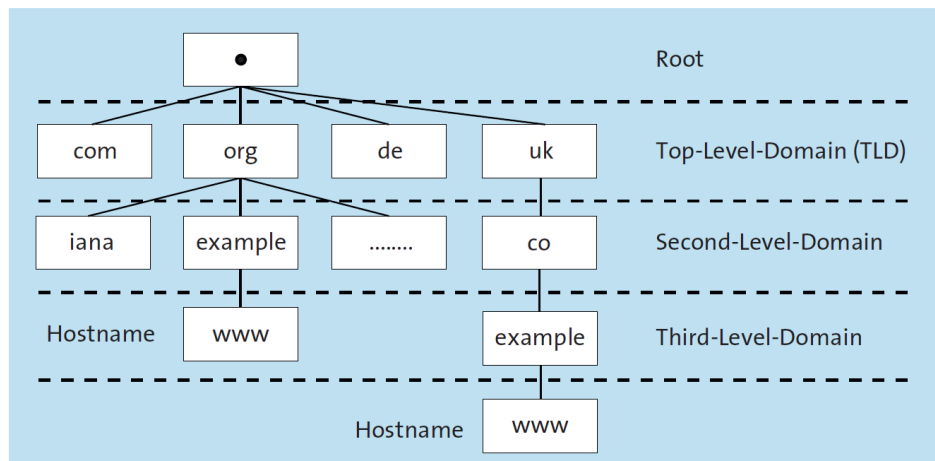


Abbildung 4.12 Aufbau des Namensraums mit Host-Namen

Unterhalb der Ihnen zugewiesenen Domain können Sie selbst Subdomains (auch in weiteren Ebenen) vergeben und verwalten. Im Regelfall werden Sie aber sofort den Host-Namen angeben, der erreicht werden soll (hier *www*). Wenn Sie eine Unterdomain (Subdomain) benutzen, kann die URL z. B. *www.europe.example.org* lauten. Die Subdomain, die Sie selbst in eigener Zuständigkeit verwalten, lautet *europe*.

Die Richtung der Suchabfragen hält sich ebenfalls an diese Hierarchie. Damit erkennen Sie auch, dass, je nach Ebene, die beteiligten Nameserver einen unterschiedlichen Status haben.

Nameserver

- ▶ **Autoritativ:** Der DNS-Server ist für eine Zone zuständig, und die von ihm darüber ausgegebenen Informationen gelten als »sicher«.
- ▶ **Primary Nameserver:** der autoritative Server einer Zone
- ▶ **Secondary Nameserver:** zusätzlicher Nameserver für eine Zone, der die gleichen Informationen wie der autoritative Server über eine Zone liefert. Die Zonendaten werden mittels Zonentransfer und nicht durch Abfrageweiterleitungen untereinander übermittelt.
- ▶ **Nicht-autoritativ:** Der DNS-Server bezieht seine Abfrageergebnisse von anderen Nameservern. Die Ergebnisse zählen als »nicht sicher«.
- ▶ **Caching-only-Nameserver:** Dieser Nameserver verwaltet selbst keine Zonen, sondern leitet Ihre Anfragen an andere Nameserver weiter. Angewendet wird dies häufig DSL-Routern.

Sie werden einen Nameserver häufig so konfigurieren, dass er eine Domain, gegebenenfalls mit Subdomains, verwaltet. Für die Subdomains selbst können Sie weitere eigene Nameserver aufsetzen. Der für diese Domain autoritative DNS-Server greift dann für die Anfragen bezüglich der Subdomains auf diese zu. Für Anfragen außerhalb der eigenen Domain leitet Ihr lokaler Nameserver die Anfragen an einen übergeordneten DNS-Server weiter (Forwarding). Wenn Ihre Anfrage von diesem nicht erledigt werden kann, gelangt sie zu einem der sogenannten *Root-Server*. Ein Root-Server teilt dem übergeordneten DNS-Server mit, welche Nameserver die gewünschten Informationen haben (iterative Antwort). Geschieht dies nicht, erhalten Sie die Fehlermeldung, dass der angefragte Host nicht existiert.

Sie müssen vor der Einrichtung eines Nameservers abklären, unter welchem Betriebssystem er laufen soll. Er muss so abgeschottet wie nur möglich vom restlichen System laufen, da trotz sorgfältiger Programmierarbeit immer wieder Sicherheitsprobleme auftreten. Unter Linux oder FreeBSD sollte er wenigstens in einer *chroot*-Umgebung laufen. Die Konfigurationsdateien finden Sie unter */etc/namedb* (bei FreeBSD) oder */etc/bind* (bei Debian-Linux), gegebenenfalls als Symlink für das *chroot*-Verzeichnis. Wenn

Sie den Nameserver *bind* bei Debian oder FreeBSD über die systemeigenen Mechanismen installieren, wird die *chroot*-Umgebung mit eingerichtet; Sie brauchen weiter nichts dafür zu tun. Wenn Sie Ihren DNS-Server konfigurieren, haben Sie zwei getrennte Arbeitsgänge vor sich. Zunächst nehmen Sie für das Serverprogramm selbst die Einstellungen vor. Sie geben unter anderem an, unter welcher IP-Adresse und gegebenenfalls auf welchem Port der Nameserver aktiv ist.

Die Vorgabe für das Konfigurationsbeispiel lautet: IP-Adressen 127.0.0.1 und ::1, IPv6 *aktiv*. Sie sehen nur die zu ändernden Zeilen. Sie können an diesen Stellen die IP-Adresse des Rechners setzen, falls er seine Dienste weiteren Rechnern im LAN anbieten soll. Dies nehmen Sie am besten nach der kompletten Konfiguration und einem anschließenden Test vor. Bis dahin prüfen Sie ausschließlich lokal. Am Schluss des Auszugs der *named.conf* finden Sie Einträge zu den *forwarders*. Dies ist eine Anweisung, in welcher Reihenfolge die Nameserver kontaktiert werden. Hier finden Sie folgende Reihenfolge: den Server selbst, den Router (Caching-only-Nameserver) sowie den Primary und Secondary Nameserver des Providers. Sie gewinnen damit meist ein besseres Antwortzeitverhalten und verursachen weniger Netzlast bezüglich der Root-Serveranfragen.

»/etc/namedb/named.conf« auf FreeBSD-Server (Auszug)

```
// If named is being used only as a local resolver,
// this is a safe default.
// For named to be accessible to the network, comment
// this option, specify
// the proper IP address, or delete this option.
    listen-on      { 127.0.0.1; };
// If you have IPv6 enabled on this system, uncomment
// this option for
// use as a local resolver. To give access to the
// network, specify
// an IPv6 address, or the keyword "any".
    listen-on-v6   { ::1; };
.....
// If you've got a DNS server around at your upstream
// provider, enter
// its IP address here, and enable the line below. This
// will make you
// benefit from its cache, thus reduce overall DNS traffic
// in the Internet.

    forwarders {
        127.0.0.1;
        192.168.0.1;
        217.237.151.97;
        134.100.33.240;
    };
```

Sie haben damit das Programm selbst schon konfiguriert. Wenn Sie den Nameserver jetzt starten, steht Ihnen dieser lokale Nameserver bereits für das Forwarding zur Verfügung. Prüfen Sie die Konfiguration mit *named-checkconf* auf ihre formelle Richtigkeit hin:

```
zentrale# named-checkconf -v named.conf
9.6.-ESV-R3
```

Mit der Option *-v* erhalten Sie die Versionsnummer. Nur im Fehlerfall bekommen Sie mehr zu sehen! Stellen Sie jetzt auch zumindest für die Tests die Datei */etc/resolv.conf* auf die lokale Adresse um:

»/etc/resolv.conf«

```
domain example.com # Durch die eigene ersetzen!
nameserver 127.0.0.1
nameserver ::1
```

Die Nameserver-Abfragen auf diesem System werden damit auf diese Adressen umgeleitet. Konfigurieren Sie jetzt die Zonendateien. Für jede Zone werden zwei dieser Dateien angelegt: eine für die Vorwärtssuche (nach Namen) und eine für die Rückwärtssuche (nach IP-Adressen). In diesen Zonendateien finden Sie die Einträge in sogenannten *Resource-Records* abgelegt. Den SOA-Record (*Start of Authority*) müssen Sie am Anfang jeder Zonendatei anlegen. Beachten Sie hierzu auch das RFC **1035** und seine Updates RFC 1101, 1183, 1348, 1876, 1982, 1995, 1996, 2065, 2136, 2137, 2181, 2308, 2535, 2673, 2845, 3425, 3658, 4033, 4034, 4035, 4343, 5936, 5966, 6604 und 7766. Im SOA-Record legen Sie (in dieser Reihenfolge) wichtige Kenndaten der Zone fest:

Aufbau des SOA-Records

- ▶ **Zonename:** kann durch das Zeichen @ ersetzt werden.
- ▶ **Zonenklasse:** beim IP-Protokoll vorgegebener Wert: IN
- ▶ **Record-Type:** hier SOA
- ▶ **Primary Master:** primärer Nameserver
- ▶ **E-Mail-Adresse des Zonenbetreuers:** Beachten Sie, dass das @-Zeichen durch einen Punkt ersetzt werden muss, weil es in dieser Datei eine andere Verwendung findet.
- ▶ **Seriennummer:** Nach jeder Änderung an der Zonendatei müssen Sie diesen Wert um 1 erhöhen. Die Seriennummer wird in der Praxis in der Form JJJJMMTT1fd angegeben, z. B. für die erste Änderung am 20. Mai 2016: 2016052101. Lesen Sie hierzu im RFC 1912 nach.

- ▶ **Refresh-Zeit:** In diesen Intervallen kontrollieren eventuell vorhandene Slave-Nameserver, ob seit der letzten Anfrage Änderungen eingetreten sind. Vorgabewert: 86400 Sekunden (1 Tag).
- ▶ **Retry-Zeit:** War Ihr Nameserver nicht erreichbar, versuchen eventuell vorhandene Slave-Nameserver nach Ablauf der vorgegebenen Zeit (in Sekunden) eine erneute Anfrage. Gebräuchlicher Wert: 7200 Sekunden (2 Stunden).
- ▶ **Expire Time:** Slave-Nameserver holen immer die gesamte Zonendatei vom (Primary) Master ab. Funktioniert das nicht, wird nach Ablauf der Zeit beim Slave-DNS-Server die betroffene Zone deaktiviert. Gebräuchlicher Wert: 3600000 Sekunden (1000 Stunden).
- ▶ **TTL:** Vorhaltezeit von Anfragen mit negativer Antwort. Sie können hier wenigstens 600 Sekunden (5 Minuten), maximal nach einer RIPE-Empfehlung 172800 Sekunden (2 Tage) einstellen.

Lesen Sie im RIPE-Dokument ripe-203 unter <http://www.ripe.net/ripe/docs/ripe-203> sowie in RFC **2308** mit den Updates RFC 4033, 4034, 4035 und 6604 nach.

Sie können den SOA-Record in eine einzige Zeile schreiben. Benutzen Sie aber für eine bessere Lesbarkeit die mehrzeilige Schreibweise.

Die weiteren Record-Typen einer Zonendatei benutzen Sie für die Beschreibung einzelner Hosts:

Weitere Record-Typen einer Zonendatei

- ▶ **NS:** Nameserver
- ▶ **MX:** Mail-Server (Zonendatei)
- ▶ **A:** IPv4-Host (Zonendatei)
- ▶ **AAAA:** IPv6-Host (Zonendatei)
- ▶ **CNAME:** Verweis auf weiteren Host-Namen (Zonendatei)
- ▶ **PTR:** »Zeiger« (Reverse-Zonendatei)

Weitere Informationen finden Sie in RFC 2317.

Innerhalb von Zonendateien können Sie auch Werte setzen, die für alle Einträge gültig sind:

Allgemeingültige Werte belegen (Zonendateien)

- ▶ **IN:** »Internetprotokoll« Setzen Sie bereits im SOA-Record, Sie müssen diese Angabe deshalb in den Host-Zeilen nicht wiederholen.
- ▶ **\$TTL:** Setzen Sie noch vor dem SOA-Record diese Angabe. Sie legen damit fest, wie lange die Information in einem Cache gehalten wird. Damit brauchen Sie diesen Wert nicht in den Host-Zeilen zu setzen.
- ▶ **\$ORIGIN:** Damit können Sie manuell (weitere) Zonnennamen (Origins) festlegen.

Eine Host-Zeile hat folgenden Aufbau:

Hostname TTL Zonenklasse Recordtyp IP-Adresse

Wenn Sie allgemeingültige Werte festlegen, verkürzen Sie diese Zeilen:

Hostname (ohne Zone) Recordtyp IP-Adresse

Für die Reverse-Zonendateien gilt:

IP-Adresse Zonenklasse Recordtyp Hostname

In den Reverse-Zonendateien werden die IP-Adressen in umgekehrter Reihenfolge eingetragen. Bei IPv6-Adressen können Sie damit auch nicht mehr die üblichen Schreibweisen benutzen, sondern müssen Stelle für Stelle rückwärts eintragen.

Bezeichnen Sie eine Datei für die Vorwärtssuche (nach Name) in der Form *<ZONE>-forward.db*, für die Rückwärtssuche (nach IP-Adresse) *<ZONE>-reverse.db*. Sie finden auch Bezeichnungen wie *db.local* für die Vorwärtssuche und dazugehörend *db.0*, *db.127* und *db.255* für die Rückwärtssuche, z. B. auf Debian-Systemen. In allen Zonendateien müssen Sie den Host-Namen mit dem Punktzeichen ».« abschließen.

Betrachten Sie drei Beispiele aus der Praxis:

- *netzbuch-forward.db*: Suche nach IP-Adressen, IPv4 + IPv6
- *netzbuch-reverse.db*: Suche nach Host-Namen, Eingabe der IPv4-Adresse
- *netzbuch-reverse6.db*: Suche nach Host-Namen, Eingabe der IPv6-Adresse

Mit den Einträgen in *netzbuch-forward.db* sucht der Nameserver die IP-Adressen aufgrund der Eingabe des Host-Namens.

»netzbuch-forward.db«

;; Cache-Haltezeit, gültig für alle Einträge

\$TTL 1800

;; SOA-Record

```
@      IN  SOA  localhost. harald.zislers.de. (
                                201105210  ; Seriennummer
                                86400      ; Refresh Time
                                7200       ; Retry Time
                                3600000    ; Expire Time
                                600        ; negative Caching Zeit
                                )
```

;; Angaben zu Hosts und IP-Adressen

```
                                NS      localhost.

ze4      A      192.168.0.34
          AAAA   fd91:e4a5:5e03::dd05
zentrale A      192.168.0.25
          AAAA   fd91:e4a5:5e03::dd04
mobarb   A      192.168.0.41
          AAAA   fd91:e4a5:5e03::dd06

lp        A      192.168.0.54
          AAAA   fd91:e4a5:5e03::dd54

;; Jails am FreeBSD-Host
daten1    A      192.168.0.65
daten2    A      192.168.0.66
netzbuch  A      192.168.0.67
```

Für die Reverse-Suche benutzt der Nameserver die beiden Dateien *netzbuch-reverse.db* (für die Abfrage mit IPv4-Adressen) und *netzbuch-reverse6.db* (Abfrage mit IPv6-Adressen).

»netzbuch-reverse.db«

\$TTL 1800

```
@      IN  SOA  localhost. harald.zislers.de. (
                                201105210  ; Seriennummer
                                86400      ; Refresh Time
                                7200       ; Retry Time
                                3600000    ; Expire Time
                                600        ; negative Caching Zeit
                                )
```

NS localhost.

»named.conf«: Eintrag der eigenen Zonendateien

```
// Eigene Netzwerkzonen
zone "netzbuch" { type master; file
    "/etc/namedb/master/netzbuch-forward.db"; };
zone "0.168.192.in-addr.arpa" { type master; file
    "/etc/namedb/master/netzbuch-reverse.db"; };
zone "0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.3.0.e.5.5.a.4.e.1.9.d.f.ip6.arpa"
    { type master; file "/etc/namedb/master/netzbuch-reverse6.db"; };
```

Sie können jetzt an dieser Stelle die *named.conf* um die IP-Adressen ergänzen, auf denen der Nameserver seine Anfragen entgegennimmt (Zeilen mit `listen-on` und `listen-on-v6`, hier stehen bereits die Adressen `127.0.0.1` und `:::1`).

Damit haben Sie Ihren Nameserver konfiguriert. Starten Sie ihn neu, am besten über das RC-Skript (Linux und FreeBSD) oder das Dienste-Menü (Windows).

4.3.3 Einstellungen beim Client

Sie müssen Ihre PCs und weitere Netzwerkgeräte so einstellen, dass sie auf den von Ihnen gewünschten Nameserver zugreifen.

Wenn Sie in Ihrem Netz einen DHCP-Server betreiben, können Sie die Information über den Nameserver gleich bei der Zuteilung der IP-Adresse dem Client mit übergeben (Eintrag in *dhcp.conf*, siehe Kasten »Muster-Konfiguration der *dhcpcd.conf*« in Abschnitt 4.2.3). Sie bekommen auch von Ihrem Internetdienstleister diese Information übermittelt, wenn Sie sich einwählen oder verbinden und den automatischen Bezug von IP-Adresse und Nameserver-Angaben im Client so eingestellt haben. In allen anderen Fällen setzen Sie die Angaben hierzu stets manuell.

Die Einstellung bei Windows-Clients für den automatischen Bezug von IP-Adresse und Nameserver-Einstellungen finden Sie in Abbildung 4.13. Ähnlich sehen die Masken bei vielen Linux-Derivaten aus.

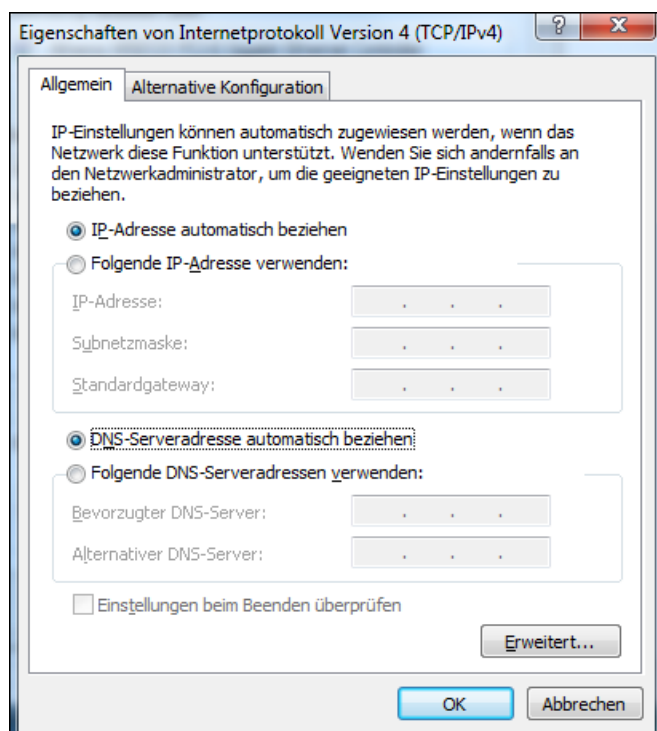


Abbildung 4.13 Einstellungen für automatischen Adressbezug und Nameserver-Eintrag

Unter Linux und FreeBSD können Sie Feineinstellungen für die Namensauflösung vornehmen. Hier im Beispiel finden Sie die Anweisung, dass zunächst auf die lokale *hosts*-Datei, anschließend auf den Nameserver zugegriffen werden soll:

Einträge in */etc/nsswitch.conf*. Hier bestimmen Sie die abzufragenden Ressourcen.

»etc/nsswitch.conf«

```
.....
hosts: files dns
.....
```

Einträge in */etc/host.conf*. Legen Sie die Reihenfolge der Abfrage von */etc/hosts* und dem Nameserver fest. Die letzte Zeile regelt den Umgang mit Hosts, die mehrere IP-Adressen haben. Mit *on* lassen Sie Ergebnisse zu, bei denen für einen Host-Namen mehrere IP-Adressen möglich sind. Bei *off* würden Sie eine Fehlermeldung erhalten.

»/etc/host.conf«

```
order hosts dns
multi on
```

Einträge in */etc/resolv.conf*. Hier geben Sie zunächst an, wie unvollständige Host-Namen vervollständigt werden. Die Angabe *domain* können Sie nur mit einem Namen belegen, *search* mit mehreren. Die Angabe von *domain* hat Vorrang vor *search*. Anschließend geben Sie die Nameserver an, auf die Ihr Rechner zugreifen soll. Bis zu drei dieser Nameserver dürfen Sie hier eintragen.

Aufbau »resolv.conf«

```
domain example.com
search example.com, example.org
nameserver 127.0.0.1
nameserver 217.237.151.97
```

Normalerweise arbeitet die Namensabfrage im Hintergrund, ohne dass es dem Benutzer auffällt. Bei größeren Um- und Neukonfigurationen von Netzen oder auch bei Angriffen auf Nameserver bricht scheinbar das Netz zusammen. Auch hier können Sie dem Arbeitsstillstand vorbeugen, indem Sie wichtige Hosts, auf die Sie bei Ihrer Arbeit angewiesen sind, in der Datei *hosts* hinterlegen.

4.4 Überprüfung der Erreichbarkeit und Namensauflösung von Hosts

Bei der Überprüfung der Erreichbarkeit eines Hosts haben Sie es oft auch gleichzeitig mit Fehlern bei der Namensauflösung zu tun. Aus diesem Grund habe ich hier beide Prüfungsfelder zusammengefasst.

4.4.1 Prüfung der Erreichbarkeit und Namensauflösung mit »ping« bzw. »ping6«

Ihr klassisches Werkzeug für die Verbindungsprüfung ist *ping* (IPv4) bzw. *ping6* (IPv6). Beide nutzen das in Kapitel 5, »Steuer- und Fehlercodes mit ICMP und ICMPv6 übertragen«, erklärte ICMP-Protokoll. In Tabelle 4.6 finden Sie ausgewählte Optionen der beiden Befehle.

Aktion	ping	ping6
Ausgabe eines Pieptons für jeden Ping	-a	-t
Begrenzen auf NN Pings	-c <NN>	-c <NN>
Zeit zwischen den Pings	-i <sec>	-i <sec>
Ausführliche Ausgabe	-v	-v

Tabelle 4.6 Optionen von »ping« und »ping6«

Der ping-Befehl hat die Grundform

```
ping -<OPTION> <IP-Adresse>
```

bzw.

```
ping -<OPTION> <HOSTNAME>.
```

Im Folgenden stelle ich das Arbeiten mit ping überaus detailliert dar. Das mache ich deshalb, weil ping *das* Werkzeug schlechthin für die Überprüfung von Netzwerkverbindungen ist und es trotz der einfachen Bedienung immer wieder zu Fehlern bei der Interpretation der Ergebnisse kommt. Ich bitte die Leser mit Netzwerkerfahrung hierfür um Nachsicht. In Abbildung 4.14 sehen Sie Versuche, bei denen die Verbindung sowohl bei der Verwendung einer IP-Adresse als auch eines Host-Namens funktioniert.

```

harald@ZE4:~$ ping6 -c2 fd91:e4a5:5e03::dd06
PING fd91:e4a5:5e03::dd06(fd91:e4a5:5e03::dd06) 56 data bytes
64 bytes from fd91:e4a5:5e03::dd06: icmp_seq=1 ttl=64 time=0.136 ms
64 bytes from fd91:e4a5:5e03::dd06: icmp_seq=2 ttl=64 time=0.150 ms

--- fd91:e4a5:5e03::dd06 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 999ms
rtt min/avg/max/mdev = 0.136/0.143/0.150/0.007 ms

harald@ZE4:~$ ping6 -c2 mobarb-6
PING mobarb-6(mobarb-6.example.org) 56 data bytes
64 bytes from mobarb-6.example.org: icmp_seq=1 ttl=64 time=0.137 ms
64 bytes from mobarb-6.example.org: icmp_seq=2 ttl=64 time=0.161 ms

--- mobarb-6 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 999ms
rtt min/avg/max/mdev = 0.137/0.149/0.161/0.012 ms
harald@ZE4:~$

```

Abbildung 4.14 Erfolgreicher Verbindungstest mit »ping6«

Ich habe bei jedem Versuch nur zwei Pings absetzen lassen. Am Ende erhalten Sie eine Zusammenfassung der Laufzeiten. Bei der Verwendung von ping (IPv4) sehen die Ausgaben identisch aus.

In Abbildung 4.15 sehen Sie drei fehlgeschlagene Pings. Versuch 1 mit der IP-Adresse und Versuch 2 mit einem bekannten Host-Namen zeigen Ihnen, dass die Geräte einfach nicht erreichbar sind. Das Ergebnis aus Versuch 3 können Sie mehrdeutig auslegen:

- Ist der Host-Name richtig geschrieben?
- Wurde der Host-Name in der Datei *hosts* oder der Zonendatei des Nameservers richtig eingetragen?
- Sie haben keine Aussage darüber, ob das Gerät tatsächlich nicht erreichbar ist.

Hinweis: Die Ausgaben für ping6 sehen ähnlich aus.

```

harald@ZE4:~$ ping -c2 192.168.0.55
PING 192.168.0.55 (192.168.0.55) 56(84) bytes of data.
From 192.168.0.34 icmp_seq=1 Destination Host Unreachable
From 192.168.0.34 icmp_seq=2 Destination Host Unreachable

--- 192.168.0.55 ping statistics ---
2 packets transmitted, 0 received, +2 errors, 100% packet loss, time 999ms
, pipe 2

harald@ZE4:~$ ping -c2 lp
PING lp (192.168.0.54) 56(84) bytes of data.
From ze4.example.org (192.168.0.34) icmp_seq=1 Destination Host Unreachable
From ze4.example.org (192.168.0.34) icmp_seq=2 Destination Host Unreachable

--- lp ping statistics ---
2 packets transmitted, 0 received, +2 errors, 100% packet loss, time 1014ms
, pipe 2

harald@ZE4:~$ ping -c2 lb
ping: unknown host lb
harald@ZE4:~$

```

Abbildung 4.15 Fehlgeschlagene Pings

4.4.2 Werkzeuge für Nameserver-Abfragen (»nslookup«, »host«, »dig«)

Sie finden wenigstens eines der Programme `nslookup`, `host` und `dig` auf Ihrem Betriebssystem. Ohne weitere Optionen zeigen sie im Erfolgsfall die gefundenen IP-Adressen zum Host-Namen an. Auch die Inverse-Suche ist möglich. Sie erhalten nach der Eingabe der IP-Adresse dann den Host-Namen angezeigt.

In Abbildung 4.16 sehen Sie die Arbeit mit `host`. Die beiden ersten Abfragen (Vorwärts- und Rückwärtssuche) führen zum Erfolg, die beiden anderen nicht. Wenn Sie auf einen bestimmten Nameserver zugreifen wollen, ergänzen Sie Ihre Abfrage einfach. Setzen Sie am Schluss Ihrer Eingabe den Namen des DNS-Servers:

```
host <SUCHBEGRIFF> <Nameserver>
```

```

$ host ze4
ze4.netzbuch has address 192.168.0.34
ze4.netzbuch has IPv6 address fd91:e4a5:5e03::dd05
$
$ host 192.168.0.34
34.0.168.192.in-addr.arpa domain name pointer ze4.netzbuch.
$
$ host ze5
Host ze5 not found: 3(NXDOMAIN)
$
$ host 192.168.0.55
Host 55.0.168.192.in-addr.arpa. not found: 3(NXDOMAIN)
$

```

Abbildung 4.16

Nameserver-Abfragen mit »host«

Mit `nslookup` (Abbildung 4.17) erhalten Sie einen Hinweis, ob der Nameserver die Information selbst hält oder von einem anderen bezogen hat (*non-authoritative*). Auch dieses Programm beherrscht die Reverse-Suche. Wenn Sie einen bestimmten Nameserver ansprechen möchten, geben Sie diesen am Schluss Ihrer Abfrage an.

```

harald : sh - Konsole
Datei Bearbeiten Ansicht Lesezeichen Einstellungen Hilfe
$ nslookup www.rheinwerk-verlag.de
Server:          217.0.43.177
Address:         217.0.43.177#53

Non-authoritative answer:
Name:   www.rheinwerk-verlag.de
Address: 46.235.24.168

$

```

Abbildung 4.17 Ausgabe von »nslookup«

Eine recht umfangreiche Ausgabe (Abbildung 4.18) liefert Ihnen dig. Sie setzen Ihren Suchauftrag, wenn Sie auf einen bestimmten DNS-Server zugreifen möchten, in der Form

```
dig @<Nameserver> <Hostname.Domain>
```

ab. Sie erhalten umfangreiche Informationen über den beteiligten Nameserver und bekommen die Antwort so dargestellt, wie sie in der Zonendatei hinterlegt wurde.

```

harald@ZE4: -- Befehlsfenster - Konsole
Sitzung Bearbeiten Ansicht Lesezeichen Einstellungen Hilfe
$ dig @localhost ze4.netzbuch

; <<>> DiG 9.6.-ESV-R3 <<>> @localhost ze4.netzbuch
; (2 servers found)
;; global options: +cmd
;; Got answer:
;; ->HEADER<- opcode: QUERY, status: NOERROR, id: 56198
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 1, ADDITIONAL: 2

;; QUESTION SECTION:
ze4.netzbuch.                IN      A

;; ANSWER SECTION:
ze4.netzbuch.                1800    IN      A      192.168.0.34

;; AUTHORITY SECTION:
netzbuch.                    1800    IN      NS      localhost.

;; ADDITIONAL SECTION:
localhost.                   10800   IN      A      127.0.0.1
localhost.                   10800   IN      AAAA   ::1

;; Query time: 0 msec
;; SERVER: ::1#53(::1)
;; WHEN: Thu May 12 22:50:11 2011
;; MSG SIZE rcvd: 113

```

Abbildung 4.18 Ausgabe von »dig«

4.4.3 Mitschnitte von DNS-Abfragen mit Netzwerkdiagnoseprogrammen

Sie können das Wirken von Schad- und Spionageprogrammen in Ihrem lokalen Netz manchmal auch durch deren DNS-Abfragen beobachten. Für die Ersteller solcher

Programme bringt die Verwendung von festen IP-Adressen je nach Konzept oft eher Nachteile. Daher sollten Sie sich mit dem Aufbau von DNS-Abfragen und der Handhabung diverser Netzwerkd Diagnoseprogramme vertraut machen. Aber: Diese Programme untersuchen jeglichen Netzwerkverkehr! Beachten Sie die rechtliche Grauzone, in der Sie sich damit bewegen. Beachten Sie strikt die Gesetzeslage, eventuell vorhandene Betriebsvereinbarungen und Informationspflichten gegenüber Mitarbeitern und Kollegen.

In Listing 4.3 bis Listing 4.7 finden Sie eine Gegenüberstellung von Abfragen und ihren Spuren im Netzwerk. Bedenken Sie, dass in Ihrem beruflichen Alltag meist andere Netzerkanwendungen die Anfragen stellen (Webbrowser, Mail-Clients usw.).

Jeweils wichtige Details, wie etwa Fragen und Antworten des Protokolls und Kommandos, finden Sie hervorgehoben. Der Nameserver lag auf dem localhost.

```
$ host ze4 localhost
zentrale# tcpdump -t -n -i lo0 -s 512 ip6
IP6 ::1.61937 > ::1.53: 43085+ A? ze4.netzbuch. (30)
IP6 ::1.53 > ::1.61937: 43085* 1/1/2 A 192.168.0.34 (113)
IP6 ::1.63863 > ::1.53: 58559+ AAAA? ze4.netzbuch. (30)
IP6 ::1.53 > ::1.63863: 58559* 1/1/2 AAAA fd91:e4a5:5e03::dd05 (125)
IP6 ::1.55738 > ::1.53: 52278+ MX? ze4.netzbuch. (30)
IP6 ::1.53 > ::1.55738: 52278* 0/1/0 (92)
```

Listing 4.3 Anfrage (Forward, »ze4« hat eine IPv4- und eine IPv6-Adresse)

In Listing 4.3 sehen Sie auch die Transaktionsnummern der Anfragen (hier fett dargestellt). Damit werden Fragen und Antworten vom Client zugeordnet. Die Host-Anfrage erkundigte sich nach dem A-, AAAA- und MX-Record des angegebenen Namens. Neben der Transaktionsnummer sehen Sie das Zeichen »*« für die Frage, »*« und eine Ziffernfolge für die Antwort. An der 1 an der ersten Stelle erkennen Sie, dass eine Auskunft vorliegt, an der 0 dagegen eine negative Antwort. Sie erkennen im Beispiel hier, dass der Host ze4 eine IPv4- und IPv6-Adresse, aber keinen MX-Eintrag in der Zonendatei hat.

```
$ host fd91:e4a5:5e03::dd05 localhost
zentrale# tcpdump -t -n -i lo0 -s 512 ip6
IP6 ::1.56839 > ::1.53: 64563+ PTR?
5.0.d.d.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.3.0.e.5.5.a.4.e.1.9.d.f.ip6.arpa.
(90)
IP6 ::1.53 > ::1.56839: 64563* 1/1/2 PTR ze4.netzbuch. (183)
```

Listing 4.4 Reverse-Abfrage mit IPv6-Adresse

Listing 4.4 zeigt Ihnen die Pointer-Frage (PTR) und die Antwort in Form der in umgekehrter Reihenfolge dargestellten IPv6-Adresse.

```
$ host 192.168.0.41 localhost
zentrale# tcpdump -t -n -i lo0 -s 512
IP6 ::1.32301 > ::1.53: 27945+ PTR? 41.0.168.192.in-addr.arpa. (43)
IP6 ::1.53 > ::1.32301: 27945* 1/1/2 PTR mobarb.0.168.192.in-addr.arpa.
(131)
```

Listing 4.5 Reverse-Abfrage mit IPv4-Adresse

In Listing 4.5 finden Sie die PTR-Frage nach dem Namen einer IPv4-Adresse. Die Adresse ist auch in der Antwort in umgekehrter Reihenfolge geschrieben.

```
$ host ze5 localhost
zentrale# tcpdump -t -n -i lo0 -s 512 ip6
IP6 ::1.60499 > ::1.53: 6436+ A? ze5.netzbuch. (30)
IP6 ::1.53 > ::1.60499: 6436 NXDomain* 0/1/0 (92)
IP6 ::1.27587 > ::1.53: 3313+ A? ze5. (21)
IP6 ::1.53 > ::1.27587: 3313 NXDomain 0/1/0 (96)
```

Listing 4.6 Abfrage (Forward) mit dem Namen eines nicht existierenden Hosts

Sie sehen in Listing 4.6 die Namensanfrage mit negativer Auskunft. Das Programm versuchte es auch mit dem verkürzten Host-Namen.

```
zentrale# tcpdump -t -n -i lo0 -s 512
IP6 ::1.38937 > ::1.53: 282+ PTR? 61.0.168.192.in-addr.arpa. (43)
IP6 ::1.53 > ::1.38937: 282 NXDomain* 0/1/0 (105)
```

Listing 4.7 Reverse-Abfrage mit nicht existierender IP-Adresse

Listing 4.7 zeigt Ihnen den Verlauf einer fehlgeschlagenen Reverse-Anfrage auf.

4.5 Zentrale Netzwerkgeräte auf Sicherungs- und Vermittlungsebene

Wenn Sie in Ihrem Netzwerk bestimmte Technologien verwenden (Koaxialkabel, WLAN, HUB), lassen Kollisionen von Datenpaketen die Übertragungsrate sinken. Je mehr Teilnehmer Sie dort zusammenschalten, desto eher sind diese mit der Verkehrsregelung statt mit der Datenübertragung beschäftigt.

4.5.1 Bridges – Verbinden von Netzwerkteilen

Um das Verkehrsaufkommen etwas zu entflechten, können Sie ein Netzwerk in verschiedene Segmente aufteilen und diese über Bridges verbinden. Die Bridge registriert anhand der ARP-Pakete, an welchem Anschluss sich welche MAC-Adresse befindet. Datenpakete, die zwei MAC-Adressen innerhalb eines Netzwerksegments betreffen, werden nun nicht mehr an die Teilnehmer im anderen Zweig geleitet. Damit tritt der gewünschte Entlastungseffekt ein. Die Bridge hält die Einträge in der Adresstabelle durchschnittlich fünf Minuten und verwirft sie dann.

Eine Bridge leitet Datenpakete immer dann weiter, wenn diesen keine MAC-Adresse zugeordnet werden kann, also ARP-Anforderungen, aber auch Multicasts. Hier verhält sich die Bridge transparent.

Auch die Bridge trägt allerdings etwas zu Verzögerungen in der Datenübertragung bei, etwa wenn Daten von einem Segment in das andere gesendet werden. Sie muss genauso wie die anderen Netzwerkteilnehmer warten, bis das Medium zum Senden für sie frei ist. Dazu speichert sie die Pakete so lange, bis sie an die Zielstation weitergereicht werden können (*Store-and-Forward-Bridging*).

Die Bridge-Technik finden Sie in vielen Geräten:

- in Dual-Speed-Hubs: Beim Verbinden von zwei Netzwerksegmenten mit verschiedenen Geschwindigkeiten müssen Datenpakete zwischengespeichert werden – eine klassische Aufgabe einer Bridge.

- in bestimmten Arten von Medienkonvertern, bei denen nicht nur transparent umgesetzt wird

- in WLAN-Zugangsgeräten

- in Switches

Im Rahmen der Rechner-Virtualisierung bilden sogenannte *Bridgedevices* die Verbindung des Gastsystems zur Außenwelt. Bridgedevices erweitern »reale« Netzwerkkarten um virtuelle Netzwerkanschlüsse für das Gastsystem.

4.5.2 Hubs – die Sammelschiene für TP-Netze

Mit einem Hub können Sie ein sternförmiges Netz mittels TP-Verkabelung (10Base-T)

aufbauen. Trotz des technisch unkomplizierteren Aufbaus bleibt Ihnen aber das CSMA-Zugriffsverfahren mit seinen Kollisionen als Nachteil erhalten. Beachten Sie deshalb, dass Hubs ohne weitere Funktionalität zum OSI-Layer 1 gehören! Sie entsprechen in diesem Fall dem Koaxialkabel der 10Base-2- und 10Base-5-Netze!

Eine Zeit lang konnten Sie auch Dual-Speed-Hubs kaufen. Zunächst konnten nur Partner mit der jeweils gleichen Geschwindigkeit miteinander kommunizieren. Mit dem Einbau einer internen Bridge konnte dagegen auch mit verschiedenen »schnellen« Partnern der Netzbetrieb ermöglicht werden. Das Problem der Kollisionen bestand aber weiter, da in Altbeständen noch teilweise Hubs in ihrer klassischen Ausprägung im Einsatz sind. Mit der internen Bridge wurde der Hub in den OSI-Layer 2 befördert.

Im Rahmen von Netzerneuerungen und -erweiterungen schaffen Sie diese »Altlasten« am besten ab.

4.6 Switches – Verbindungsknoten ohne Kollisionen

Heute bilden Switches die zentralen Netzknoten. Sie erhalten Switches in verschiedenen Größen- und Leistungsvarianten im Fachhandel. Für den Heimbereich und kleine Büros reichen meist nicht weiter konfigurierbare Kleingeräte mit Steckernetzteil aus. Möchten Sie maximale Sicherheit, Konfigurationsmöglichkeiten und Anschlüsse, greifen Sie auf ein Gerät mit 19-Zoll-Einschub-Gehäuse, vielen Höheneinheiten, redundanten Netzteilen und mit allerlei Zusatzfunktionen zurück. Dazwischen finden Sie ebenfalls konfigurierbare Geräte im unteren und mittleren Preissegment. Für den Heimbereich und das Kleingewerbe bekommen Sie auch Ausführungen mit eingebautem Router für den Internetzugang.

4.6.1 Funktionalität

Ein Switch besteht im Grunde aus einem Hub, der an jedem Port mit einer Bridge ausgestattet ist. Intern arbeitet ein schneller Hochleistungsdatenbus oder einfach eine Matrix. Die Bridges wiederum sind damit (gemeinsam) verbunden. Damit ergibt sich für jeden einzelnen Anschluss ein eigener Kollisionsbereich, was im Allgemeinen einen sehr hohen Datendurchsatz bewirkt, weil Overhead durch die Kollisionsregelung kaum vorhanden ist bzw. wegfällt. Durch den Einsatz eines Switchs vergrößert sich die Reichweite des Netzwerks. Weil die Lauflänge des jeweiligen Netzwerktyps ja zwischen dem Switch-Anschluss und dem Teilnehmergerät gilt, bewirkt dies quasi eine Verdopplung der Reichweite. Mit einem Switch isolieren Sie den Datenverkehr zwischen den Teilnehmern dergestalt, dass im Idealfall 1:1-Verbindungen mit voller Geschwindigkeit gebildet werden (Abbildung 4.19).

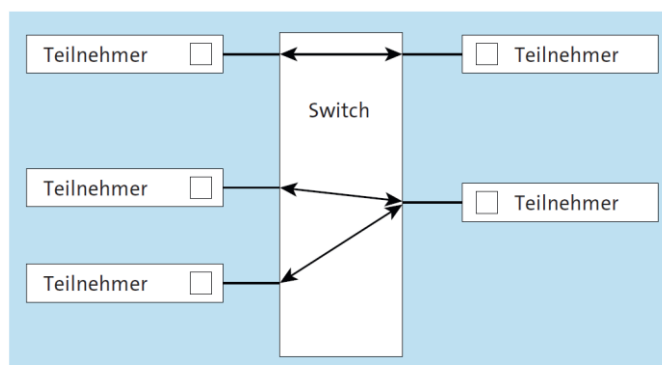


Abbildung 4.19 Geschaltete Verbindungen im Switch

Was Geschwindigkeit und Datendurchsatz beim Switch betrifft, sollten Sie auch folgende Tatsachen berücksichtigen:

Eine Bridge muss sich alle MAC-Adressen merken, die an einem Hub auftauchen. Gleiches gilt für die interne Bridge eines Switch-Ports, wenn ein Hub mit ihr verbunden

wird. In diesem Fall treten auch wieder vermehrt Kollisionen auf, die den Datendurchsatz bremsen.

Mit dem Einsatz des Switchs ist auch der Vollduplex-Betrieb möglich, weil auf Kollisionen in der Regel keine Rücksicht genommen werden muss. Dies ist so lange möglich, wie kein Hub an den Switch angeschlossen wird.

Ist der Switch dagegen die einzige Netzwerktechnologie, die in einem Netz zum Einsatz kommt, muss sich die einzelne »Port-Bridge« nur die eine Adresse des angeschlossenen Geräts merken.

Broadcasts, wie z. B. ARP-Anforderungen, werden über alle Ports eines Switchs weitergegeben.

Anstelle der Kollisionen stellen diese nun den Hauptverursacher für eine Reduzierung des effektiven Durchsatzes dar.

4.6.2 Schleifen – Attentat oder Redundanz?

Wenn Sie »einfache« Switches verwenden, können Sie das gesamte Netzwerk einfach außer Gefecht setzen. Schließen Sie ein Patchkabel an zwei Ports des Switchs (Abbildung 4.20) an, und setzen Sie ein Broadcast-Paket ab.

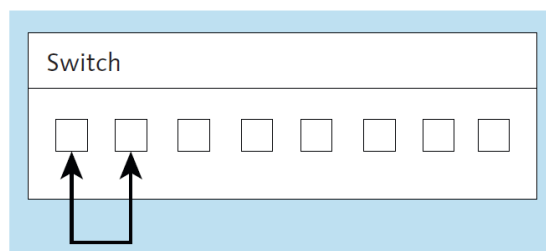


Abbildung 4.20 Loop

Was passiert? Das Datenpaket gelangt über mehrere Wege an beide Seiten der Bridges. Diese leiten das Paket buchstäblich überallhin, und die Bridges bekommen es vervielfältigt von überallher zurück. Das schaukelt sich hoch: Der Netzwerkverkehr wird nur noch von diesen Paketen beherrscht und bricht schließlich zusammen. Sicher kennen Sie die akustische Rückkopplung bei Beschallungsanlagen (das Pfeifen), wenn das Mikrofon den Schall der Lautsprecher aufnimmt und wieder zum Verstärker schickt. Damit können Sie die Wirkung von Schleifen (Loops) in etwa vergleichen.

Das Gleiche passiert auch, wenn Sie doppelte Verbindungen über mehrere Switches schalten (Abbildung 4.21).

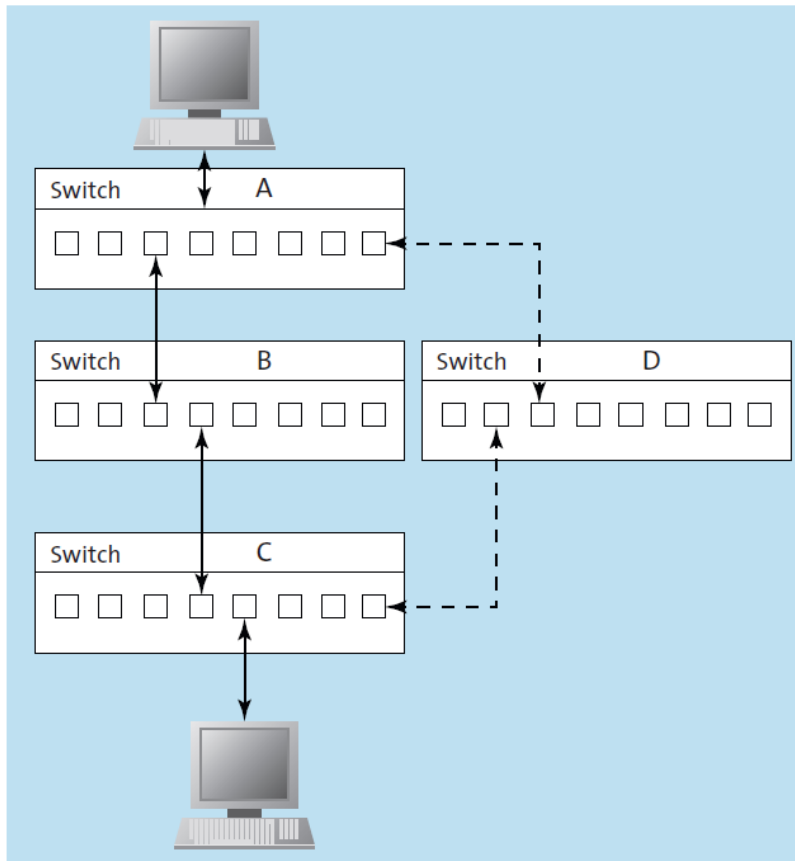


Abbildung 4.21 Redundante Übermittlungswege über mehrere Switches

Beherrschen die beteiligten Switches das *Spanning Tree Protocol (STP)* und haben Sie es auf allen auch konfiguriert und aktiviert, schlägt der Attentatsversuch mit dem Patchkabel fehl. Schleifen werden erkannt, einer der Ports wird gesperrt. Der Switch überprüft in festgelegten Intervallen den Zustand. Aber das ist nicht alles, was das STP zu leisten vermag. Vielmehr bekommen Sie beim Ausfall einer Strecke ohne persönlichen Eingriff eine Ersatzverbindung. Im Beispiel der Abbildung 4.21 werden die Pakete zwischen den Partnern über die Switches A, B und C geleitet. Fällt zwischen den Switches die Verbindung aus, wird der Verkehr automatisch über den sonst geblockten Weg über Switch D geführt.

Der Spanning Tree bildet ein Baummodell. Wie in der Natur auch sind die Äste eines Baums stets nur am Stamm miteinander gemeinsam verbunden, ihre Enden ragen aber ins Freie und bilden keine Verbindungen aus.

Switches tauschen mit diesem Protokoll Topologie-Informationen im Layer 2 aus. Damit funktioniert dies unabhängig vom gewählten Netzwerkprotokoll. Ein Switch wird dabei als *Root-Bridge* tätig, die die Topologie ermittelt und berechnet. Dies geschieht dadurch, dass sich die beteiligten Switches gegenseitig Pakete nach der *Bridge Protocol Data Unit (BPDU)* zusenden. Aus der darin enthaltenen Prioritätsangabe und gegebenenfalls der MAC-Adresse wird das Gerät mit den niedrigsten Werten ermittelt und zur Root-Bridge erklärt. Mit in die Konfiguration fließen auch *Pfadkosten* ein. Je schneller eine Verbindung ist, desto »kostengünstiger« im Sinne der Pfadkosten gilt sie. Die Folgen daraus erkennen Sie leicht: Langsame Verbindungen werden zu Redundanzstrecken herabgestuft, und die Switches sperren die betroffenen Anschlüsse. Dabei bleibt aber die Root-Bridge stets erreichbar. Sie sendet die Konfigurations-BDUs aus. Fällt eines der Geräte aber aus, wird die Topologie-Änderung von den anderen Switches erkannt. Die Baumstruktur (Spanning Tree) wird neu berechnet. Dabei werden die gesperrten Ports nun teilweise geöffnet, damit die Datenpakete auf alternativen Wegen ihr Ziel erreichen können.

Verbindungsabbrüche lassen sich während der Phase der Neuberechnung einer Topologie

nicht vermeiden. Beim klassischen STP sind Ausfallzeiten von bis zu einer Minute in großen Netzen als normal zu betrachten. Störungen des Betriebs sind damit leider unvermeidbar.

Das klassische STP wurde schon vor geraumer Zeit durch das *Rapid Spanning Tree Protocol (RSTP)* bzw. bei VLANs durch das *Multiple Spanning Tree Protocol (MSTP)* abgelöst. Bei Letzterem bildet jedes VLAN eine eigene Instanz des Spanning Trees. RSTP und MSTP vermindern die Ausfallzeiten dramatisch. Es wird nach dem Erkennen und der Signalisierung von Änderungen durch *Topology Change Notification BPDUs (TCN)* nicht die komplette Topologie neu berechnet, sondern der Betrieb zunächst so weit wie möglich weitergeführt. Erst nach der Neuberechnung der nun zur Verfügung stehenden Strecken wird der neue Spanning Tree zusammengestellt.

IEEE-Normen zum STP und zu Switches

802.1D-2004, 802.1d, 802.1w, 802.1s, 802.1Q

4.6.3 Verbindungen zwischen Switches

(Link Aggregation, Port Trunking, Channel Bundling)

Sie können zwei Switches mit einem Patchkabel verbinden. Im Idealfall haben beide Switches hierfür einen eigenen Hochgeschwindigkeits-Port, sodass Sie die Verbindung

auch wirklich nur mit einem Kabel aufbauen können. Ein anderer Weg besteht darin, einige der Anschlüsse zu opfern und die Switches durch mehrere Kabel miteinander zu koppeln (Abbildung 4.22). Damit jetzt nicht das STP einschreitet und bis auf eine alle parallelen Verbindungen blockiert, müssen Ihre Geräte die *Link Aggregation* nach IEEE 802.3ad beherrschen. Damit aus den zwei oder mehreren Strecken auch eine einzige logische Verbindung wird, gibt es auf dem Layer 2 als Erweiterung den *Link Aggregation Control Layer (LACL)*. Die Switches transportieren die Daten mittels des *Link Aggregation Control Protocols (LACP)*.

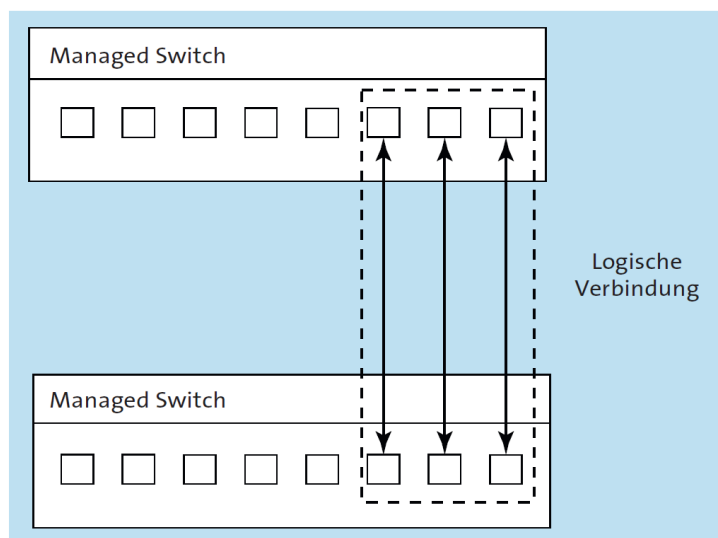


Abbildung 4.22 Zusammenfassung von Einzelverbindungen zu einer logischen Verbindung

Gegenüber der Verwendung von Hochgeschwindigkeits-Ports mit einem Kabel haben Sie hier den Vorteil der Redundanz. Fällt ein Kabel aus, reduziert sich zwar die Geschwindigkeit, die Strecke selbst bleibt aber erhalten. Wollen Sie mit dem *Port Trunking* schnelle Übertragungen ermöglichen, müssen Sie aber ein entsprechend großes »Port-Opfer« bringen. Die Geschwindigkeit addiert sich aus der der einzelnen Ports, die gleich schnell sein und im Vollduplex-Modus laufen müssen. Andererseits sind Sie auch flexibel. Wenn Sie nur aus besonderen Anlässen einmal eine höhere Übertragungsbandbreite

benötigen, benutzen Sie diese Vorgehensweise nur im Bedarfsfall. Vergleichen Sie unbedingt auch die Preise bei der Anschaffung. Möglicherweise ist der Switch mit dem Hochgeschwindigkeits-Port (z. B. 10 Gbit/s) teurer als jener, der acht oder zehn Ports mehr besitzt und mit dem Sie im Grunde das Gleiche erreichen können.

4.6.4 Virtuelle Netze (VLAN)

Virtuelle Netze sind logische Teilnetze, die Sie an einem oder mehreren dafür geeigneten Switches bilden können. Gründe für den Einsatz von virtuellen Netzen gibt es einige:

Die Unterteilung großer Netzwerke in Unterabschnitte hilft Ihnen, Broadcasts einzudämmen und damit die Netzlast geringer zu halten (Bildung mehrerer Broadcast-Domänen).

Sie können die betriebliche Organisation mittels VLANs abbilden (Einkauf, Personal, Vertrieb, Fertigung ...).

Sie können Ihr Netz auch nach Anwendungen einteilen: Internettelefonie (VoIP), Multimedia-Dienste usw.

Virtuelle Netze können Sie auf verschiedene Art und Weise erstellen. Ausschlaggebend hierfür ist, welche Verfahren Ihre Switches beherrschen. Wenn Sie jetzt hier in diesem Zusammenhang auch von *Routern* lesen, hängt dies damit zusammen, dass sogenannte *Layer-2-Switches* aus Kostengründen wohl weiter verbreitet sind als Layer-3-Geräte. Diese beinhalten Router-Funktionen.

Arten der VLANs

Portbasiertes VLAN (Abbildung 4.23): ein Port – ein VLAN. Ein Port kann immer nur Bestandteil eines VLANs sein. Möchten Sie die virtuellen Netze miteinander kommunizieren lassen, benötigen Sie hierfür einen Router. Es ist so, als würden Sie zwei voneinander getrennte Switches verwenden. Dieses Verfahren hat auch den Vorteil, dass eine klare und übersichtliche Trennung möglich ist und es von kostengünstigeren Switches schon beherrscht wird. Sollen sich portbasierte VLANs über mehrere Switches erstrecken, müssen Sie für jedes VLAN ein eigenes Verbindungskabel benutzen.

Paketbasiertes VLAN (tagged VLAN, Abbildung 4.24): Ein Port kann mehreren VLANs angehören. Vor allem, wenn Sie per Hochgeschwindigkeits-Port zwei Switches verbinden möchten, ist dies von Vorteil. Teilen Sie diese Ports allen an den Switches definierten *tagged VLANs* zu, genügt ein einziges Kabel, damit die Daten von und in alle beteiligten VLANs geleitet werden. Diese Verbindung trägt auch die Bezeichnung *Trunk*.

VLAN-Trunks sind nicht identisch mit denen der Link Aggregation! Beim *tagged VLAN* fügt der Switch oder der Netzwerkteilnehmer an die Ethernet-Frames eine Kennzeichnung an, die das Paket einem VLAN zuweist. Ältere Switches und Netzwerkgeräte (Printserver, PCs) können unter Umständen nicht damit umgehen und verwerfen die überlangen Pakete (1522 statt 1518 Byte Länge). Norm: IEEE 802.1Q.

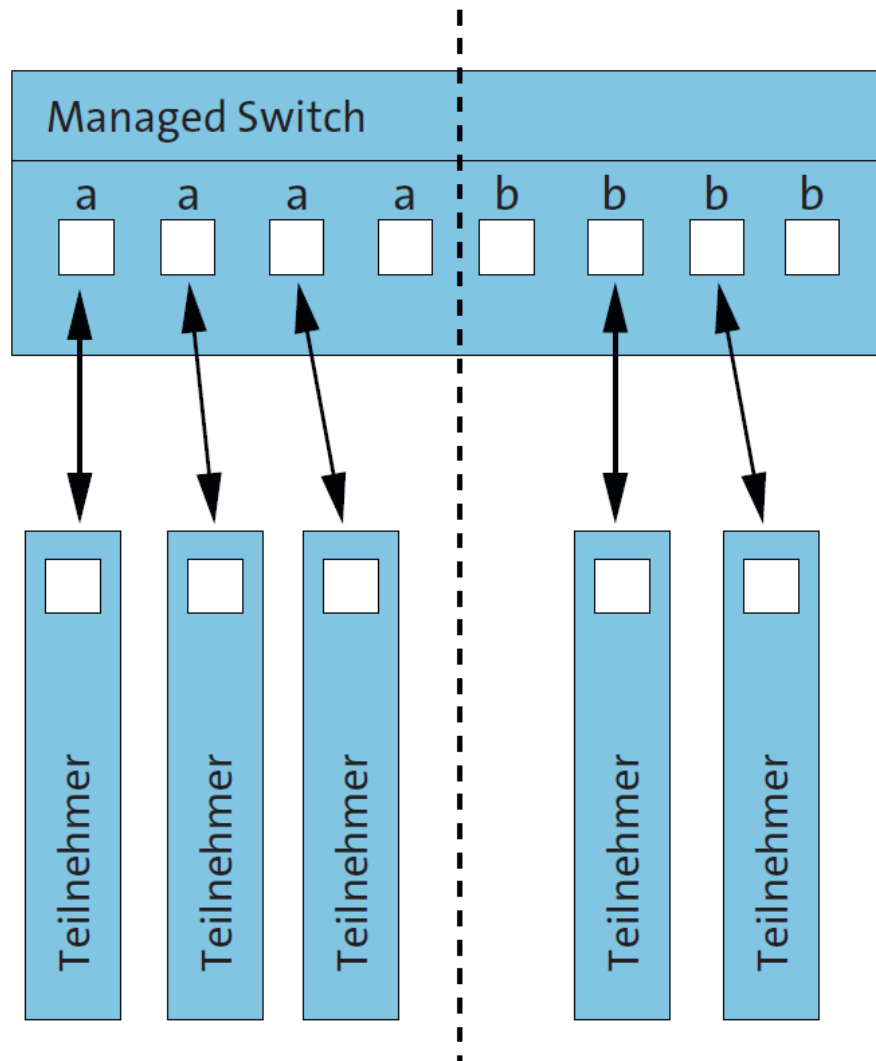


Abbildung 4.23 Portbasiertes VLAN

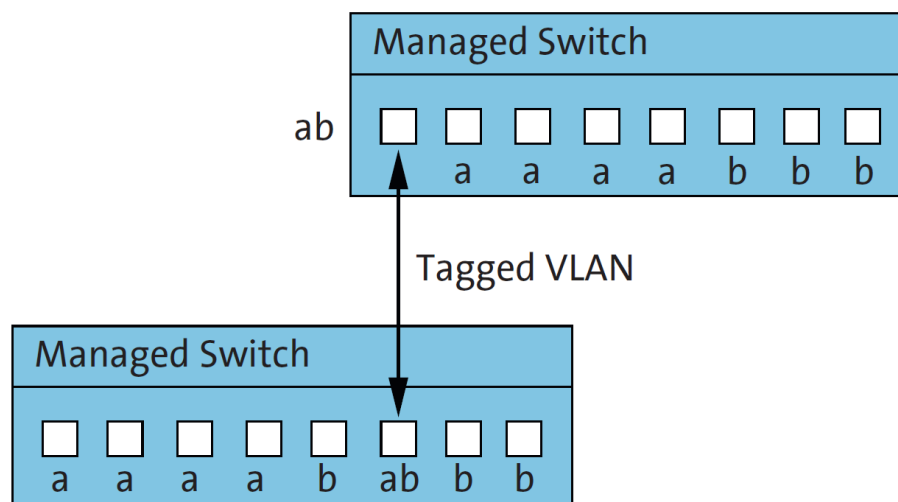


Abbildung 4.24 Paketbasiertes VLAN (tagged VLAN)

Wie vorhin schon erwähnt, sind die Pakete mit dem VLAN-Tag wegen der Segmentkennung um 4 Byte länger. Diese 4 Byte werden zwischen der Quell-MAC-Adresse und dem Typen-Feld eingeschoben. Das VLAN-Tag selbst hat den in Abbildung 4.25 gezeigten Aufbau.

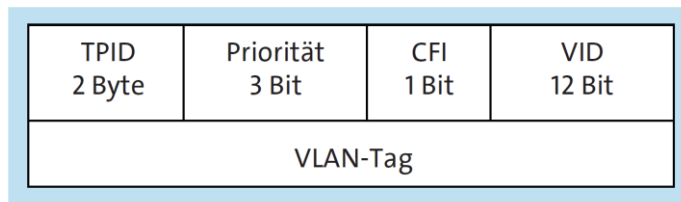


Abbildung 4.25 Aufbau des VLAN-Tags

Bestandteile des VLAN-Tags nach IEEE 802.1Q (zu [Abbildung 4.25](#))

- ▶ **TPID** – *Tag Protocol Identifier*, vorgegebener Wert 0 × 8100. Damit erkennen Netzwerkteilnehmer, dass es sich um einen »übergroßen« Frame handelt.
- ▶ **Priorität** – Informationen über die Priorität
- ▶ **CFI** – *Canonical Format Indicator*, Wert 0: Die MAC-Adresse wird kanonisch angegeben, Wert 1: Die Angabe der MAC-Adresse ist nicht kanonisch.
- ▶ **VID** – *VLAN Identifier*: Angabe des VLANs, zu dem der Frame gehört

Wenn Sie den Switch konfigurieren, stehen Sie vor der Wahl, ob Sie statische oder dynamische VLANs möchten.

Statisches oder dynamisches VLAN?

- ▶ **Statisches VLAN**: Die einzelnen Ports werden fest konfiguriert. Die Eigenschaften bleiben unabhängig vom anschließend darüber laufenden Netzwerkverkehr erhalten.
- ▶ **Dynamisches VLAN**: Hier bestimmt der Inhalt der Frames die Zugehörigkeit zu den VLANs: MAC-Adressen, IP-Adressen, Netzwerkprotokolle und IP-Port-Nummern können ausgewertet werden. Da hier die Manipulationsgefahr recht groß ist, sollten Sie zumindest in sicherheitskritischen Umgebungen darauf verzichten!

4.6.5 Switch und Sicherheit

Switches werden als zentrale Netzwerkinfrastruktur allzu gerne angegriffen. Der Angreifer möchte sie entweder gleich außer Gefecht setzen und den Netzwerkverkehr zusammenbrechen lassen oder in durch VLANs abgeschottete Bereiche eindringen. Da die Angriffe sich ja auf der OSI-Ebene 2 abspielen, schlagen sie auf die höheren Schichten mit allen Folgen durch!

Angriffspunkte beim Switch

- ▶ **Missbrauch des STPs:** Durch gezielte Falschinformationen werden Zusammenbrüche und nicht vorhandene Links gemeldet, womit der Angreifer zur Root-Bridge aufsteigt und sämtlichen Netzwerkverkehr entweder mitliest oder (teilweise) unterbricht.
- ▶ **Einfall über ungesicherten Trunking-Port:** Der Angreifer tarnt sich selbst als Switch und fügt VLAN-Tags an die Pakete an. Der angegriffene Switch möchte den Pseudo-Switch aufnehmen und bietet ihm alle nicht portbasierten VLANs zum Zugriff an.
- ▶ **Überlauf:** Mit einer Flut von Paketen mit den unterschiedlichsten MAC-Adressen läuft die MAC-Tabelle im Switch über. Wenn keine Sicherheitsfunktionen greifen, wird der Switch alle Pakete an alle Ports weiterleiten. Er wird damit zum Hub herabgestuft. Der Datendurchsatz sinkt dramatisch, aber der Angreifer kann sämtlichen Netzwerkverkehr mitlesen.
- ▶ **Gefälschte Frames:** Beim dynamischen VLAN öffnen manipulierte Pakete den Zugang zu eigentlich verborgenen VLANs. MAC-Adressen und Dienstekennungen lassen sich sehr einfach fälschen!

Sie können einem Großteil der Gefahren durch die richtige Konfiguration begegnen. Natürlich kommt es auch auf die Gefährdung selbst an. Von einem Büro voller EDV-Laien, die mit ihrem Job zufrieden sind, geht natürlich eine geringere Gefahr aus als von einem Hochtechnologie-Entwicklungsbüro, das von Schnüfflern und Personalabwerbern geradezu umzingelt ist.

Bei Letzterem müssen Sie mit qualifizierten Angriffen aller Art über das Netzwerk rechnen. Und die Angriffe können durchaus von innen her passieren, sei es durch frustrierte Mitarbeiter mit dem notwendigen Fachwissen oder durch arglos eingeschleppte Schadprogramme. Bedenken Sie auch, dass die Bequemlichkeit des (Netzwerk-)Administrators das Netzwerk für Eindringlinge öffnet.

Sicherheit für den Switch

- ▶ **Unbenutzte Anschlüsse:** Stecken Sie die Patchkabel zu unbenutzten Räumen und Netzwerkdozen vom Switch ab.
- ▶ **Zugriff auf den Switch:** Bauen Sie den Switch in einen abschließbaren Schrank ein, der in einem nicht für jedermann zugänglichen Raum steht.
- ▶ **Sicherheitsmechanismen:** Aktivieren Sie die Sicherheitsmechanismen, die die Firmware des Switches enthält, z. B. gegen das »Überschwemmen« mit MAC-Adressen.
- ▶ **Kennwörter:** Ändern Sie als Erstes überhaupt das Standardpasswort am Switch gegen eines, das nicht zu erraten ist.
- ▶ **Dynamisches VLAN:** Wenn irgend möglich, verwenden Sie es nicht. Eindringlinge finden immer wieder Wege, Sicherheitsmaßnahmen am Switch zu überlisten.

MAC-Adressen lassen sich leicht ändern und sind kein Sicherheitskriterium für den Netzwerkzugang.

- ▶ **Nicht benutzte Ports:** Weisen Sie diese einem eigenen VLAN zu (portbasiert!), oder deaktivieren Sie sie.
- ▶ **Port-Trunking:** Deaktivieren Sie es bei den Ports, die nicht entsprechend mit einem anderen Switch verbunden sind.
- ▶ **Firmware:** Beachten Sie Sicherheitswarnungen des Herstellers Ihres Switches, und führen Sie gegebenenfalls Updates der Firmware durch.

Wenn Sie all dies umsetzen, ist der Switch nicht mehr die große Gefahrenquelle im Netzwerk.

4.6.6 Geräteauswahl

Nachdem Sie jetzt alles über die Fähigkeiten von Switches wissen, können Sie Ihre Kaufentscheidungen leichter treffen. Sie klären zunächst die Zahl der zu verbindenden Netzwerkteilnehmer, das Sicherheitsbedürfnis und die Notwendigkeit weiterer Leistungsmerkmale wie VLANs, Spanning Tree usw. ab. Letztendlich werden Sie auch auf den Preis achten.

Vielleicht hilft Ihnen die kleine Zusammenstellung in Tabelle 4.7, die Kaufentscheidung leichter zu treffen.

Nutzer	Ausstattung
Privathaushalte und Kleinstgewerbe	Oftmals sind Switches mit vier oder fünf Anschlüssen zusammen mit Internet-Routern und teilweise WLAN-Einrichtungen kombiniert. Sie sind meist nicht weiter konfigurierbar und verfügen über einen relativ kleinen Speicher für die MAC-Adressen (2, 4 oder 8 kB). Standardgeräte bieten einen Durchsatz von 100 Mbit/s, seltener 1 Gbit/s. Reine »Klein-Switches« bieten meist fünf bis acht Anschlüsse. Die Stromversorgung erfolgt in der Regel mittels Steckernetzteil. Für die bei Jugendlichen beliebten LAN-Partys kommen überwiegend nicht managebare Geräte mit 16 oder 48 Ports infrage, die selbst mit 1-Gbit/s-Ports schon günstig auf dem Gebrauchtmärkte erhältlich sind.
Klein- und Mittelbetriebe, größere Haushalte	Hier kommen Ausführungen mit 8, 16, 24 oder 48 Ports zum Zuge. In diesen Größenordnungen sind sowohl nicht managebare als auch managebare Geräte üblich. Teilweise besitzen sie Gehäuse im 19-Zoll-Formfaktor und können so in Server- oder Netzwerkschränke eingebaut werden. Erweiterungssteckplätze für LWL-Anschlüsse oder Hochgeschwindigkeits-Ports können eingebaut sein.
Großbetriebe und -behörden	Managebare Switches mit redundanten Netzteilen und umfangreichen Managementfunktionen und einer hohen Port-Zahl arbeiten hier in 19-Zoll-Schränken. Zudem werden Switches benutzt, die man zu einem »logischen« Gerät zusammenkoppeln kann (Trunking, Bündelung). In diesem Umfeld kommen auch Klein-Switches mit vier oder acht Ports zum Einsatz, teilweise auch mit zusätzlichem Glasfaseranschluss. Sie dienen zum Verbinden von wenigen Geräten in einem Büro, ein »Uplink« (Kupfer oder Glasfaser) dient der Verbindung mit dem zentralen Switch. Was hier an Verkabelungskosten gespart wird, geht zulasten des Datendurchsatzes. Die Leistungsfähigkeit eines Ports des zentralen Switchs wird am angeschlossenen Klein-Switch mit den weiteren Anschlüssen geteilt. Im normalen Büroalltag hat dies wenig Bedeutung. Bei besonders netzlastigen Verfahren (Multimedia, IP-Telefonie) allerdings können solche Konfigurationen an die Grenzen ihrer Leistungsfähigkeit geraten. Ferner sind

	im Bezug auf Sicherheitsanforderungen nicht alle Konfigurationen möglich.
--	---

Tabelle 4.7 Einkaufshilfe für Switches

4.6.7 Anzeigen und Anschlüsse am Switch

In Tabelle 4.8 finden Sie Begriffe, mit denen Leuchtanzeigen und Anschlüsse bei den meisten Switch-Modellen beschriftet sind.

Anzeige/Bedienelement	Bedeutung
10/100/1000 oder Farbkennzeichnung	Anzeige der Übertragungsrate
Link	leuchtet: Verbindung ist in Ordnung bzw. angeschlossenes Gerät ist aktiv. flackert/blinkt: Datenübertragung ist aktiv.
ACT	flackert/blinkt: Datenübertragung ist aktiv.
PWR	leuchtet: Netzstromversorgung, Switch ist mit Stromnetz verbunden.
Zahlenangaben über Steckplatz/bei Link-LEDs	leuchtet: Port ist mit aktivem Gerät verbunden (siehe Link).
UPLINK	Verbindungsanschluss für weiteren Switch oder Router, meist mit höherer Geschwindigkeit
Auto MDI(X)	Beschriftung: Gerät erkennt automatisch, ob das angeschlossene Kabel gemäß EIA/TIA-T568 A oder -B belegt ist (Crosskabel/Normalbelegung).
Console	Anschluss: SUB-D-9 oder RJ45: für Konfigurationsarbeiten per serieller Schnittstelle
SFP	Anschluss: <i>Small Form-factor Pluggable</i> , nimmt kleine Erweiterungssteckmodule mit jeweils einer Anschlussmöglichkeit auf (LWL, Kupfer). Meist sind diese Module Mini-GBICs.
GBIC	Anschluss: <i>GigaBit Interface Converter</i> , ermöglicht den Anschluss (LWL, Kupfer) von 1-Gbit/s- oder 10-Gbit/s-Leitungen (wird durch SFP Mini-GBICs abgelöst).
COL, Collision	Anzeige: Kollision
HX, HDX	Anzeige: Halbduplex-Übertragung (ein Partner hört, der andere sendet)
DX, VDX, Duplex	Anzeige: Vollduplex-Übertragung (beide Partner können gleichzeitig senden und empfangen)
FAULT	Anzeige: fehlerhafte Verkabelung bzw. Loop, je nach Ausstattung
Config Port	Anschluss: Netzwerkanschluss für die Konfiguration, wird nicht über die »normalen« Ports erreicht

Tabelle 4.8 Bezeichnung von Anzeigen und Anschlüssen an Switches (Forts.)

4.6.8 Konfiguration eines Switchs allgemein

Am Beispiel des *D-Link DGS-1210-16* können Sie die wichtigsten Schritte verfolgen, mit denen Sie einen Switch für Ihr Netz konfigurieren. Das Beispiel betrifft ein sehr kleines Netz, aber Sie sehen alle notwendigen Arbeiten.

Bauen Sie den Switch in den Netzwerk- oder EDV-Schrank (19-Zoll-Standard) ein, und stellen Sie die Stromversorgung her.

Ersteinrichtung des Switchs

1. Ändern Sie die IP-Adresse so, dass sie zu Ihrem eigenen Netz passt. Für den Fall, dass Sie ein eigenes Administrationsnetz benutzen, passen Sie die Adresse dafür an.
2. Vergeben Sie ein sicheres Administrationskennwort.
3. Stellen Sie anschließend sicher, dass sich nur Berechtigte am Gerät selbst zu schaffen machen können (abschließbarer EDV-Schrank, -Raum).

Diese Arbeiten müssen Sie immer ausführen. Vergessen Sie nicht, die neuen Einstellungen auch zu sichern (links oben Save • Save Configuration). Damit werden die Einstellungen in den Gerätespeicher geschrieben. Sie sollten diese Daten auch außerhalb des Switchs sichern. Dazu gehen Sie auf Tools und wählen hier Configuration Backup & Restore. Klicken Sie auf Backup, und laden Sie damit die Datei *config.bin* auf Ihren Rechner.

4.6.9 Spanning Tree am Switch aktivieren

Mit wenigen Handgriffen haben Sie den Spanning Tree aktiviert. Wählen Sie am Switch Configuration • Spanning Tree • STP Global Settings, und setzen Sie den RSTP Status auf Enabled und STP Version auf RSTP (Abbildung 4.26). Die Bridge-Priorität stellt sich normalerweise von selbst auf den angezeigten Wert ein.

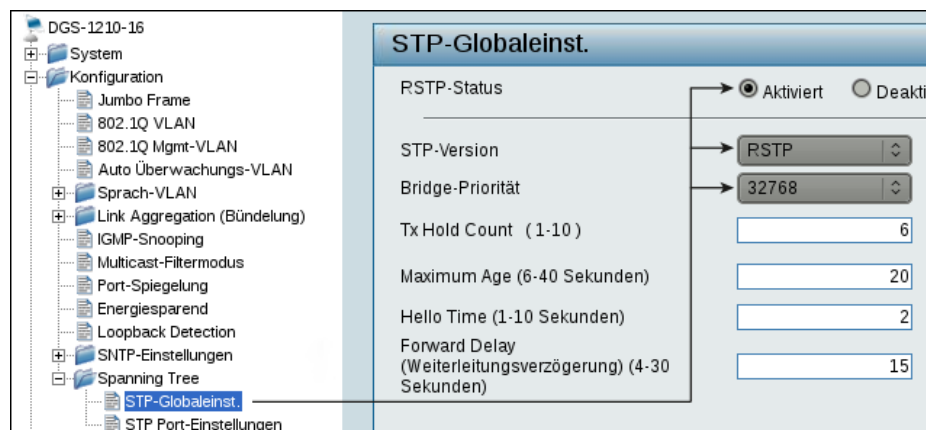


Abbildung 4.26 Aktivieren von Spanning Tree

Im Menü STP Port Settings (Abbildung 4.27) haben Sie die Möglichkeit, die Konfiguration zu verfeinern. Sie sehen auch den Zustand der Anschlüsse: Forwarding oder Blocking. Wenn Sie nun zwei Ports mit einem Patchkabel verbinden, wird der Switch zunächst nach der festgesetzten Zeit (15 Sekunden) den Port prüfen. Stellt er die redundante Netzverbindung fest, wird einer der Ports blockiert und das Netzwerk nicht weiter belastet.

Port	State	Priority	External Cost	Edge	P2P	Restricted Role	Restricted TCN	Port State
01	Enable	128	2000000	Auto	Auto	False	False	Blocking
02	Enable	128	2000000	Auto	Auto	False	False	Blocking
03	Enable	128	2000000	Auto	Auto	False	False	Blocking
04	Enable	128	2000000	Auto	Auto	False	False	Blocking
05	Enable	128	2000000	Auto	Auto	False	False	Forwarding
06	Enable	128	2000000	Auto	Auto	False	False	Forwarding
07	Enable	128	2000000	Auto	Auto	False	False	Forwarding
08	Enable	128	2000000	Auto	Auto	False	False	Blocking
09	Enable	128	2000000	Auto	Auto	False	False	Blocking
10	Enable	128	2000000	Auto	Auto	False	False	Blocking
11	Enable	128	2000000	Auto	Auto	False	False	Blocking
12	Enable	128	2000000	Auto	Auto	False	False	Blocking
13	Enable	128	2000000	Auto	Auto	False	False	Blocking
14	Enable	128	2000000	Auto	Auto	False	False	Blocking
15	Enable	128	2000000	Auto	Auto	False	False	Blocking
16	Enable	128	2000000	Auto	Auto	False	False	Forwarding

Abbildung 4.27 Feinkonfiguration und Statusanzeige für Spanning Tree

4.6.10 VLAN-Konfiguration von Switches

Für das Beispiel habe ich zwei VLANs, 1 und 3, angelegt. Die VLAN-ID 1 funktioniert nicht auf allen Switches. In diesem Fall verwenden Sie eine andere.

Einrichtung der VLANs am Switch

- ▶ VLAN 1 (VID 1): untagged 1–10, tagged: 11
- ▶ VLAN 2 (VID 3): untagged 12–16, tagged: 11
- ▶ Benennen von VID 1 in Netz 1
- ▶ Benennen von VID 3 in Netz 3

Melden Sie sich über den Webbrowser am Switch an, und wählen Sie im linken Funktionsbaum die Einträge Configuration • 802.1Q VLAN. Setzen Sie links oben im Hauptfenster Asymmetric VLAN auf Enabled (Abbildung 4.28).

Abbildung 4.28 Setzen von »Asymmetric VLAN«

Geben Sie VID 1 einen Namen. Klicken Sie dazu auf Rename, und geben Sie den neuen Namen »Netz 1« ein.

Klicken Sie auf den Link von VID 1, und nehmen Sie die Ports 12 bis 16 durch Klicken auf Not Member daraus heraus. Port 11 setzen Sie auf Tag (Abbildung 4.29).

Port	Select All	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16
Untag	All	☒	☒	☒	☒	☒	☒	☒	☒	☒	☒	☒	☐	☐	☐	☐	☐
Tag	All	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☒	☐	☐	☐	☐	☐
Not Member	All	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☒	☒	☒	☒	☒

Abbildung 4.29 Festlegen der Ports für VID 1

Klicken Sie auf Add VID, damit Sie ein weiteres virtuelles Netzwerk, VID 3, mit dem Namen »Netz 3« anlegen können. Tragen Sie die Informationen in die Felder ein, und klicken Sie bei den Ports 12 bis 16 auf Untag sowie bei Port 11 auf Tag. Die fertige Konfiguration sehen Sie in Abbildung 4.30.

VID: 3

VLAN Name: Netz3 (Name should be less than 20 characters)

Port	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16
Untagged	All	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐
Tagged	All	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐
Not Member	All	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐

Cancel Apply

Abbildung 4.30 Anlegen von VID 3

Klicken Sie auf PVID-Settings, und setzen Sie Port 11 auf den Wert 3, wie Abbildung 4.31 zeigt. (Das ist nicht zwingend; es dient hier als Beispiel für Geräte, die nicht selbst *tagged VLAN* unterstützen.)

Port	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16
PVID	1	1	1	1	1	1	1	1	1	1	3	1	1	1	1	1

Cancel Apply

Abbildung 4.31 Setzen der PVID-Einträge (VLAN-Nummer)

Am Ende sollte die VLAN-Konfiguration so wie in Abbildung 4.32 aussehen.

IEEE 802.1Q Asymmetric VLAN Configuration

Asymmetric VLAN [Example] ☒ Enabled ☐ Disabled

(Maximum Entries : 256)

VID	VLAN Name	Untagged VLAN Ports	Tagged VLAN Ports	VLAN Rename	Delete VID
1	Net1	01,02,03,04,05,06,07,08,09,10,11,12,13,14,15,16		Rename	Delete VID
3	Net3	12,13,14,15,16	11	Rename	Delete VID

Apply

Abbildung 4.32 Fertige VLAN-Konfiguration

Sichern Sie die Konfiguration über Save • Save configuration (links oben über den Funktionsbaum).

4.6.11 Konfiguration von Rechnern für tagged VLANs

Anstelle eines (weiteren) Switches können Sie auch einen Rechner an den *tagged VLAN*Port anschließen. Obwohl hier auch IP-Adressen vergeben werden, was thematisch in den nächsten Abschnitt gehört, geht es jetzt aber um die Einstellungen, mit denen Sie ein *tagged VLAN* schaffen. Ein so konfigurierter Rechner kann verschiedene Aufgaben für alle *tagged VLANs* übernehmen, die den Port führen (Datenbank, Plattenspeicher, Anwendungsserver, Internetzugang, Netzwerkverkehrsverschlüsselung usw.). Im Folgenden sehen Sie die Einstellungen für verschiedene Betriebssysteme.

VLAN-Konfiguration unter FreeBSD

Sie brauchen nur die Datei `/etc/rc.conf` wie im Beispiel abzuändern. Der bisherige Eintrag für die Netzwerkkarte wird auskommentiert. Die Netzwerkkarte in diesem Beispiel ist `/dev/ae0`.

```
#ifconfig_ae0="inet 192.168.0.46 netmask 255.255.255.0"
cloned_interfaces="vlan1 vlan3"
ifconfig_vlan1="inet 192.168.0.36 netmask 255.255.255.0 vlan 1 vlandev ae0"
ifconfig_vlan3="inet 192.168.1.36 netmask 255.255.255.0 vlan 3 vlandev ae0"
ifconfig_ae0="up"
ipv6_enable="YES"
defaultrouter="192.168.0.1"
hostname="RECHNERNAME.DOMAIN.TLD"
```

Kontrollieren Sie die Einstellungen. Wenn Sie eine Anzeige wie diese erhalten, können Sie anschließend über die jeweiligen Switch-Ports Verbindung zum Rechner aufnehmen.

Kontrolle der Einstellungen durch Absetzen von »ifconfig« unter FreeBSD

```
$ ifconfig
ae0: flags=8843<UP,BROADCAST,RUNNING,SIMPLEX,MULTICAST> metric 0 mtu 1500
options=82018<VLAN_MTU,VLAN_HWTAGGING,WOL_MAGIC,LINKSTATE>
ether 00:1d:60:75:92:de
inet6 fe80::21d:60ff:fe75:92de%ae0 prefixlen 64 scopeid 0x1
nd6 options=3<PERFORMNUD,ACCEPT_RTADV>
media: Ethernet autoselect (100baseTX <full-duplex>)
status: active
.....
vlan1: flags=8843<UP,BROADCAST,RUNNING,SIMPLEX,MULTICAST> metric 0 mtu 1500
ether 00:1d:60:75:92:de
inet 192.168.0.36 netmask 0xffffffff broadcast 192.168.0.255
inet6 fe80::21d:60ff:fe75:92de%vlan1 prefixlen 64 scopeid 0x6
nd6 options=3<PERFORMNUD,ACCEPT_RTADV>
media: Ethernet autoselect (100baseTX <full-duplex>)
status: active
vlan: 1 parent interface: ae0
vlan3: flags=8843<UP,BROADCAST,RUNNING,SIMPLEX,MULTICAST> metric 0 mtu 1500
ether 00:1d:60:75:92:de
inet 192.168.1.36 netmask 0xffffffff broadcast 192.168.1.255
inet6 fe80::21d:60ff:fe75:92de%vlan3 prefixlen 64 scopeid 0x7
nd6 options=3<PERFORMNUD,ACCEPT_RTADV>
media: Ethernet autoselect (100baseTX <full-duplex>)
status: active
vlan: 3 parent interface: ae0
```

Linux unterliegt einer gewissen Vielfalt. Ich zeige Ihnen deshalb das Vorgehen für Debian-basierte Distributionen (unter anderem Ubuntu), die gerne im Server-Bereich eingesetzt werden. Hier müssen Sie zunächst ein Programmpaket (*vlan*) installieren, bevor Sie mit der Einrichtung der VLANs beginnen können. Ich zeige hier die Methode, VLANs durch Einträge in */etc/network/interfaces* zu konfigurieren.

VLAN-Konfiguration unter Linux (Debian)

Installieren Sie das Paket `vlan`, und richten Sie Ihre VLANs als Benutzer `root` wie folgt ein:

Einträge unter `/etc/network/interfaces`:

```
auto vlan1
iface vlan1 inet static
address 192.168.0.36

netmask 255.255.255.0
vlan_raw_device eth1
```

```
auto vlan3
iface vlan3 inet static
```

```
address 192.168.1.36
netmask 255.255.255.0
vlan_raw_device eth1
```

Setzen Sie anschließend noch folgende Kommandos ab, um die neuen Einträge im laufenden System zu aktivieren:

```
ZE4:/etc/network# ifup vlan1
Added VLAN with VID == 1 to IF -:eth1:-
WARNING:  VLAN 1 does not work with many switches,
consider another number if you have problems.
if-up.d/mountnfs[vlan1]:
  waiting for interface vlan3 before doing NFS mounts (warning).
ZE4:/etc/network# ifup vlan3
Set name-type for VLAN subsystem. Should be visible in /proc/net/vlan/config
Added VLAN with VID == 3 to IF -:eth1:-
```

Das System warnt vor der Vergabe der VID 1. Es gibt einige Switch-Modelle, die hier Probleme verursachen. In diesem Fall ist alles gut gegangen, wie Sie nach der Abfrage erkennen können:

Kontrolle der Einstellungen durch Absetzen von »ifconfig« unter Linux (Debian)

```
ZE4:/etc/network# ifconfig
.....
eth1  Link encap:Ethernet  Hardware Adresse 00:18:e7:16:71:30
      UP BROADCAST MULTICAST  MTU:1500  Metrik:1
      RX packets:0 errors:0 dropped:0 overruns:0 frame:0
      TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
      Kollisionen:0 Sendewarteschlangenlänge:1000
      RX bytes:0 (0.0 B)  TX bytes:0 (0.0 B)
      Interrupt:17 Basisadresse:0x8c00
.....

vlan1  Link encap:Ethernet  Hardware Adresse 00:18:e7:16:71:30
```

```

inet Adresse:192.168.0.36 Bcast:192.168.0.255 Maske:255.255.255.0
UP BROADCAST MULTICAST MTU:1500 Metrik:1
RX packets:0 errors:0 dropped:0 overruns:0 frame:0
TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
Kollisionen:0 Sendewarteschlangenlänge:0
RX bytes:0 (0.0 B) TX bytes:0 (0.0 B)

vlan3 Link encap:Ethernet Hardware Adresse 00:18:e7:16:71:30
inet Adresse:192.168.1.36 Bcast:192.168.1.255 Maske:255.255.255.0
UP BROADCAST MULTICAST MTU:1500 Metrik:1
RX packets:0 errors:0 dropped:0 overruns:0 frame:0
TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
Kollisionen:0 Sendewarteschlangenlänge:0
RX bytes:0 (0.0 B) TX bytes:0 (0.0 B)

```

Unter Microsoft Windows müssen Sie sich eine Netzwerkkarte kaufen, bei der der Hersteller VLAN per Treiber unterstützt. Nach dem Einbau der Baugruppe und der Installation des Treibers können Sie dann die VLANs konfigurieren. Hier hilft Ihnen nur der Blick in die der Karte beiliegende Handbuchdatei.

4.7 Routing – Netzwerkgrenzen überschreiten

Der Weg Ihrer Datenpakete endet gewöhnlich an der Grenze des Netzwerksegments, in dem Sie diese erzeugt haben. Mittels Router überwinden Ihre Datenpakete diese Grenzen und können weltweit ihren Empfänger erreichen.

Router arbeiten auf der Ebene des IP-Protokolls (OSI-Ebene 3) und arbeiten damit mit den IP-Adressen. Diese Geräte kennen sowohl die an sie angeschlossenen Netze als auch ihre Nachbarn. Router tauschen mittels eigener Protokolle untereinander Informationen über Wege und Hosts aus. Sie benutzen dazu auch teilweise das ICMP- bzw. IGMP-Protokoll.

Router lösen den Datenteil des anfragenden Ethernet-Frames heraus. Die Daten werden in einen neuen Ethernet-Frame gesetzt, der die MAC-Adresse des nächsten Routers oder Zielrechners trägt. Die Ziel-IP-Adresse hilft dem Router, die für ihn »nächste« MAC-Adresse zu erfragen. Die Weitervermittlung über einen Router trägt den Namen *Hop*.

Router beherrschen redundante Verbindungen. Sie verwenden stets die schnellste. Im Beispiel in Abbildung 4.33 (linker Teil des Schemabildes) verläuft die schnelle Verbindung über die Router 1 und 3. Die langsamere Verbindung über die Router 2, 4 und 5 ist als Alternative hinterlegt.

Wenn nun Ihre schnelle Verbindung nicht mehr genutzt werden kann, kennen die Router die Ausweichstrecke, hier im Beispiel (Abbildung 4.33, rechter Teil des Schemabildes) über die Router 2, 4 und 5.

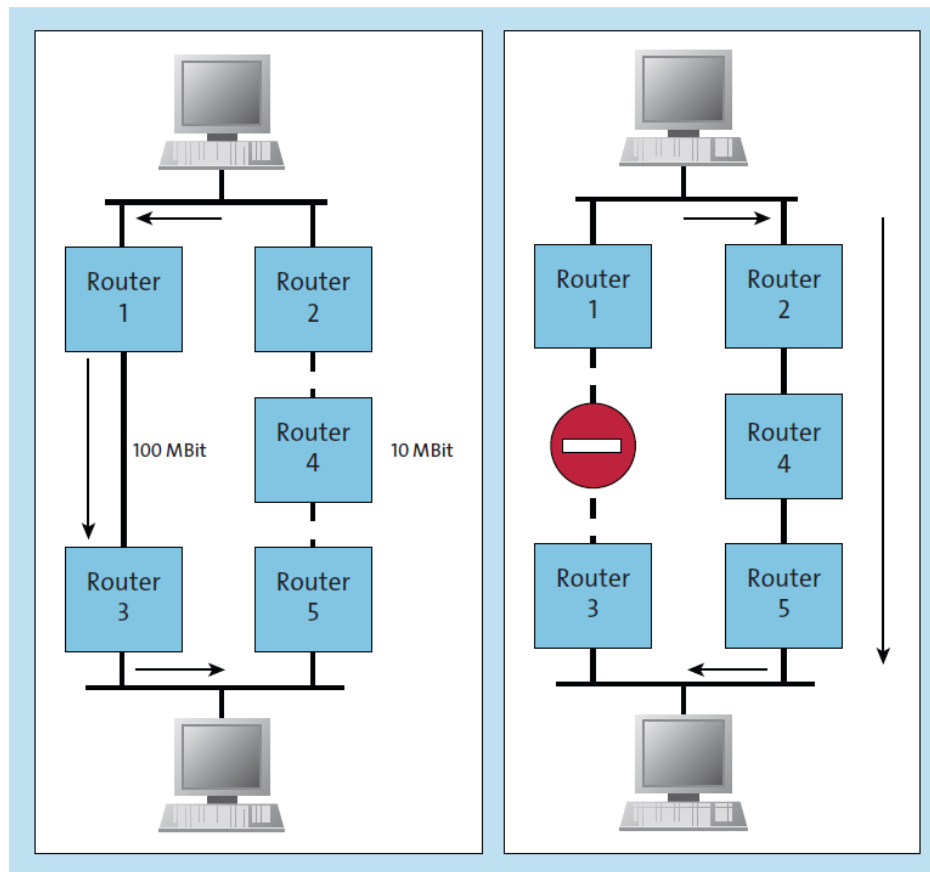


Abbildung 4.33 Links: Router bevorzugen stets die schnellste Verbindung; rechts: aktivierte Ersatzroute.

In der Praxis finden Sie bei der Internetanbindung kleinerer Netze eine ISDN-Anbindung als Notfalllösung vor. Fällt die DSL- oder Mietleitung aus, baut der Router über einen eingebauten ISDN-Anschluss eine »Notverbindung« per Einwahlleitung auf.

Sie können auch PCs als Router konfigurieren. Normalerweise verwenden Sie aber eigens dafür geschaffene Geräte. Bei Privatkunden und Firmen finden Sie oft DSL-Router in den verschiedensten Ausführungen. In manche Geräte ist ein kleiner Switch integriert. Andere Modelle besitzen WLAN-Funkleinrichtungen. Router finden Sie auch in den LAN-Verteilerschrank von Unternehmen, die ihre Filialen untereinander vernetzt haben.

Allgemeines über das Routing

Aufgabe: Datentransport durch ein IP-Netzwerk über Netzwerkgrenzen hinweg

Router tauschen ihre Informationen sowohl mit UDP als auch mit TCP aus.

Routing-Arten:

- ▶ **Dynamisches Routing:** Die Routing-Tabelle ändert sich. Über das Routing-Protokoll werden Erreichbarkeitsdaten zwischen den Routern ausgetauscht.
- ▶ **Statisches Routing:** Feste Vorgaben der Routen; Router kann nicht bei Netzproblemen reagieren und z. B. Ausweichstrecken benutzen.

Wichtige Protokolle:

- ▶ BGP (*Border Gateway Protocol*)
- ▶ IS-IS (*Intermediate System to Intermediate System Protocol*)
- ▶ OSPF (*Open Shortest Path First*)
- ▶ RIP (*Routing Information Protocol*, zum Teil veraltet)

Begriffe:

- ▶ **Standard-Gateway (Standard-Router):** leitet alle Anfragen eines Subnetzes an ein anderes (Sub-)Netz weiter. Gateways im ursprünglichen Sinne verbinden verschiedenartige Netze.
- ▶ **Standard-Route:** Angabe des Standard-Gateways
- ▶ **AS: autonomes System;** von einem Internet-(Infrastruktur-)Provider betreuter Bereich des Internets
- ▶ **Metrik:** Angabe über den Aufwand für eine Verbindung (Hops, Geschwindigkeit usw.)

Über die Eigenschaften der Routing-Protokolle müssen Sie dann informiert sein, wenn Sie Router-Hardware beschaffen. Im Allgemeinen treffen Sie auf Geräte, die auch mit mehreren Protokollen arbeiten können:

BGP: Dieses Protokoll kann gut mit mehreren Verbindungen zu verschiedenen Routing-Domains umgehen (Inter-Domain Routing). Internetbenutzer könnten damit sogar gleichzeitig mit mehreren Internet-Service-Providern verbunden sein. In den RFCs 1195, 2474, 3168, 3260, 5302, 5304, 6232 und 6233 sowie unter www.bgp4.as/ finden

Sie weitere Informationen darüber. BGP4 ist derzeit ein Quasistandard.

IS-IS: Sie finden hier viele Dinge von OSPF. Zusätzlich steht Ihnen hier ein Domänenkonzept zur Verfügung, mit dem Sie hierarchische Netzstrukturen gut bedienen können. In der Berechnung der Metrik finden Sie unter anderem die Werte von Fehlerrate, Leitungskapazität und Laufzeit. Details finden Sie in den RFCs 4271, 4456, 6286, 6608, 6793, 7606, 7607 und 7705.

OSPF: Innerhalb eines autonomen Systems (AS) können Sie mit diesem Protokoll Routing-Aufgaben erledigen lassen (Intra-Domain Routing). Es kommt auch mit hierarchischen Strukturen zurecht. OSPF verwendet stets den kürzestmöglichen Weg (also wenige Hops). Bei mehreren gleichwertigen Verbindungsmöglichkeiten benutzt das Protokoll alle und ermöglicht damit eine Lastverteilung. In den RFCs 2328, 5709, 6549, 6845, 6860 und 7474 finden Sie weitere Details.

RIP: Dieses Protokoll sollten Ihre Router nicht mehr verwenden. OSPF und BGP haben es abgelöst. Die RFCs 2453 und 4822 erläutern Ihnen die Funktionsweise.

Tiefer möchte ich an dieser Stelle nicht in die Theorie der Routing-Protokolle einsteigen. In der Praxis kaufen Sie die Geräte, konfigurieren einige wenige Punkte und haben damit weiter nichts mehr zu tun. Sie führen allenfalls Firmware-Updates durch oder pflegen Konfigurationsänderungen ein.

4.7.1 Gemeinsame Nutzung einer IP-Adresse mit PAT

Ein kleines lokales Netz schließen Sie normalerweise per ISDN-Einwahl oder DSL an das Internet an. Ihr Provider weist Ihnen entweder dauerhaft oder temporär eine IPv4-Adresse zu. Ihre Rechner im lokalen Netz verfügen über private IPv4-Adressen. Nach außen, im Internet, werden sie aber nur durch diese einzige IPv4-Adresse repräsentiert.

Sie möchten vom Internet aus auf einen FTP-Server zugreifen, der auf einem Ihrer Rechner im lokalen Netz läuft. Ihr Router tritt als Stellvertreter dieses Rechners auf und leitet die entsprechenden Pakete an diesen Rechner weiter. Zum Gelingen dieser Aktion trägt *PAT (Port and Address Translation)* bei, auch bekannt unter dem Kürzel *NAPT (Network Address Port Translation)*.

Network Address Translation (NAT) und PAT bieten Ihnen mehr Netzwerksicherheit. Ihre Rechner des lokalen Netzes sind im Internet nicht direkt »sichtbar«. Verschiedene Angriffsmethoden funktionieren deshalb nicht.

Wenn IPv6 flächendeckend eingeführt worden ist, werden Sie normalerweise keine einzelne IP-Adresse mehr mit mehreren Rechnern teilen müssen. Sie werden dennoch einen Router benötigen, da Sie weiterhin im lokalen Netz »private« Adressen verwenden werden.

4.7.2 Festlegen des Standard-Gateways

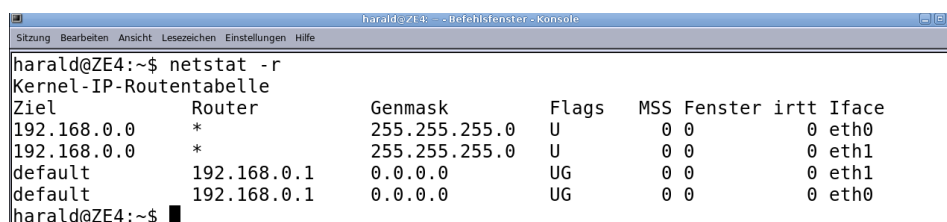
In Abbildung 4.4 in Abschnitt 4.2.1 sehen Sie die manuelle Netzwerkkonfiguration eines Windows-PCs. In diese Maske tragen Sie auch das Standard-Gateway ein, das auch als *Default-Router* bezeichnet wird.

Bei Debian-Linux tragen Sie die Information für die jeweilige Netzwerkkarte in die Datei */etc/network/interfaces* ein (siehe den Abschnitt »Linux« in Abschnitt 4.2.1) ein. Wenn Sie FreeBSD verwenden, setzen Sie den Eintrag unter */etc/rc.conf* (siehe den Abschnitt »FreeBSD« in Abschnitt 4.2.1) ebenfalls bei der Netzwerkkarte (*defaultrouter*). Natürlich können Sie auch *sysinstall* zur Konfiguration verwenden.

Sie können diesen Eintrag (und alle anderen Routen auch) später nach dem vollständigen Rechnerstart ändern. Bei Linux und FreeBSD erstellen Sie dazu einfach ein RCScript, das die neuen Einträge setzt und sie gegebenenfalls beim Herunterfahren oder Wechsel in den Runlevel 1 wieder zurücksetzt. Solange das System nicht »für die Öffentlichkeit« arbeitet, werden seine Anfragen über das Gateway in das »geschützte« Netz geleitet. Sie können dann per Netz Sicherheits-Updates aufspielen oder andere, ebenso sicherheitsrelevante Arbeiten vornehmen.

4.7.3 Routing-Tabelle abfragen (»netstat«)

Sie können mit *netstat* die Routing-Tabellen Ihrer Systeme kontrollieren. Sie kommen damit falschen Einträgen, z. B. für das Default-Gateway, auf die Spur. Dieses Kommando können Sie auf allen gängigen Betriebssystemen absetzen; es unterscheiden sich aber teilweise die Optionen und der Umfang der Ergebnisanzeigen. Ich erkläre hier nur das Abfragen der Routing-Einträge. Weitere Ausführungen zu *netstat* finden Sie in Abschnitt 10.7.2, »Verbindungen mit »netstat« anzeigen«. Geben Sie im Terminal (oder dem »DOS«-Fenster) *netstat -r* ein. In Abbildung 4.34 sehen Sie die Abfrage unter Debian-Linux. Dieselbe Ausgabe erhalten Sie übrigens mit dem Kommando *route -n*.



```

harald@ZE4:~$ netstat -r
Kernel-IP-Routentabelle
Ziel          Router        Genmask       Flags    MSS  Fenster  irtt  Iface
192.168.0.0   *             255.255.255.0 U        0 0           0 eth0
192.168.0.0   *             255.255.255.0 U        0 0           0 eth1
default      192.168.0.1   0.0.0.0       UG       0 0           0 eth1
default      192.168.0.1   0.0.0.0       UG       0 0           0 eth0

```

Abbildung 4.34 Ausgabe »netstat«

Sehen Sie sich die Ausgabe von `netstat` genauer an. Der Rechner hat zwei aktive Netzwerkkarten (`eth0` und `eth1`), die in der rechten Spalte `Iface` aufgeführt werden. Die linke Spalte gibt das Ziel an. Hier finden Sie entweder eine IP-Adresse oder den Eintrag `default`. In diesem Fall wird hierher der gesamte Datenverkehr außerhalb des eigenen Netzes (hier `192.168.0.0`) geschickt. In der zweiten Spalte von links finden Sie die Angabe für den Router. Das Zeichen `»*«` besagt, dass für das angegebene Ziel kein Router notwendig ist (hier lokales Netz). Dafür finden Sie zusätzlich eine Spalte weiter die passende Netzmaske (`255.255.255.0`). Für den Standard-Router finden Sie die IP-Adresse (`192.168.0.1`) sowie den Netzmaskeneintrag (`0.0.0.0`). Letzterer zeigt Ihnen zusätzlich an, dass alle nicht für das weiter oben angeführte lokale Netz bestimmten Pakete an diesen Router gegeben werden. Die Spalte `Flags` gibt Ihnen weitere Auskünfte (Tabelle 4.9). Die weiteren Angaben sind für die Kontrolle der Route ohne Belang.

Flag-Wert	Bedeutung
!	Sämtliche Datagramme werden verworfen.
D	Bezeichnet eine dynamische Route, die mittels ICMP-Nachricht oder eines Routing-Daemons erzeugt wird.
G	Die Route verläuft über ein Gateway.
H	Die Route bezieht sich auf einen einzelnen Host und kein (Teil-)Netz.
M	Der Eintrag wurde durch eine ICMP-Nachricht verändert.
U	aktive Netzwerkschnittstelle

Tabelle 4.9 Flag-Werte der »netstat«-Ausgabe

4.7.4 Routenverfolgung mit »traceroute«

Für die Verfolgung einer Route verwenden Sie `traceroute`. Sie müssen immer das Ziel als IP-Adresse oder Namen angeben. Die Option `-6` prüft die Routen mit IPv6-Adressen. In Abbildung 4.35 sehen Sie zunächst eine Anfrage innerhalb des lokalen Netzes (1 Hop), weiter unten eine nach außerhalb (10 Hops).

```

root@artikel:~# # Traceroute im lokalen Netzwerk
root@artikel:~#
root@artikel:~# traceroute -I fs2000
traceroute to fs2000 (192.168.0.53), 30 hops max, 60 byte packets
 1 fs2000 (192.168.0.53)  0.445 ms  0.482 ms  0.511 ms
root@artikel:~#
root@artikel:~#
root@artikel:~# # Traceroute mit Ziel im Interne
root@artikel:~#
root@artikel:~# traceroute -I www.rheinwerk-verlag.de
traceroute to www.rheinwerk-verlag.de (46.235.24.168), 30 hops max, 60 byte packets
 1 OpenWrt.lan (192.168.0.1)  0.589 ms  0.667 ms  0.744 ms
 2 217.0.116.196 (217.0.116.196)  46.655 ms  47.674 ms  49.617 ms
 3 217.0.76.82 (217.0.76.82)  52.949 ms  54.199 ms  59.069 ms
 4 217.239.37.118 (217.239.37.118)  77.681 ms  82.150 ms  82.277 ms
 5 80.156.162.106 (80.156.162.106)  85.113 ms  96.829 ms  97.103 ms
 6 ae5.scr3.LON3.gblx.net (67.17.72.22)  89.798 ms  90.236 ms  92.479 ms
 7 po4.ar2.FRA4.gblx.net (67.16.130.233)  113.210 ms  82.280 ms  84.738 ms
 8 surfplanet-gmbh.gigabithernet3-2.ar1.fra4.gblx.net (207.138.122.202)  79.288 ms  76.310 ms  78.254 ms
 9 www.rheinwerk-verlag.de (46.235.24.168)  80.289 ms  81.682 ms  83.790 ms
root@artikel:~#

```

Abbildung 4.35 Ausgabe von »traceroute«

4.7.5 Route manuell hinzufügen (»route add«)

Mit dem Befehl `route add` setzen Sie neue Einträge in die Routing-Tabelle des Systemkerns. Sie können sowohl einen Host (`-host <ADRESSE>`) als auch ein ganzes Netzwerk (`-net <ADRESSE>`) als Ziel angeben. Unter Linux und FreeBSD finden Sie weitere Informationen, wenn Sie in der Shell `man route` eingeben. In Abbildung 4.36 sehen Sie das Setzen der Default-Route (Standard-Gateway). Mit der Zusatzangabe `default gw` drücken Sie

dies in diesem Kommando aus.

```

ZE4:~# #### Abfragen Routing-Tabelle ####
ZE4:~# netstat -r
Kernel-IP-Routentabelle
Ziel          Router      Genmask      Flags    MSS  Fenster  irtt  Iface
192.168.0.0    *          255.255.255.0 U        0 0      0  eth0
ZE4:~#
ZE4:~# #### Setzen default-Route auf 192.168.0.1
ZE4:~#
ZE4:~# route add default gw 192.168.0.1
ZE4:~#
ZE4:~# #### Anfragen Routing-Tabelle ####
ZE4:~#
ZE4:~# netstat -r
Kernel-IP-Routentabelle
Ziel          Router      Genmask      Flags    MSS  Fenster  irtt  Iface
192.168.0.0    *          255.255.255.0 U        0 0      0  eth0
default        192.168.0.1 0.0.0.0      UG       0 0      0  eth0
ZE4:~#

```

Abbildung 4.36 Setzen der Default-Route

Eine Route in ein weiteres Netz richten Sie mit der Angabe `-net` ein, wie Abbildung 4.37 zeigt. Hierbei müssen Sie auch die Netzmaske und die Schnittstelle angeben.

```

ZE4:~# #### Kernel-Routing-Tabelle anzeigen ####
ZE4:~#
ZE4:~# netstat -r
Kernel-IP-Routentabelle
Ziel          Router      Genmask      Flags    MSS  Fenster  irtt  Iface
192.168.0.0    *          255.255.255.0 U        0 0      0  eth0
default        192.168.0.1 0.0.0.0      UG       0 0      0  eth0
ZE4:~#
ZE4:~#
ZE4:~# #### Setzen Route für zusätzliches Netzwerk ####
ZE4:~#
ZE4:~# route add -net 192.168.1.0 netmask 255.255.255.0 dev eth0
ZE4:~#
ZE4:~#
ZE4:~# #### Kernel-Routing-Tabelle anzeigen ####
ZE4:~#
ZE4:~# netstat -r
Kernel-IP-Routentabelle
Ziel          Router      Genmask      Flags    MSS  Fenster  irtt  Iface
192.168.1.0    *          255.255.255.0 U        0 0      0  eth0
192.168.0.0    *          255.255.255.0 U        0 0      0  eth0
default        192.168.0.1 0.0.0.0      UG       0 0      0  eth0
ZE4:~#

```

Abbildung 4.37 Setzen einer Route in ein (weiteres) Netz

Sie können auch zu einem einzelnen Host eine Route setzen (Abbildung 4.38). Sie müssen die Netzwerkkarte angeben. In der Ausgabe von `netstat` sehen Sie die Kennzeichnung `H` für den Host.

```

harald@ZE4: ~ - Befehlsfenster - Konsole
Sitzung Bearbeiten Ansicht Lesezeichen Einstellungen Hilfe

ZE4:~# #### Kernel-Routing-Tabelle anzeigen #####
ZE4:~#
ZE4:~# netstat -r
Kernel-IP-Routentabelle
Ziel          Router      Genmask      Flags      MSS  Fenster  irtt  Iface
192.168.0.0    *          255.255.255.0 U          0 0        0 eth0
default       192.168.0.1 0.0.0.0      UG         0 0        0 eth0
ZE4:~#
ZE4:~#
ZE4:~# #### Setzen Host als Routing-Ziel #####
ZE4:~#
ZE4:~# route add -host 192.168.1.1 dev eth0
ZE4:~#
ZE4:~# #### Kernel-Routing-Tabelle anzeigen #####
ZE4:~#
ZE4:~# netstat -r
Kernel-IP-Routentabelle
Ziel          Router      Genmask      Flags      MSS  Fenster  irtt  Iface
192.168.1.1    *          255.255.255.255 UH         0 0        0 eth0
192.168.0.0    *          255.255.255.0 U          0 0        0 eth0
default       192.168.0.1 0.0.0.0      UG         0 0        0 eth0
ZE4:~# █

```

Abbildung 4.38 Setzen einer Route für einen Host

Für die Betriebssysteme FreeBSD und Microsoft Windows weicht die Syntax des `route`-Kommandos ab. Sie finden die Befehle für das Setzen von Routen in Tabelle 4.10 und Tabelle 4.11.

Aktion	Beispiel
Default-Gateway setzen	<code>route add default 192.168.0.1</code>
Routing in ein weiteres Netz; der Router (192.168.1.1) muss mit angegeben werden.	<code>route add -net 192.168.1.0/24 192.168.1.1</code>
Routing an einen einzelnen Host	<code>route add -host 192.168.1.2 192.168.1.1</code>

Tabelle 4.10 FreeBSD: »route add«

Dauerhafte Routeneinträge bringen Sie in der Datei `/etc/rc.conf` unter. Schlagen Sie hierzu im Systemhandbuch von FreeBSD nach.

Aktion	Beispiel
Route anzeigen	<code>route print</code>
Standard-Gateway setzen	<code>route add 0.0.0.0 mask 0.0.0.0 192.168.0.1</code>
Routing in ein weiteres Netzwerk	<code>route add 192.168.1.0 mask 255.255.255.0 192.168.1.1</code>
Dauerhafte Route setzen	<code>route -p add 192.168.1.0 mask 255.255.255.0 192.168.1.1</code>

Tabelle 4.11 Microsoft Windows: »route add«

4.7.6 Route löschen (»route«)

Mit der Option `-del` löschen Sie Routen aus der Tabelle. Sie müssen dabei die vollständigen Angaben zur Route wie beim Setzen anführen (Abbildung 4.39).

Unter FreeBSD löschen Sie mittels `route del -net 192.168.1.0/24` ein Netz oder mit `route del -host 192.168.1.2` einen Host aus der Routing-Tabelle.

```

harald@ZE4: ~ - Befehlsfenster - Konsole
Sitzung Bearbeiten Ansicht Lesezeichen Einstellungen Hilfe

ZE4:~# #### Kernel-Routing-Tabelle anzeigen #####
ZE4:~#
ZE4:~# netstat -r
Kernel-IP-Routentabelle
Ziel          Router      Genmask      Flags      MSS  Fenster  irtt  Iface
192.168.1.1    *          255.255.255.255 UH         0  0      0  eth0
192.168.1.0    *          255.255.255.0  U         0  0      0  eth0
192.168.0.0    *          255.255.255.0  U         0  0      0  eth0
default       192.168.0.1 0.0.0.0      UG         0  0      0  eth0
ZE4:~#
ZE4:~#
ZE4:~# #### Löschen des Host-Eintrages in der Routing-Tabelle
ZE4:~#
ZE4:~# route del -host 192.168.1.1 dev eth0
ZE4:~#
ZE4:~# #### Löschen des Netz-Eintrages in der Routing-Tabelle
ZE4:~#
ZE4:~# route del -net 192.168.1.0 netmask 255.255.255.0 dev eth0
ZE4:~#
ZE4:~#
ZE4:~# #### Kernel-Routing-Tabelle anzeigen #####
ZE4:~#
ZE4:~# netstat -r
Kernel-IP-Routentabelle
Ziel          Router      Genmask      Flags      MSS  Fenster  irtt  Iface
192.168.0.0    *          255.255.255.0  U         0  0      0  eth0
default       192.168.0.1 0.0.0.0      UG         0  0      0  eth0
ZE4:~# █

```

Abbildung 4.39 Löschen von Routen (zu einem Host und einem Netzwerk)

Bei Microsoft Windows geben Sie Folgendes ein:

```
route del 192.168.1.0 mask 255.255.255.0
```

Der route-Befehl von FreeBSD und Windows kennt einen Änderungsmodus (change). Damit ändern Sie Angaben in der Routing-Tabelle.

4.8 Multicast-Routing

Software-Updates per Internet, Multiplayer-Spiele, Musik- und Videoübertragungen fordern eine Abkehr von den 1:1-Verbindungen. Hier müssen Sie die gleichen Pakete an viele Empfänger übertragen. Damit es im Netz keinen Stau wegen unzähliger Routing-Anweisungen gibt, verwenden Sie das Multicast-Routing.

Für jedes abgehende Datenpaket müssen Sie mehrere Empfänger angeben. In der Praxis bekämen Sie mit der praktischen Handhabung aber Probleme. Ihr Rechner hätte große Probleme, die Adressen der anfragenden Hosts samt Routing-Anweisungen sicher unterzubringen. Zudem sollen verloren gegangene Pakete keinesfalls wiederholt werden, damit es z. B. bei Videoübertragungen nicht bei allen Teilnehmern »ruckelt«, nur weil einige von ihnen eine langsame oder schlechte Internetanbindung haben.

Datenschutzgründe

stehen der Verwendung von Einzeladressen ebenfalls entgegen.

Es ist Sache spezieller Multicast-Router, die untereinander in Verbindung stehen, die Pakete weiterzuleiten. Beachten Sie hierbei auch die Reservierung spezieller Adressen, die in Tabelle 4.11 in Abschnitt 3.5.7, »Weitere reservierte Adressen«, aufgelistet sind.

Mittels des *Internet Group Management Protocols (IGMP)* werden Ihre Datenpakete an Multicast-Router gesendet, die Empfängergruppen verwalten. Sie benötigen damit wenige Zieladressen. Eine Kommunikation zwischen den Mitgliedern einer Empfängergruppe ist nicht möglich. Sie bleiben untereinander anonym. Das IGMP kennt nur drei Nachrichtentypen: die *membership_query*, als Antwort darauf den *membership_report* und *leave_group*. Mit der *membership_query* fragen Router bei einem Host an, in welcher Multicast-Gruppe er Mitglied ist. Die Antwort sendet der Host mit dem *membership_report*. Diesen sendet er auch ohne Anfrage, wenn er einer Multicast-Gruppe beitreten möchte. Möchte Ihr Host eine Multicast-Gruppe verlassen, sendet er *leave_group*.

Das MBone ist hierzulande das bekannteste virtuelle Multicast-Netzwerk. Für die weitere Vertiefung dieses Themas lesen Sie die RFCs **1112**, 2003, 2236, 3168, 3376, 4301, 4604, 6040, 6864 und 7619.

4.9 Praxisübungen

Hier finden Sie zusammenfassende Aufgaben, mit denen Sie Ihr Wissen überprüfen und gegebenenfalls auch vertiefen können. Die Aufgaben behandeln den gesamten bisherigen Stoff, also von den physikalischen Verbindungen bis hin zur Namensauflösung

4.9.1 Glasfasern

1. Sie vernetzen Ihre Filiale mit dem Hauptbetrieb per Glasfaser. Zwischen den beiden Orten liegen ca. 8 km. Welche Faser verwendet der Vermieter Ihrer Standleitung?
2. Was passiert, wenn Sie den vom Kabelhersteller vorgegebenen Biegeradius Ihres LWL-Kabels unterschreiten?
3. Welche Vorteile bieten Ihnen Glasfasern gegenüber der Kupferverkabelung?
4. An beiden Enden einer bisher funktionierenden LWL-Strecke schließen Sie mittels SC-Duplex-Steckverbindern Medienkonverter an. Die Kontroll-LEDs für den Link leuchten nicht. Die Medienkonverter selbst sind fehlerfrei. Wie stellen Sie eine funktionierende Verbindung her?

4.9.2 TP-Verkabelung

1. Welche Kabelkategorie verbauen Sie bei Netzerweiterungen?
2. Sie renovieren Büroräume. Für die Stromversorgung bestehen bereits Kabelkanäle aus Kunststoff, in denen die Stromkabel liegen. Dürfen Sie diese Kanäle für die Netzkabel (Cat. 7) mitbenutzen?
3. Was passiert, wenn Sie die maximale Länge einer Kabelstrecke überschreiten?

4.9.3 Switches

1. Wie konfigurieren Sie Ihren Switch, damit er trotz bestehender Schleife weiter betriebsfähig bleibt?
2. Was bedeutet die Angabe Auto-MDI(X) an einem Switch?
3. Welche Vorteile bringt die Verwendung von VLANs?

4.9.4 MAC- und IP-Adressen

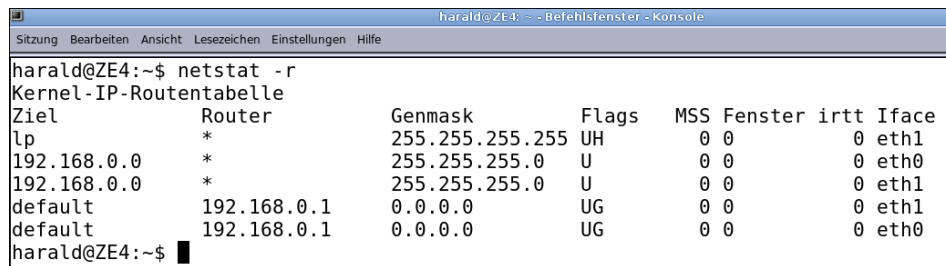
1. Was benötigen Sie, um in einem Netz jedem Host schon während seines Boot-Vorgangs eine IP-Adresse zuzuweisen?
2. Mit welchem Parameter legen Sie damit die Gültigkeit einer Adresszuweisung fest?

4.9.5 Namensauflösung

1. Mit welcher Maßnahme setzen Sie für den jeweiligen Rechner gültige »Websperren« zur Erhöhung der Sicherheit?
2. Mit welchem Record-Typ versehen Sie eine IPv6-Adresse in einer Zonendatei?
3. Mit welcher Maßnahme stellen Sie sicher, dass bei der Namensauflösung auf einem Linux- oder FreeBSD-Rechner zuerst die Datei */etc/hosts* und erst anschließend der Nameserver bemüht wird?

4.9.6 Routing

1. Was verstehen Sie unter einem Default-Router oder einem Standard-Gateway?
2. Erklären Sie die folgende Abfrage der Routing-Tabelle in Abbildung 4.40.



```

harald@ZE4:~$ netstat -r
Kernel-IP-Routentabelle
Ziel          Router      Genmask      Flags        MSS  Fenster  irtt  Iface
lp            *          255.255.255.255 UH           0    0        0    eth1
192.168.0.0   *          255.255.255.0  U           0    0        0    eth0
192.168.0.0   *          255.255.255.0  U           0    0        0    eth1
default       192.168.0.1 0.0.0.0      UG           0    0        0    eth1
default       192.168.0.1 0.0.0.0      UG           0    0        0    eth0
harald@ZE4:~$

```

Abbildung 4.40 Abfrage einer Routing-Tabelle

3. Wie lautet Ihre Eingabe zum Löschen der Route zum Host lp?

4.9.7 Sicherheit im lokalen Netz

1. Kann die MAC-Adresse zur eindeutigen Identifizierung eines Netzwerkteilnehmers herangezogen werden?

2. Was zeigt Ihnen diese Programmausgabe?

```

ZE4:~# arp -a
192.168.0.12 auf 00:c0:ee:83:7f:0d [ether] auf eth0
192.168.0.18 auf 00:c0:ee:83:7f:0d [ether] auf eth0
192.168.0.40 auf 00:c0:ee:83:7f:0d [ether] auf eth0
192.168.0.42 auf 00:c0:ee:83:7f:0d [ether] auf eth0
192.168.0.44 auf 00:c0:ee:83:7f:0d [ether] auf eth0

```

3. Sie haben leicht zugängliche, meist unbenutzte Räume in Ihrem Firmengebäude, die mit Netzwerkdosen ausgestattet sind. Mit welchen Maßnahmen verhindern Sie, dass sich Unbefugte physikalisch mit Ihrem Netzwerk verbinden?

4. Wie erhöhen Sie die Sicherheit beim NDP (IPv6) bezüglich Spoofing-Attacken?

5. Mit welcher vorbeugenden Maßnahme können Sie Ihre wichtigen Adressen im Internet trotz DNS-Ausfall erreichen?

5. Steuer- und Fehlercodes mit ICMP

und ICMPv6 übertragen

Während TCP und UDP der reinen Datenübertragung dienen, helfen Ihnen die Protokolle ICMP und ICMPv6 beim Übermitteln von Steuerungs- und Fehlermeldungen. Vertiefende Informationen finden Sie in RFC 792 (Updates in den RFCs 950, 4884, 6633 und 6918) sowie in RFC 1122 (Updates in den RFCs 1349, 4379, 5884, 6093, 6298, 6633 und 6864).

ICMP-Pakete erkennen Sie am Servicetyp 0 und an der Protokollnummer 1 im IPv4-Header. Beim ICMPv6 lautet die Protokollnummer 58.

Sicher ist Ihnen `ping` vertraut, mit dem Sie IP-Netzwerkverbindungen überprüfen (siehe Abschnitt 4.4.1, »Prüfung der Erreichbarkeit und Namensauflösung mit ›ping‹ bzw. ›ping6‹«, und Abschnitt 10.7.8, »Die Erreichbarkeit von Hosts mit ›ping‹ bzw. ›ping6‹ prüfen«). Hier arbeiten Sie selbst mit ICMP. Sonst benutzen Rechner und Router dieses Protokoll ohne Ihr Zutun. Bestimmte Anwendungen arbeiten ebenfalls mit ICMP und lassen Ihnen – je nach Situation – Status- und Fehlermeldungen (im Klartext) zukommen. ICMP-Meldungen können Ihnen beim Beheben von Netzwerkproblemen behilflich sein. Egal, ob IPv4 oder IPv6, Sie finden immer den gleichen Aufbau bei den Nachrichten vor (Tabelle 5.1). Beachten Sie hierbei eine Grundregel: Eine ICMP-Nachricht löst keine weitere Reaktion aus. Die Ausnahme hiervon finden Sie mit `ping (echo request)`.

8 Bit	8 Bit	16 Bit
Typ	Code	Prüfsumme
ICMP-Nachricht/Daten		

Tabelle 5.1 Aufbau von ICMP-Nachrichten

Für einen Eintrag im Feld `Typ` finden Sie ein oder mehrere zugehörige Einträge bei `Code`.

5.1 ICMP-Pakete (IPv4)

Sie finden in Tabelle 5.2 einige Einträge für das Feld `Typ` und die Codes dazu dargestellt.

Typ	Name	Code	Nachricht
0	Echo-Antwort	0	Echo-Antwort
8	Ping Echo-Anfrage	0	Echo-Anfrage
3	Ziel unerreichbar	0	Netzwerk nicht erreichbar
		1	Host nicht erreichbar
		2	Protokoll nicht erreichbar
		3	Port nicht erreichbar
		4	Fragmentierung benötigt, aber nicht erlaubt (DON'T FRAGMENT gesetzt)
		5	Source-Route fehlgeschlagen
		6	Zielnetzwerk unbekannt
		7	Ziel-Host unbekannt
		8	Quell-Host isoliert
		9	Kommunikation mit dem Zielnetzwerk ist verboten.
		10	Kommunikation mit dem Ziel-Host ist verboten.
		11	Zielnetzwerk ist für den angegebenen Service-Typ nicht erreichbar.
		12	Ziel-Host ist für den angegebenen Service-Typ nicht erreichbar.
		13	Kommunikation verboten (durch Firewall)
9	Router Advertisement	0	Normal Router Advertisement
		16	routet keinen allgemeinen Verkehr
10	Router Selection	0	(Router-Auswahl)
30	Traceroute		Traceroute

Tabelle 5.2 Auswahl von ICMP-Nachrichten

Eine komplette Übersicht hierfür erhalten Sie unter www.iana.org/assignments/icmpparameters. In dieser Aufstellung finden Sie auch alle zugehörigen RFCs.

5.2 ICMPv6-Pakete

Das Format der ICMP-Pakete für IPv6 entspricht zwar jenen des IPv4. Doch finden Sie hier eine andere Nummerierung der Typen vor. Zusätzlich begegnen Ihnen neue Arten von Typen und die Unterscheidung in Fehler- und Informationsnachrichten. Die Fehlernachrichten finden Sie in Tabelle 5.3, einige Informationsnachrichten in Tabelle 5.4.

Typ	Name	Code	Nachricht
1	Destination unreachable	0	keine Route zum Ziel
		1	Kommunikation mit Ziel untersagt (z. B. durch Firewall)
		2	Zieladresse außerhalb des Bereichs der Quelladresse
		3	Adresse nicht erreichbar
		4	Port unerreichbar
		5	Quelladresse verletzt Ein-/Ausgangsregeln.
		6	Routing zum Ziel verweigert
		7	Fehler im Quell-Routing-Header
2	Packet too big	0	(zulässige Paketgröße überschritten)
3	Time exceeded	0	Hop-Limit beim Durchgang überschritten
		1	Zeit zum Zusammensetzen von Fragmenten überschritten
4		0	fehlerhaftes Header-Feld entdeckt
		1	unerkannter Next-Header-Typ entdeckt
		2	unerkannte IPv6-Option entdeckt

Tabelle 5.3 Fehlernachrichten des ICMPv6

In Tabelle 5.4 finden Sie einige Informationsnachrichten. Die darin aufgeführten Typen 133 bis 137 sind auch in Tabelle 3.4 in Abschnitt 3.3.2, »Neighbor Discovery Protocol (NDP), IPv6«, genauer behandelt. Der Code-Eintrag lautet für die meisten Einträge 0.

Typ	Name
128	Echo Request (ping)
129	Echo Reply
130	Multicast Listener Query
131	Version 1 Multicast Listener Report
132	Multicast Listener Done
133	Router Solicitation
134	Router Advertisement
135	Neighbor Solicitation
136	Neighbor Advertisement
137	Redirect
138	Router Renumbering
139	ICMP Node Information Query
140	ICMP Node Information Response
141	Inverse Neighbor Discovery Solicitation Message
142	Inverse Neighbor Discovery Advertisement Message
143	Version 2 Multicast Listener Report
144	Home Agent Address Discovery Request Message
145	Home Agent Address Discovery Reply Message
146	Mobile Prefix Solicitation
147	Mobile Prefix Advertisement
151	Multicast Router Advertisement
152	Multicast Router Solicitation
153	Multicast Router Termination

Tabelle 5.4 Informationsnachrichten des ICMPv6

Für die Typen 138, 139 und 140 finden Sie in Tabelle 5.5 die möglichen Code-Einträge.

Type	Code	Nachricht
138	0	Router Renumbering Command
	1	Router Renumbering Result
	255	Sequence Number Reset
139	0	Datenfeld enthält eine IPv6-Adresse.
	1	Datenfeld enthält einen Host-Namen.
	2	Datenfeld enthält eine IPv4-Adresse.
140	0	Antwort erfolgreich, das Datenfeld kann leer oder gefüllt sein.
	1	Antwort wurde verweigert.
	2	Fragetyp (s. 139, 0–2) unbekannt

Tabelle 5.5 Code-Einträge ausgewählter ICMPv6-Nachrichten

Die komplette Übersicht über die ICMPv6-Nachrichten können Sie unter www.iana.org/assignments/icmpv6-parameters nachlesen. Dort sind auch die dazugehörigen RFCs mit angegeben.

6. Datentransport mit TCP und UDP

Der weltweite Datentransport geschieht überwiegend innerhalb von IP-Netzen. Ihre Daten werden hier in Pakete zerlegt und am Zielort wieder zusammengesetzt. Den dabei entstehenden Datenstrom können Sie kontrolliert per TCP übertragen. Damit entlasten Sie die Anwendung, die die Korrektheit und Vollständigkeit überprüfen müsste. Natürlich erkaufen Sie sich diese Sicherheit mit erhöhter Netzlast. Diese senken Sie dadurch, dass Sie UDP zur Datenübertragung einsetzen. In diesem Fall sendet Ihr Rechner auf gut Glück die Datenpakete ins Netz. Die jeweils kommunizierenden Anwendungen kontrollieren den Fluss, die Korrektheit und die Vollständigkeit der Daten selbst – und verursachen sowohl beim Sender als auch beim Empfänger eine höhere Rechnerauslastung.

Die TCP- oder UDP-Datagramme liegen im Datenbereich der IP-Pakete. Egal, welches Transportprotokoll Sie benutzen, Ihre Daten brauchen jeweils einen »Hafen« für den Start und das Ziel ihrer Reise, die Ports.

6.1 Transmission Control Protocol (TCP)

Mit TCP benutzen Sie ein verbindungsorientiertes Übertragungsprotokoll für Ihre Datenpakete. Verbindungen werden förmlich auf- und abgebaut, die Datenübertragung selbst unterliegt einer Flusskontrolle. Durch diese Flusskontrolle gehen keine Daten verloren. Unvollständige oder fehlende Pakete werden von der Gegenstelle bemerkt und erneut angefordert. Der Empfänger-Rechner setzt die Datenpakete in ihrer richtigen Reihenfolge wieder zusammen und übergibt sie der Anwendung.

Die TCP-Verbindung kann von den beiden Hosts gleichberechtigt in jede Richtung genutzt werden. Insoweit wird es für Sie manchmal schwierig sein, mitprotokollierten Netzwerkverkehr richtig zu deuten. Die beim TCP verwendeten Ports bilden zusammen mit der IP-Adresse die *Sockets*.

6.1.1 Das TCP-Paket

Das TCP-Paket, auch *TCP-Datagramm* genannt, beinhaltet keine Absender- oder Ziel-IP-Adresse. Vielmehr finden Sie nur die *Port-Nummern* von Sender und Empfänger vor.

Die IP-Adressen der beteiligten Hosts sind bereits im *IP-Header* angegeben, in dem das Datagramm im Datenteil abgelegt ist.

TCP könnten Sie auch innerhalb anderer Netzwerkprotokolle verwenden. Beim IP-Protokoll gibt Ihnen der Ethernet-Frame einen Nutzlastanteil von 1500 Byte vor. Hiervon müssen Sie aber noch den IP-Header mit 20 Byte und den TCP-Header mit ebenfalls 20 Byte abziehen. Unter Umständen benutzen Sie zusätzlich eine Übertragungsmethode wie PPP oder PPPoE (DSL). In diesem Fall gehen davon nochmals 8 Byte verloren. Es verbleiben

1460 oder nur 1452 Byte für Ihre Daten. Diesen Wert nennt man die *Maximum Segment Size (MSS)*. Auf diesen Wert können sich Sender und Empfänger beim Verbindungsaufbau

einigen. Hierzu belegen sie das Optionsfeld mit diesem Parameter. Den Aufbau des TCP-Datagramms finden Sie in Abbildung 6.1.

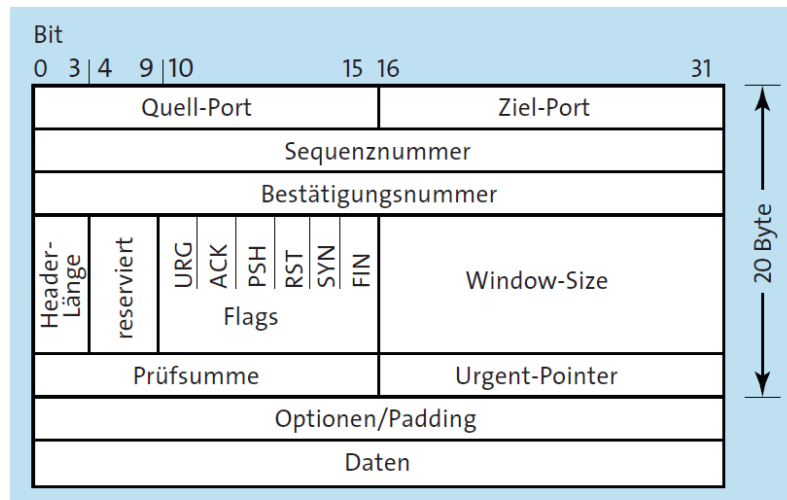


Abbildung 6.1 Aufbau des TCP-Datagramms

Die Bedeutung der einzelnen Felder habe ich in Tabelle 6.1 erklärt.

Feld	Inhalt	Größe (Bits)
Quell-Port	Port-Nummer der sendenden Anwendung	16
Ziel-Port	Port-Nummer der empfangenden Anwendung	16

Tabelle 6.1 TCP-Header

Feld	Inhalt	Größe (Bits)
Sequenznummer	Nummer des Datenpakets innerhalb des Datenstroms. Sie darf während der Verbindung nur einmal vorkommen.	32
Bestätigungsnummer	Bestätigung eines Pakets oder Blocks	32
Header-Länge	Header-Länge, angegeben mit der Zahl der benutzten 32-Bit-Blöcke. Damit wird der Beginn der Nutzdaten beschrieben.	4
reserviert	unbenutzt	6
Flags	1-Bit-Kennzeichen (Werte 1 oder 0) mit folgenden Bedeutungen:	-6
	URG: kennzeichnet die Einleitung einer Steueranweisung. Alle Pakete ab diesem Flag werden direkt an die Anwendung weitergereicht und nicht gepuffert. Das Ende einer Steueranweisung setzt der <i>Urgent-Pointer</i> . In der Praxis brechen Sie z. B. damit Datenübertragungen oder Kommandos ab.	1
	ACK: Acknowledgement. Das ACK-Flag bestätigt die Gültigkeit der Bestätigungsnummer (ACK-Number). Damit bestätigt TCP den vollständigen und richtigen Empfang eines Pakets (oder eines Blocks mit mehreren Paketen).	1
	PSH: So gekennzeichnete Pakete übergehen den Empfangspuffer und gelangen direkt zur Anwendung. Damit vermeiden Sie z. B. »Hänger« bei Fernsitzungen.	1
	RST: Abbruch von bestehenden Verbindungen, Abweisen von unerwünschten Verbindungen (z. B. geschlossener Port). Im Gegensatz zu UDP sendet der Empfänger kein Port-Unreachable-ICMP-Paket an den Absender.	1
	SYN: Anzeigen des Verbindungsaufbaus und damit verbunden Synchronisieren der Sequenznummer. Der Empfänger bestätigt den Verbindungsaufbau mit SYN + ACK. Er kann den Verbindungsaufbau mit RST ablehnen.	1

Tabelle 6.1 TCP-Header (Forts.)

Feld	Inhalt	Größe (Bits)
	FIN: Schlusszeichen einer Übermittlung. Der Sender zeigt das Ende der Daten an und gibt die Verbindung frei.	1
Window-Size	Anzahl der Bytes, die der Sender übermitteln darf, ohne ein Überlaufen des Empfangspuffers hervorzurufen.	16
Prüfsumme (Checksum)	Quersumme zum Prüfen der Unversehrtheit der Pakete	16
Urgent-Pointer	Angabe der Position des letzten Bytes der Urgent-Daten (Flag URG) im Datenstrom. Das Flag URG muss gleichfalls gesetzt sein, damit diese Anweisung gültig ist.	16
Optionen/Padding	Übergabe von Optionen. Damit dieser Bereich immer 32 Bit Länge aufweist, wird der ungenutzte Platz mit »0« aufgefüllt (<i>Padding</i>).	32
Daten	die Nutzdaten für die empfangende Anwendung	–

Tabelle 6.1 TCP-Header (Forts.)

Weitere vertiefende Informationen finden Sie in den RFCs **793**, 1122, 1349, 3168, 4379, 5884, 6093, 6298, 6528, 6633, 6864 und 7323.

6.1.2 TCP: Verbindungsaufbau

Zunächst sendet Host 1 ein Anfragepaket an Host 2 (SYN). Im Beispiel der Abbildung 6.2 nimmt er die Anfrage an und sendet seine Zustimmung als SYN ACK-Paket zurück. Host 1 bestätigt wiederum mit einem ACK den Erhalt und beginnt mit der Übermittlung der Datenpakete. Schon beim Verbindungsaufbau kommen beidseitig Sequenznummern zum Einsatz.

Diese Sequenznummern bildet jeder Host für sich ($SEQ = X$, $SEQ = Y$). Bei der ersten Antwort sendet Host 2 die um den Wert 1 erhöhte Sequenznummer als Bestätigungsnummer (ACK, ACK-Nummer) zusammen mit seiner eigenen Sequenznummer zurück. Host 1 bestätigt den Erhalt, indem er mit ACK und der um 1 erhöhten Sequenznummer des Hosts 2 als Bestätigungsnummer antwortet. Gleichzeitig erhöht er den Wert seiner eigenen Sequenznummer und übermittelt diese ebenfalls. Die Verbindung ist aufgebaut.

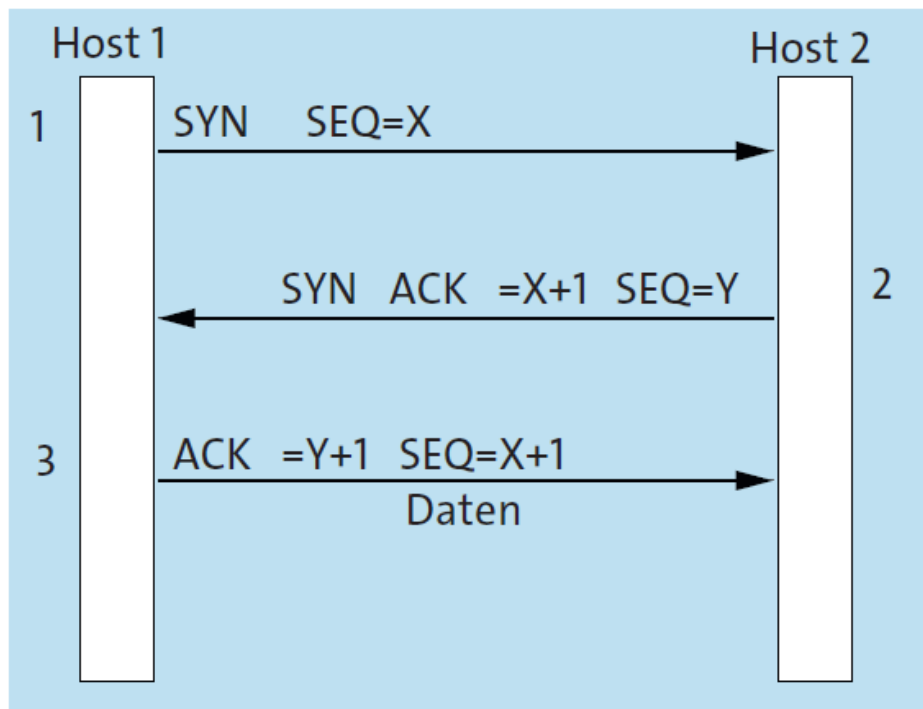


Abbildung 6.2 TCP: Verbindungsaufbau

Einen Verbindungswunsch kann ein Host durch Senden eines Pakets mit dem RST-Flag ablehnen.

6.1.3 TCP: Transportkontrolle

Das Datenvolumen kann größer sein als der für die Nutzlast bemessene Speicherplatz. Der sendende Host teilt die Daten paketweise auf (Segmentierung). Jedes Segment bekommt einen eigenen TCP-Header und wird an den Empfänger übermittelt. Dieser setzt aus den einzelnen Paketen die Daten in der Reihenfolge der Sequenznummern wieder zusammen. Anschließend übergibt TCP die kompletten Daten der anfordernden Anwendung.

Nach dem erfolgreichen Verbindungsaufbau sendet Host 1 seine Daten an Host 2 (Abbildung 6.3). Für jedes erfolgreich übermittelte Paket antwortet Host 2 mit einem ACK-Paket, das auch die Bestätigungsnummer (= Sequenznummer des erhaltenen Pakets) beinhaltet.

Im Beispiel aus Abbildung 6.3 sehen Sie zunächst den Ablauf der Einzelquittierung. Bei der reinen Datenübermittlung kommt diese kaum zum Einsatz, da sie das Netz zusätzlich belastet. Die gebräuchliche Blockquittierung erlaubt es dem Sender, mehrere Pakete zu übermitteln, wenn diese miteinander zusammenhängen. Der Empfänger gibt in seinem ACK-Paket als Bestätigungsnummer die letzte empfangene Sequenz zurück. Tritt bei der Übermittlung ein Fehler auf, erkennt der Sender nach Ablauf eines Timers das Fehlen von unbestätigten Sequenzen (*Retransmission Timer*). Er sendet diese Pakete nochmals ab.

Mit der Prüfsumme werden Unregelmäßigkeiten innerhalb der Pakete erkannt. Für die Berechnung der Prüfsumme zieht TCP nur folgende Daten heran:

- die IP-Adresse des Senders
- die IP-Adresse des Empfängers
- das Leerfeld 00000000
- die Protokoll-ID (6)
- die Länge des TCP-Headers mit den Nutzdaten in Bytes

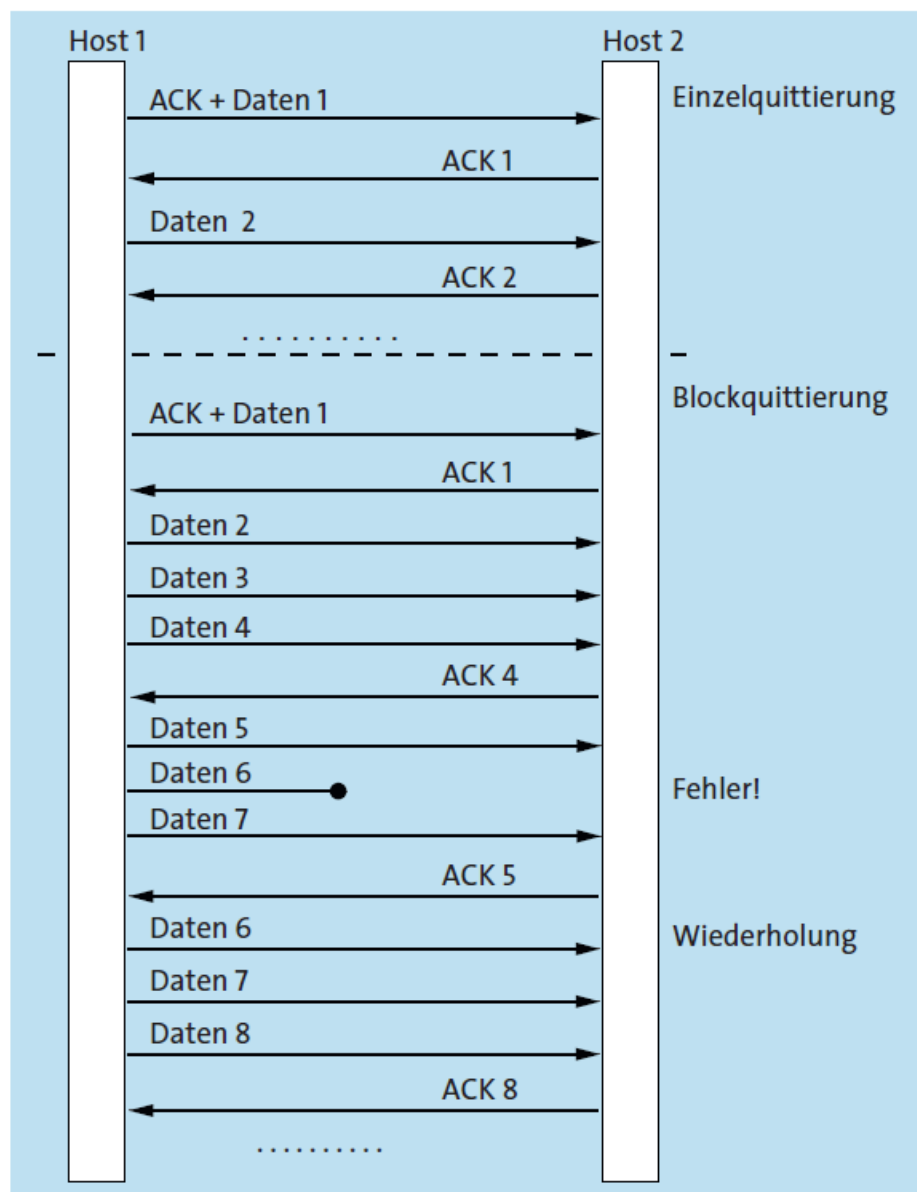


Abbildung 6.3 TCP: Datenübertragung

6.1.4 TCP: Verbindungsabbau

Liegen keine zu übermittelnden Daten mehr vor, baut der Sender die TCP-Verbindung wieder ab. Auch dies unterliegt einem eigenen Formalismus. Der sendende Host 1 sendet ein Paket mit dem FIN-Flag, das der Empfänger-Host 2 bestätigt. Dieser (Host 2) erklärt daraufhin seinerseits mittels eines FIN-Pakets die Verbindung für beendet. Host 1 bestätigt den Erhalt mittels ACK. Die TCP-Verbindung zwischen den beiden Rechnern ist damit aufgehoben (Abbildung 6.4).

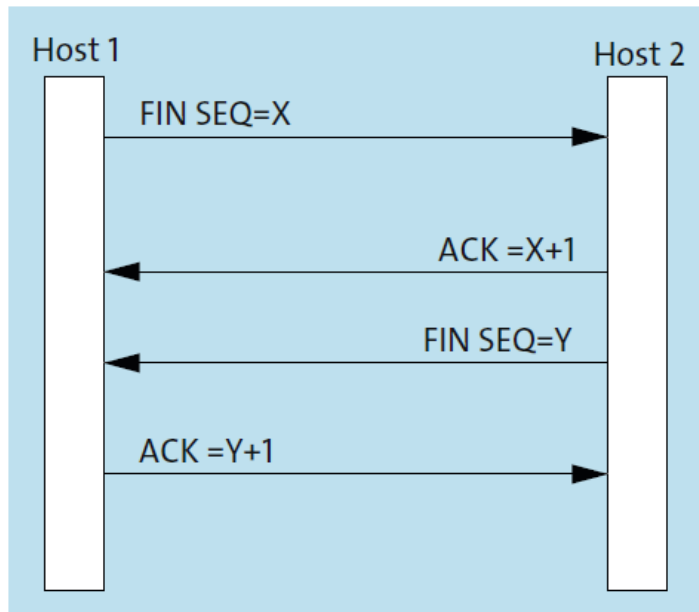


Abbildung 6.4 TCP: Aufhebung der Verbindung

Es ist notwendig, dass sich beide Hosts das Verbindungsende gegenseitig erklären. TCP-Verbindungen sind bidirektional.

Eine weitere Darstellung einer TCP-Verbindung im Zusammenhang mit einem Sitzungsprotokoll (HTTP) finden Sie in Abschnitt 7.3.1, »Grundlagen des HTTP-Protokolls«.

6.2 User Datagram Protocol (UDP)

UDP arbeitet ohne das Einleiten, Überwachen und Beenden einer Verbindung. Die Datenpakete gehen auf gut Glück auf die Reise. Die beteiligten Anwendungen sind für die Fehlererkennung und -behandlung selbst verantwortlich.

Es entfällt alles an Protokoll-Overhead, was einer schnelleren Datenübermittlung zugutekommt. Der Verzicht auf jegliche Verbindungs- und Transportkontrolle ermöglicht aber auch Anwendungen, die mittels TCP nicht zu verwirklichen sind, z. B. Ton- und Videoübertragungen per Internet. Gerade hier kommt es nicht darauf an, dass alle Pakete auf ihre Richtig- und Vollständigkeit hin »abgesegnet« werden. Es ist nicht weiter schlimm, wenn das Video mal ein bisschen ruckelt, solange dafür Hunderte von Teilnehmern es zeitgleich empfangen können.

6.2.1 UDP: Der UDP-Datagram-Header

In Abbildung 6.5 sehen Sie den UDP-Datagram-Header. Er besitzt nur wenige Felder, die Sie in Tabelle 6.2 aufgeführt finden.

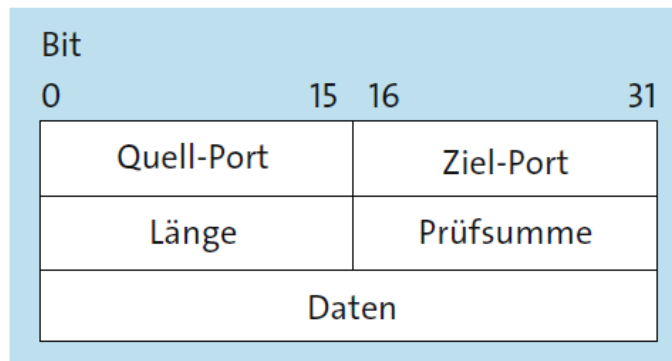


Abbildung 6.5 UDP: Header

Feld	Inhalt	Größe (Bits)
Quell-Port	Port-Nummer der sendenden Anwendung	16
Ziel-Port	Port-Nummer der empfangenden Anwendung	16
Länge	Gesamtlänge des Datagramms (Header + Daten)	16
Prüfsumme	Prüfsumme oder Wert 0	16
Daten	zu übertragende Daten der beteiligten Anwendungen	*

Tabelle 6.2 Aufbau des UDP-Headers

Die Bildung der Prüfsumme erfolgt unterschiedlich zwischen IPv4 und IPv6. Bei IPv4 werden folgende Angaben zur Berechnung herangezogen:

- die IP-Adresse des Absenders
- die IP-Adresse des Empfänger-Rechners
- das Leerfeld mit der Ziffernfolge 00000000
- die Protokoll-ID (17)
- die Länge des Datagramms

Die verbindlichen Festlegungen hierzu finden Sie in den RFCs **768**, **791**, 1349, 2460, 2474, 3168, 3260, 3828, 4301, 5095, 5722, 5871, 6040, 6437, 6564, 6864, 6935, 6946, 7045 und 7112.

RFC 6335 spricht eine Variante von UDP an, die sich in der Berechnung der Prüfsumme unterscheidet. Das »UDP-Lite« wird für Multimedia-Übertragungen bevorzugt eingesetzt.

6.3 Nutzung von Services mittels Ports und Sockets

Mittels der IP-Adresse allein können die Anwendungen Ihrer Rechner nicht miteinander kommunizieren. Die IP-Adressen sind nur für die Adressierung zwischen den Rechnern zuständig. Eine Netzwerkanwendung selbst kann nicht mit einer IP-Adresse versehen werden. Die sendenden und empfangenden Anwendungen benötigen für ihre Kommunikation im Netzwerkverkehr aber eine Art Kanal, was mit den Ports umgesetzt wird.

Die von Ihnen bemühten Netzwerkdienste benötigen ebenfalls einen Antwort-Port, einen Rückkanal. Ihre Client-Anwendungen benutzen dafür nicht die Standard-Port-Nummern, sondern verwenden Nummern aus dem dynamischen Bereich (49152–65535, siehe Tabelle 6.3). Diese Port-Nummer übergibt Ihre Client-Anwendung schon beim

Verbindungsaufbau dem betreffenden Netzwerkservice (Abbildung 6.6).

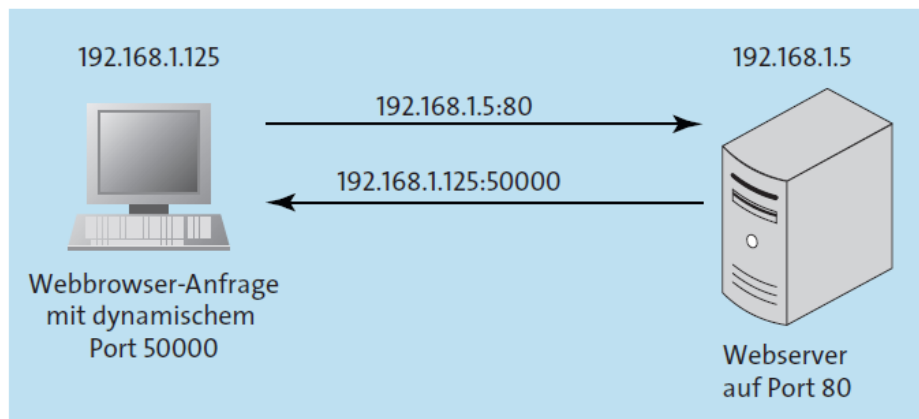


Abbildung 6.6 Port-Nummern für Anfrage und Antwort

Port-Nummern erweitern nicht nur die IP-Adressen um ein weiteres Merkmal, sondern ermöglichen darüber hinaus

- die Annahme und Abgabe der zu transportierenden Daten,
- die gleichzeitige Nutzung mehrerer verschiedener Netzwerkdienste
- (WWW, Mail, SSH ...),
- die mehrfache Nutzung gleichartiger Dienste
- (gleichzeitiges Anzeigen mehrerer Webseiten) und
- die mehrfache Bedienung von Clients.

Eine vereinfachte Darstellung dieser Mechanismen finden Sie in Abbildung 6.7. Wie Sie sehen, greifen die PCs 192.168.1.125 bis 192.168.1.127 gleichzeitig auf den Webserver 192.168.1.5 zu. Dabei haben zwei dieser PCs dem Server sogar dieselbe Port-Nummer für die Rückantwort übermittelt. Das ist kein Problem, weil die Port-Nummer ja nur ein erweiterter Bestandteil der Adressierung ist.

Auf dem PC 192.168.1.127 arbeiten gleichzeitig ein Webbrowser, ein SSH-Client und ein E-Mail-Programm. Dieser PC hat gleichzeitig zu verschiedenen Servern Verbindung, im Fall des E-Mail-Verkehrs sogar mehrere Verbindungen zu dem gleichen Server.

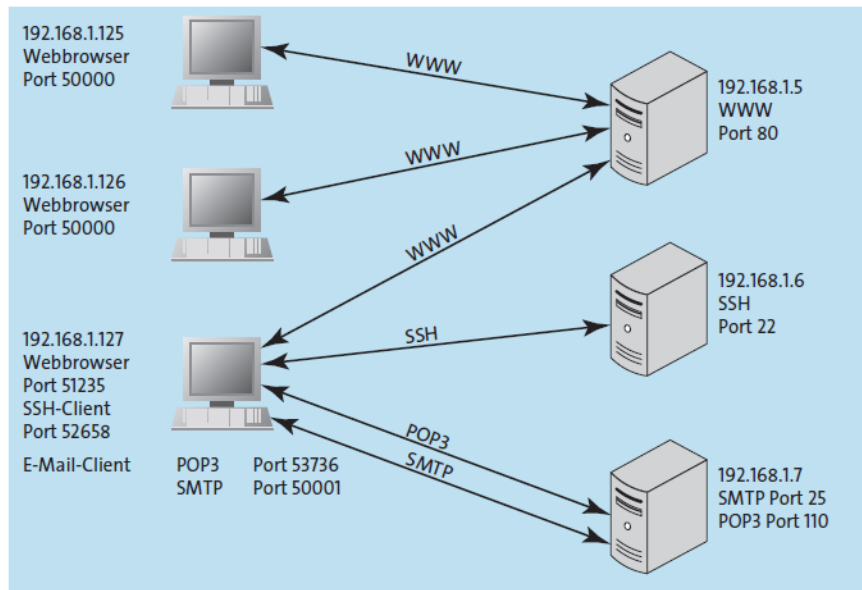


Abbildung 6.7 IP-Adressen und Port-Nummern, Zugriff auf Netzwerkdienste

6.3.1 Sockets und deren Schreibweise

Bei Verbindungen mittels TCP bildet die IP-Adresse zusammen mit einem Port einen *Socket*. IPv6-Sockets können Sie mit unterschiedlichen Schreibweisen darstellen.

Sockets und deren Schreibweisen

IPv4, Telnet: 192.168.1.125:23

IPv6, Telnet: [fd91:e4a5:5e03::dd05]:23

oder

fd91:e4a5:5e03::dd05:23

oder

fd91:e4a5:5e03::dd05.23

oder

fd91:e4a5:5e03::dd05#23

(RFCs 5952, Nr. 6 und 3986, 6874, 7320)

6.3.2 Übersicht über die Port-Nummern

Die Ports werden in drei Gruppen unterteilt:

- 0–1023: *Well-known-Ports*; diese sind weltweit eindeutig durch die IANA festgelegt.
- 1024–49151: *Registered Ports*; diese sind teilweise für Anwendungen bei der IANA registriert.
- 49152–65535: frei für die dynamische Nutzung

Die komplette Liste finden Sie unter www.iana.org/assignments/port-numbers.

Manche Ports sind mehrfach belegt, was Konflikte hervorrufen kann. Auch Trojaner und andere Schadsoftware benötigen einen Port für die Kommunikation mit der Außenwelt.

In Tabelle 6.3 finden Sie einige häufig benötigte Port-Nummern aufgelistet. Ihre Rechner verfügen ebenfalls über eine derartige Aufstellung:

- Linux/FreeBSD: */etc/services*
- Windows: *%WINDIR%\system32\drivers\etc\services*

Nr.	Protokoll	Service	Zweck
20	UDP/TCP	FTP-Daten	Datenübertragung per <i>File Transmission Protocol</i> (FTP)
21	UDP/TCP	FTP-Sitzungssteuerung	Steuerung der FTP-Datenübertragung
22	UDP/TCP	SSH SFTP	Secure-Shell-Fernsitzung (SSH) und <i>Secure File Transport Protocol</i> , SFTP
23	UDP/TCP	TELNET	Telnet-Fernsitzung
25	UDP/TCP	SMTP	<i>Simple Mail Transfer Protocol</i> , Versenden von E-Mails (Annahme durch den Server)
53	UDP/TCP	DNS	Domain-Nameserver
69	UDP/TCP	TFTP	<i>Trivial File Transfer Protocol</i>
80	UDP/TCP	WWW	HTTP-Webserver
110	UDP/TCP	POP3	<i>Post Office Protocol</i> V.3, Auslieferung von E-Mails
119	UDP/TCP	NNTP	<i>Network News Transfer Protocol</i> , Übertragen von News-Group-Nachrichten

Tabelle 6.3 Häufig benutzte Ports

Nr.	Protokoll	Service	Zweck
137	UDP/TCP	NETBIOS-NS	NetBIOS-Name-Service
138	UDP/TCP	NETBIOS-DGM	NetBIOS-Datagramm-Service
139	UDP/TCP	NETBIOS-SSN	NetBIOS-Session-Service
143	UDP/TCP	IMAP	<i>Internet Message Access Protocol (IMAP)</i> , Zugriff auf E-Mails
161	UDP/TCP	SNMP	<i>Simple Network Management Protocol</i> , Geräteüberwachung im Netzwerk
162	UDP/TCP	SNMPTRAP	Übermittlung der SNMP-Traps
179	UDP/TCP	BGP	<i>Border Gateway Protocol</i> , Routing-Protokoll zwischen autonomen Systemen
194	UDP/TCP	IRC	<i>Internet Relay Chat Protocol</i> , Chat-Server
220	UDP/TCP	IMAP V3	IMAP Version 3, E-Mail-Zugriff
396	UDP/TCP	NETWARE-IP	Novell-Netware, über IP-Netzwerk
401	UDP/TCP	UPS	Meldungen von unterbrechungsfreien Stromversorgungen
434	UDP/TCP	MOBILE-IP-AGENT	Mobile-IP-Agent
443	UDP/TCP	HTTPS	Webserver mit verschlüsselter Verbindung
514	UDP	SYSLOG	zentrale Aufzeichnung von Systemereignissen
989	UDP/TCP	FTPS-Daten	FTP über SSL-Verschlüsselung
990	UDP/TCP	FTPS	Steuerung FTP über SSL-Verschlüsselung
992	UDP/TCP	TELNETS	Telnet über SSL-Verschlüsselung
993	UDP/TCP	IMAPS	IMAP V.4 über SSL-Verschlüsselung
994	UDP/TCP	IRCS	IRC über SSL-Verschlüsselung
995	UDP/TCP	POP3S	POP3 über SSL-Verschlüsselung

Tabelle 6.3 Häufig benutzte Ports (Forts.)

Im Normalfall geben Sie beim Benutzen von Standardanwendungen (Webbrowser, E-Mail-Programm, FTP-Client ...) keine Port-Nummer beim Adressaufruf an. Ihre Anwendungen werden stets diese Standard-Port-Nummern benutzen. Aber es gibt genug

Situationen, in denen Sie Internet-Services eine von den IANA-Vorgaben abweichende Port-Nummer zuweisen:

Testumgebungen auf Rechnern mit »Echtanwendungen«: Prüfen Sie Ihre neuen Webseiten usw. in der künftigen Zielumgebung. Sie können auch externe Tester mit in die Arbeit einbinden. Außenstehende bemerken dies in der Regel aber nicht.

Sicherheitsgründe: Webgestützte Administrationsoberflächen oder firmeninterne E-Mail-Dienste für Außendienstmitarbeiter lassen Sie besser abseits der Standard-Ports abrufbar sein.

Ein Netzwerkdienst soll mehrfach vorhanden sein, z. B. SSH bei FreeBSD-Servern innerhalb von Jails.

6.3.3 Ports und Sicherheit

Ihre Rechner verfügen über offene Ports. Über diese kann Ihr Rechner von außen her auf verschiedene Weise angegriffen werden.

Mit einem *Portscanner* (z. B. *nmap*) überprüfen Sie, welche Ports an Ihrem Rechner geöffnet sind:

```
harald@ZE4:~$ nmap -p 1-65535 192.168.0.34
Starting Nmap 4.62 ( http://nmap.org ) at 2016-06-19 20:30 CEST
Interesting ports on ze4 (192.168.0.34):
Not shown: 65530 closed ports
PORT STATE SERVICE
22/tcp open  ssh
53/tcp open  domain
111/tcp open  rpcbind
7741/tcp open unknown
32955/tcp open unknown
Nmap done: 1 IP address (1 host up) scanned in 0.
949 seconds
```

Sie sehen die geöffneten Ports. Damit alle Ports gescannt werden, geben Sie den Bereich mit der Option `-p` an. Eine genaue Anleitung erhalten Sie mit `man nmap`, falls Sie unter Linux oder FreeBSD arbeiten.

Sie dürfen mit einem Portscanner nur Ihre eigenen Systeme überprüfen. Eine darüber hinausgehende Nutzung kann strafrechtliche Folgen nach sich ziehen!

Ein Portscanner hat verschiedene Modi. Der TCP-SYN-Scan z. B. sendet ein Paket mit SYN-Flag zur vermeintlichen Verbindungsaufnahme. Sie können damit drei verschiedene Ergebnisse erhalten:

keine Antwort: Ein Paketfilter lässt weder Anfrage noch Antwort passieren.
 SYN/ACK-Flag-Paket: Der Port ist offen.
 RST-Flag-Paket: Der Port ist geschlossen.

UDP-Portscans arbeiten nicht so zuverlässig. Der Portscanner sendet ein leeres UDPPaket an die jeweiligen Ports. Sendet der so kontrollierte Rechner ein UDP-Paket zurück, ist der Port offen. Aber auch beim Ausbleiben einer Antwort kann der Port offen oder durch einen Paketfilter geblockt sein. Diese Mehrdeutigkeit führt zu Problemen bei der Auswertung. Geschlossene Ports erkennen Sie an der Port-Unreachable-Meldung (ICMPv4: Typ 3 Code 3, ICMPv6: Typ 1 Code 4). Moderne Betriebssysteme drosseln die Ausgabe von ICMP-Meldungen, sodass der Portscan sehr lange dauern kann.

Auch mit `netstat` können Sie Ihren Rechner auf offene Ports hin überprüfen. Im Beispiel aus Abbildung 6.8 sehen Sie die Ausgabe unter Linux (bei FreeBSD funktioniert der Befehl genauso).

```

harald@ZE4:~$ netstat -atn
Aktive Internetverbindungen (Server und stehende Verbindungen)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp        0      0 0.0.0.0:111             0.0.0.0:*               LISTEN
tcp        0      0 192.168.0.102:53        0.0.0.0:*               LISTEN
tcp        0      0 192.168.0.34:53         0.0.0.0:*               LISTEN
tcp        0      0 127.0.0.1:53            0.0.0.0:*               LISTEN
tcp        0      0 0.0.0.0:22              0.0.0.0:*               LISTEN
tcp        0      0 127.0.0.1:631           0.0.0.0:*               LISTEN
tcp        0      0 127.0.0.1:5432          0.0.0.0:*               LISTEN
tcp        0      0 127.0.0.1:25            0.0.0.0:*               LISTEN
tcp        0      0 127.0.0.1:953           0.0.0.0:*               LISTEN
tcp        0      0 0.0.0.0:32955           0.0.0.0:*               LISTEN
tcp        0      0 0.0.0.0:7741            0.0.0.0:*               LISTEN
tcp        0      0 192.168.0.34:52076      192.168.0.25:22         VERBUNDEN
tcp        0      0 192.168.0.34:22         192.168.0.25:27599      VERBUNDEN
tcp6       0      0 :::53                   :::*                     LISTEN
tcp6       0      0 :::22                   :::*                     LISTEN
tcp6       0      0 :::1:631                 :::*                     LISTEN
tcp6       0      0 :::1:5432                 :::*                     LISTEN
tcp6       0      0 :::1:953                 :::*                     LISTEN
harald@ZE4:~$

```

Abbildung 6.8 Ausgabe von »netstat«

Für das Ergebnis in Abbildung 6.8 lautet die Eingabe:

```
netstat -atn.
```

Hier werden Ports und Adressen numerisch ausgegeben. Sie erkennen übrigens gut die Socket-Schreibweise. Möchten Sie hingegen die Ausgabe mit Rechner- und Port-Namen, geben Sie nur

```
netstat -at
```

ein.

Mit der Eingabe von

```
netstat -tun
```

listen Sie alle von außen zugreifenden Verbindungen auf.

Möchten Sie die zugehörigen Prozessnummern in Erfahrung bringen, geben Sie unter Windows zusätzlich die Option »o« und bei Linux/FreeBSD »p« ein. Dass kann Ihnen beim Eingrenzen unerwünschter offener Ports behilflich sein. Über das jeweilige Prozessmanagement bekommen Sie mit dieser Prozess-ID weitere Informationen geliefert. Weitere Erläuterungen zu netstat finden Sie in Abschnitt 10.7.2, »Verbindungen mit »netstat« anzeigen«.

Damit Ihr Rechner so wenige offene Ports wie möglich aufweist, können nachstehende Maßnahmen nützlich sein:

- Deinstallieren Sie nicht benötigte Netzwerkanwendungen.

- Bei Netzerkanwendungen, die Ihr Rechner nur lokal für sich benötigt, beschränken Sie den Zugriff entsprechend auf localhost bzw. 127.0.0.1.
- Blocken Sie Ports mittels Firewall.
- Linux/FreeBSD: Verwenden Sie keinesfalls inetd, und verwenden Sie für jeden Netzwerkservice
- ein eigenes Start-Stopp-Skript.

Weitere Informationen zum Thema Portscan finden Sie in Abschnitt 6.7.2, »Durchführen von Portscans zum Austesten von Sicherheitsproblemen«.

6.4 Die Firewall

Eine Firewall regelt, welche Dienste des »dahinterliegenden« Netzes von einem anderen Netz aus (meist dem Internet allgemein) erreichbar sind. Damit erhält Ihr Netz einen gewissen Schutz, was direkte Angriffe von außen her angeht. Sie können die Firewall aber auch als eine von mehreren Komponenten innerhalb Ihres Schutzkonzepts verwenden.

Die Firewall befindet sich zwischen dem Internet und einem privaten Netz (Intranet). Sie können sie als zentrale Netzwerkkomponente (Hardware-Firewall) oder als Softwarelösung auf einem einzelnen Rechner (Software-Firewall, Personal Firewall) einsetzen. Eine Hardware-Firewall benutzt schnelle Spezialhardware, die auch stärkeres Datenaufkommen gut bewältigt. Die für den Anschluss kleiner Büros oder von Privathaushalten eingesetzten (DSL-)Router sind oftmals zusätzlich mit einer Firewall ausgestattet. Diese reicht für den Schutz kleiner Netze aus.

Die Firewall

- ▶ Eine Firewall regelt den Zugriff allein auf Basis von IP-Adresse, Protokoll (TCP/UDP) und Ports. Sie untersucht nicht den Inhalt der Pakete hinsichtlich unerwünschter Inhalte.
- ▶ Eine Firewall ist im Grunde nichts anderes als ein Router. Nur versucht dieser, alle Pakete uneingeschränkt weiterzuleiten. Die Firewall hingegen lässt mehr oder weniger nichts passieren, es sei denn, Sie erlauben es ihr explizit.
- ▶ Sie können zwei Firewalls verwenden und zwischen ihnen eine *demilitarisierte Zone* (DMZ) einrichten. Mit der DMZ schaffen Sie eine sichere Möglichkeit, dass bestimmte Rechner ihre Dienste sowohl im Internet als auch im Intranet anbieten (FTP, WWW, Mail ...). Die Sicherheit für das Intranet wird dabei nicht aufgegeben. Wird ein Rechner in der DMZ »feindlich« von außen her übernommen, droht den Geräten innerhalb des Intranets keine unmittelbare Gefahr. Allerdings müssen Sie die Einstellungen der beiden Firewalls genau aufeinander abstimmen, was vor allem Mehrarbeit bedeutet. Alternativ können Sie eine Firewall mit DMZ-Anschluss verwenden.
- ▶ Oftmals übernimmt die Hardware-Firewall weitere Funktionen: NAT/PAT (Adressumsetzung), Proxyserver.
- ▶ Gewöhnliche Firewalls überprüfen nicht, ob freigegebene Ports auch gemäß ihrer Bestimmung genutzt werden. Hier können Sicherheitsregeln umgangen werden (Beispiel: Ein Spielteilnehmer im Intranet bekommt Zugriff nach außen).
- ▶ Durch die Auswertung der Control-Flags (ACK bei TCP) stehen weitere Kontrollmöglichkeiten zur Verfügung.
- ▶ Bestimmte Dienste (z. B. HTTPS) können nicht inhaltlich überwacht werden.
- ▶ Mit Tunnels über die freigegebenen Ports können Sie Firewalls umgehen. (Aber das Blockieren der IP-Adressen, wie es in [Abschnitt 4.3.1](#), »Der Urtyp: Adressauflösung in der ›hosts«-Datei«, beschrieben wurde, führt ebenfalls zum Blockieren unerwünschter Inhalte!)
- ▶ Firewalls können Sie zusätzlich um weitere Funktionen (VPN-Kopf, QoS, Link Aggregation zum Switch) ergänzen.
- ▶ Eine Sonderform der Paketbehandlung innerhalb einer Firewall stellt die *Stateful Packet Inspection (SPI)* dar. Hier ordnet die Firewall die Pakete aktiven Sessions zu.
- ▶ Zum Nachlesen: RFCs 1579, 2588, 2979 und 5207

6.4.1 Integration der Firewall in das Netzwerk

Jede Firewall ist wenigstens über einen Anschluss mit dem Internet verbunden. Im einfachsten Fall befindet sich am zweiten Anschluss das Intranet mit der sicheren Zone (Abbildung 6.9).

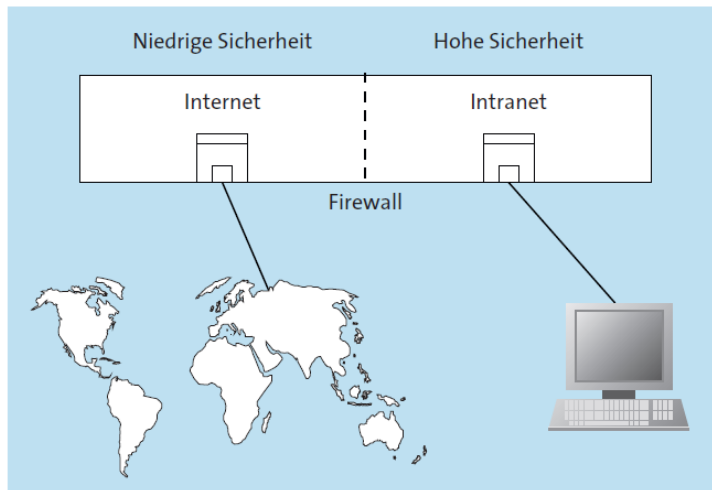


Abbildung 6.9 Firewall als Trennung von Internet und Intranet

Wenn Sie Rechner mit Diensten betreiben, auf die vom Intranet und vom Internet her zugegriffen werden soll, müssen Sie eine DMZ schaffen. Sie können hierzu entweder zwei Firewalls (Abbildung 6.10) oder eine Firewall mit DMZ-Anschluss verwenden (Abbildung 6.11).

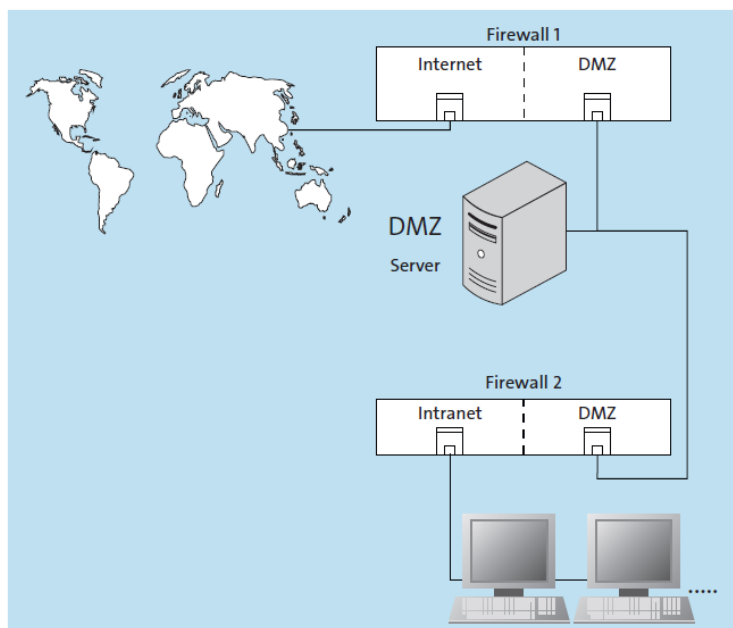


Abbildung 6.10 Aufbau einer DMZ mit zwei Firewalls

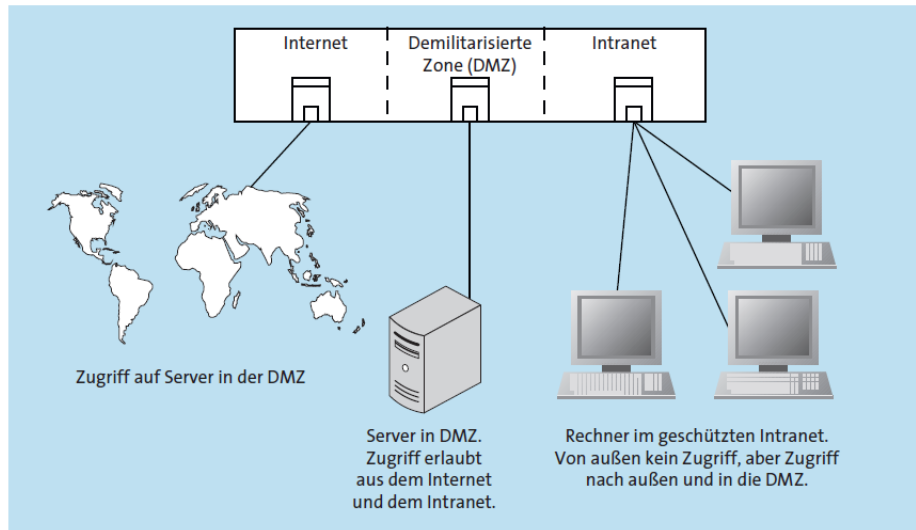


Abbildung 6.11 Firewall mit eigenem DMZ-Anschluss

6.4.2 Regeln definieren

Firewall-Regeln geben an, was mit den Datenpaketen geschehen soll (Tabelle 6.4).

Aktion (Beschreibung)	Aktion
Datenpaket passieren lassen	pass, allow
Datenpaket ablehnen	deny, reject, block
Datenpaket verwerfen	drop

Tabelle 6.4 Aktionen für Firewall-Regeln

Die Regeln selbst bestehen aus mehreren Angaben:

- IP- oder Netzwerkadresse abgehend
- Ziel-IP-Adresse
- Protokoll (ICMP, TCP, UDP usw.)
- Port-Nummer (nur bei TCP und UDP)
- Aktion (Tabelle 6.4)
- Eintrag in Log-Datei (ja/nein)

Bei kleineren Hardware-Firewalls werden Sie durch ein Webinterface bequem alle notwendigen Einstellungen per Mausklick vornehmen (Abbildung 6.12). Hier kümmert sich das Gerät um die richtige Umsetzung der Regeln.

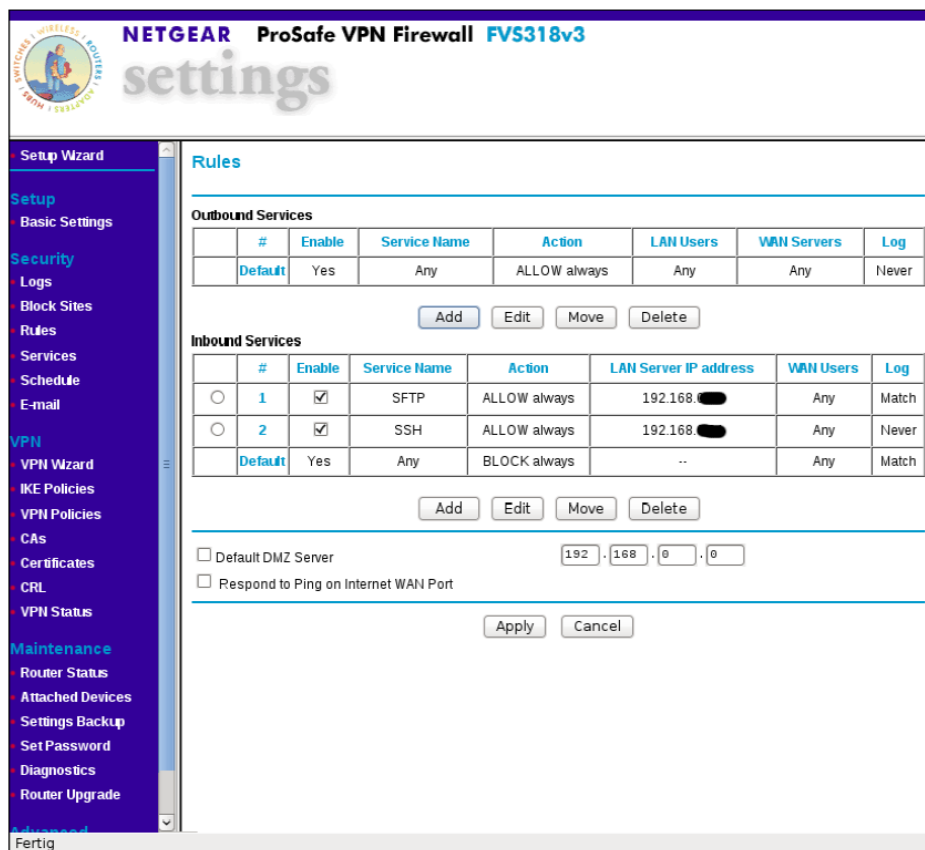


Abbildung 6.12 Webinterface einer Firewall

Für den Anschluss kleinerer Netze an das Internet können Sie auch einen Mini-PC (Boardformfaktor ITX o. Ä.) einsetzen. Diese kleinen Geräte weisen eine geringe Stromaufnahme auf und verfügen oft sogar über mehrere Netzwerkschnittstellen. So ein Gerät werden Sie für diese Netzwerkzwecke meist unter Linux oder FreeBSD betreiben.

Unter Linux können Sie entsprechende Firewall-Softwarepakete installieren. Mit dem `iptables`-Kommando setzen Sie Ihre Anweisungen ab. Dabei müssen Sie beachten, dass Sie mit der `-t`-Option eine der vier Tabellen direkt ansprechen können: `filter`, `nat`, `mangle` und `raw`. Die wichtigsten Kommandos finden Sie in Tabelle 6.5.

Aktion	Kommando
Regel an Tabelle (Ende) anfügen	-A
Regel löschen	-D
Regel einfügen	-I
Regel ersetzen	-R
Regeln auflisten	-L

Tabelle 6.5 Kommandos von »iptables«

Weiterführende Informationen finden Sie in der Manual-Seite von `iptables` (`man iptables`). Wenn Sie die Firewall unter FreeBSD einsetzen wollen, müssen Sie zunächst unter `/etc/rc.conf` den Eintrag

```
firewall_enable="YES"
firewall_type="<TYP>"
```

setzen. Mit TYP wählen Sie, wie und was die Firewall schützt (Tabelle 6.6).

Anweisung	Eintrag TYP
Sämtlichen Netzwerkverkehr durchlassen	open
Nur den Rechner schützen, auf dem die Firewall läuft	client
Das ganze angeschlossene private Netzwerk schützen	simple
Sämtlichen Netzwerkverkehr (ausgenommen loopback) unterbinden	closed
Keine Firewall-Regeln laden	UNKNOWN
Angabe einer Datei mit (eigenen) Firewall-Regeln	<DATEINAME>

Tabelle 6.6 Angaben für den Firewall-Start unter FreeBSD

Das genaue Regelwerk zum Setzen eigener Regeln finden Sie im FreeBSD-Handbuch sehr gut dokumentiert. Da es sich hierbei ähnlich wie bei anderen Systemen und Programmen verhält, gebe ich nur die Handbuchseite an: <http://www.freebsd.org/doc/de/books/handbook/firewalls-ipfw.html>. Weitere Informationen finden Sie zudem in der Manual-Seite zu ipfw. Wenn Sie eine Firewall unter FreeBSD konfigurieren, bearbeiten Sie meist gleich die Einstellungen für die Adressumsetzung (NAT). Hier sehen Sie besonders gut, dass die Firewall nur ein Baustein in einem ganzen Sicherheitskonzept ist.

Machen Sie sich mit den Grundzügen der richtigen Regelsetzung vertraut:

Achten Sie auch auf den ausgehenden Netzwerkverkehr! Lassen Sie ausgehende E-Mails nur über den eigenen SMTP-Server zu. Nur hier können Sie unter Umständen hinausgeschmuggelte

Betriebsgeheimnisse abfangen (Gesetze und Betriebsvereinbarungen beachten!). Internet-Onlinespiele, die nicht den HTTP-Port benutzen, sollten Sie ebenfalls blockieren. Die Netzwerkaktivitäten von Schadprogrammen aus dem Intranet werden damit beeinträchtigt und fallen auch leichter auf.

Wenden Sie die richtige Strategie an: Grundsätzlich ist alles gesperrt. Nur das, was wirklich notwendig ist, darf die Sperre passieren. Natürlich ist das arbeitsintensiv, dafür aber wirkungsvoll!

Log-Dateien sind nützlich für die Fehlersuche. Sie haben damit zudem die (fast einzige) Möglichkeit, Angriffe im Nachhinein zu erkennen.

Sie müssen die Firewall selbst schützen. Sie darf nicht zu sich selbst Verbindung aufnehmen können. Setzen Sie zuvor eine Regel über den Administrationszugriff vom Intranet aus fest.

Aktion drop: Der sendende Rechner bekommt keine Information über die blockierte Verbindungsanfrage. Das ist für Anfragen von außen sinnvoll, weil damit keine zusätzliche Netzlast bei DOS-Angriffen erzeugt wird.

Aktion reject: Der sendende Rechner bekommt bei UDP ICMP-Unreachable oder bei TCP das Control-Flag RST. Für Anfragen vom Intranet aus ist das sinnvoll, da Sie weniger Probleme bei eventuellen Fehlerklärungen vorfinden.

ICMP: Schalten Sie nur folgende ICMP-Nachrichtentypen frei:

Für IPv4:

- Typ 3 Code 1 (unreachable)
- Typ 3 Code 4 (Fragmentation needed)
- Typ 8 (Echo Request)

Für IPv6:

- Typ 1 (Destination unreachable), Code 3 (Adresse nicht erreichbar),
- Typ 2 zulässige Paketgröße überschritten (Code 0),
- Typ 3 Zeitablauf, Code 0 (Hop-Limit überschritten) und Code 1 (Zeit zum Zusammensetzen von Fragmenten überschritten) und
- Typ 128 (Echo Request)

6.5 Der Proxyserver

Mit einem Proxyserver erhöhen Sie die Sicherheit Ihres Netzwerks und können auch die Netzlast des Internetzugangs verringern. Der Server nimmt Ihre Anfragen in das Internet entgegen und führt diese unter seiner Identität durch. Die Antworten reicht er wiederum an Sie selbst zurück. Betrachten Sie ihn als weitere Komponente eines Sicherheitskonzepts für Ihr Netzwerk.

Proxyserver

- ▶ **Sicherheit:** keine bloße Weiterleitung von Datenpaketen. Der Proxy tritt als Ihr Stellvertreter im Internet auf (Anonymisierung).
 - ▶ **Sicherheit:** Angreifer vom Internet her sehen nur den Proxyserver und greifen gegebenenfalls nur diesen an. Die anfragenden Clients bleiben verschont. Leider können Sie auf diese Weise nur einen Teil moderner Angriffstechniken abwehren.
 - ▶ **Sicherheit:** Teilweise ist eine Paketanalyse möglich. Damit erkennt Ihr System Missbräuche (Backdoors, Datenklau usw.).
 - ▶ **Sicherheit:** Umgekehrt kann Ihr Proxyserver Anfragen vom Internet entgegennehmen und an einen Server im geschützten Bereich weiterleiten. Damit können Sie den Zugriff auf dessen Dienste (Webserver, Datenbanken usw.) auf wenige Adressen beschränken, von denen eine dem Proxyserver gehört.
 - ▶ **Wirtschaftlichkeit:** Bandbreitenkontrolle ermöglicht die Benutzung auch schmalbandigerer Internetzugänge.
 - ▶ **Wirtschaftlichkeit:** Durch die Pufferung eingehender Daten, z. B. Webseiten, vermindern Sie die Netzlast für den Internetzugang.
 - ▶ **Kontrolle:** Überwachen Sie die Häufigkeit von Seitenzugriffen, die Netzlast und andere statistische Größen auf Ihren Webseiten.
 - ▶ **Transparenter Proxyserver:** Er arbeitet mit einer Firewall zusammen, der Benutzer muss keine besonderen Einstellungen an seinem Client-Programm vornehmen.
 - ▶ **Dedicated Proxyserver:** Dieser Proxyserver arbeitet für genau ein Protokoll (HTTP, HTTPS, FTP ...) und kann gegebenenfalls sehr tief in die Kommunikation eingreifen. Die zu übertragenden Daten können Sie damit analysieren oder gar verändern. Für jedes Protokoll müssen Sie einen eigenen Dedicated Proxyserver installieren. Diesen Typ können Sie auch als Cache zum Puffern verwenden. Wird z. B. die gleiche Webseite häufig aufgerufen, stellt der Proxyserver diese dem Benutzer bereit, ohne dass auf das Internet zugegriffen werden muss. Damit verringern Sie die Zugriffszeit und die Netzlast. Allerdings muss der Inhalt des Caches in regelmäßigen Abständen erneuert werden, damit Sie keine veralteten Informationen vorhalten.
-
- ▶ **Generischer Proxyserver:** Dieser Typ eines Proxyservers reicht Ihre Pakete unbesehen weiter. Er wertet weder Protokolle noch Inhalte aus. Genau besehen haben Sie es »nur« mit einem Paketfilter Ihrer Firewall zu tun, mit dem Sie unter anderem eine 1:1-Weiterleitung sowie Sperren von Ports und IP-Adressen vornehmen können.
 - ▶ **Reverse Proxyserver:** Diesen Proxyserver setzen Sie immer in Zusammenhang mit anderen Rechnern ein, die Netzwerkdienste (HTTP, FTP ...) anbieten. Nach außen verbergen Sie die IP-Adresse des Dienste-Servers und tragen somit zu dessen Schutz bei. Sie entlasten z. B. Ihren Webserver dadurch, dass der Proxyserver Verschlüsselungsaufgaben übernimmt (HTTPS) und häufig abgefragte Seiten puffert. Außerdem können Sie hier auch Zugriffsstatistiken und andere Auswertungen erstellen ([Abbildung 6.14](#)).
 - ▶ **RFCs zum Nachlesen:** 2607, 3143

6.5.1 Lokaler Proxyserver

Wenn Sie einen lokal auf dem Rechner installierten Proxyserver betreiben, gibt dieser die IP-Adresse des Rechners nach außen bekannt. Für Caching- und Statistikzwecke, Bandbreitenkontrolle und in Zusammenarbeit mit einer Firewall für Sicherheitszwecke können Sie den Proxyserver trotzdem einsetzen.

6.5.2 Proxyserver als eigenständiger Netzwerkteilnehmer

Setzen Sie den Proxyserver als eigenständigen Rechner ein, verbirgt dieser die IP-Adresse des Clients bei Abfragen in das Internet. Zusammen mit einer Firewall erreichen Sie eine gute Absicherung des Netzwerks (bei richtiger Konfiguration). Diesen Proxyserver setzen Sie vor die Firewall (Abbildung 6.13).

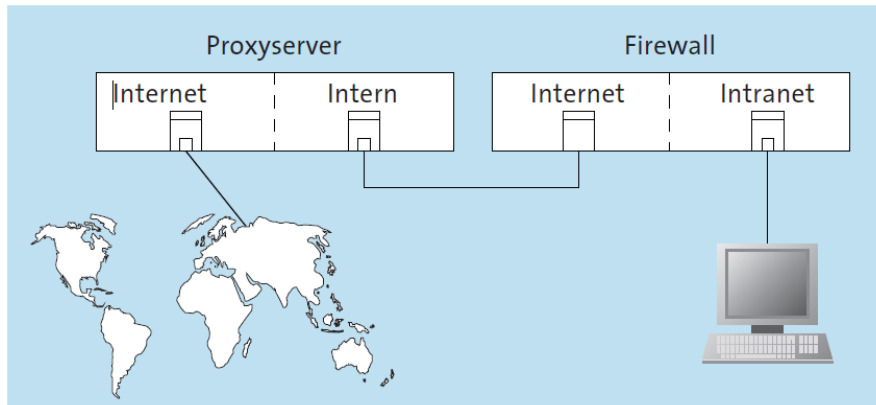


Abbildung 6.13 Lage des Proxyservers

Das umkehrte Szenario des Reverse Proxyservers finden Sie in Abbildung 6.14 dargestellt. Der Webserver wird durch den Reverse Proxyserver im Internet »vertreten«.

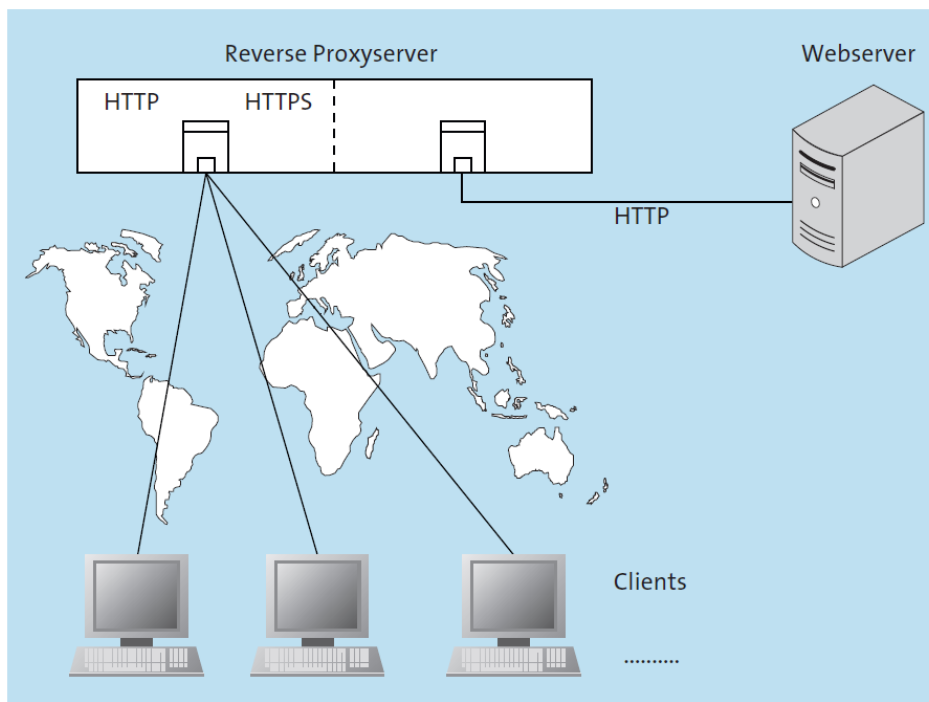


Abbildung 6.14 Lage und Einsatz eines Reverse Proxyservers im Netzwerk

6.5.3 Squid, ein Proxyserver

Die meisten Proxyserver finden Sie auf Hardware-Firewalls oder ähnlichen, nur für Netzwerkzwecke gebauten Geräten. Für eigene Installationen, sowohl im Zusammenhang mit Servern als auch mit Ihrer Netzwerkinfrastruktur, können Sie Squid verwenden. Squid können Sie gleichermaßen für kleine als auch für große Anwendungsszenarien einsetzen.

Squid erhalten Sie unter www.squid-cache.org. Dort finden Sie neben dem Programm auch eine sehr umfangreiche Dokumentation über den Betrieb und die Konfiguration. Ich möchte hier keine schnelle Beschreibung über dessen Konfiguration liefern. Wenn Ihnen Konfigurationsfehler unterlaufen, kann der Schutz des Netzwerks oder eines Servers leicht ausgehebelt werden. Damit verlieren Sie den Schutz Ihrer Daten und Ihrer Rechner.

Prüfen Sie nach der Installation und Konfiguration von Squid vor allem auch das richtige Zusammenspiel mit weiteren Netzwerkkomponenten wie Firewall und Router. Der Proxyserver ist nur sinnvoll einzusetzen, wenn Sie (keine allzu leicht erreichbaren) »Schleichwege« an ihm vorbei geschaffen haben.

6.6 Port and Address Translation (PAT),

Network Address Translation (NAT)

IPv4-Adressen sind schon längere Zeit ein knappes Gut. Solange Sie nicht flächendeckend auf IPv6 zurückgreifen können, müssen Sie deshalb zumindest noch übergangsweise die damit verbundenen Probleme meistern. Solange Sie genau einen Rechner mit dem Internet verbinden möchten, brauchen Sie keine zusätzlichen Maßnahmen zu ergreifen (Abbildung 6.15).

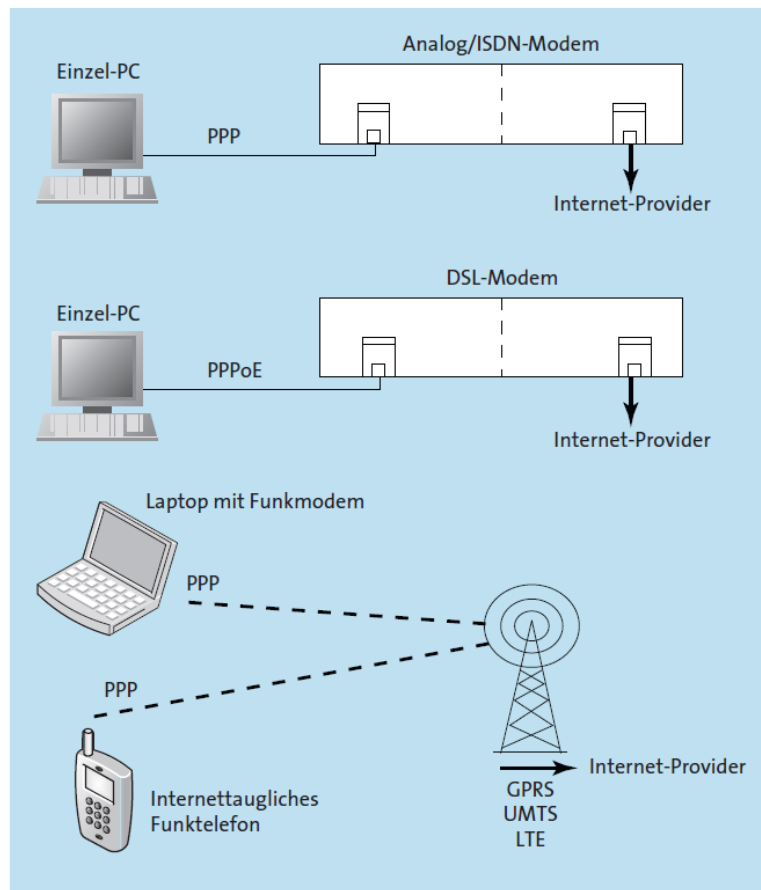


Abbildung 6.15 Anbindung einzelner Rechner an das Internet

Sobald Sie aber ein kleines LAN mit dem Internet verbinden möchten, müssen Sie diese einzige IP-Adresse vom Internet-Provider mit all Ihren lokalen Rechnern teilen.

Mittels der *Port and Address Translation* (PAT, auch als *IP-Masquerading* bezeichnet) setzen Sie (viele) private IP-Adressen auf eine (einzige) im Internet gültige Adresse um. Hierzu ersetzt Ihr Einwahl-Router die Quelladressen und -Ports aus dem lokalen Netz durch seine eigenen Angaben und sendet sie in das Internet. Anhand einer Tabelle hält er fest, für welche Rechner er das aktuell vornimmt.

Damit erkennt er die Antworten aus dem Internet und sendet Antwortpakete an die anfragenden lokalen Rechner zurück. In Abbildung 6.16 sehen Sie ein Beispiel für PAT. Darin wird der gleichzeitige Zugriff mehrerer Rechner aus dem privaten Netz auf ein einziges Ziel im Internet dargestellt. Beachten Sie die Umsetzung der Port-Nummern.

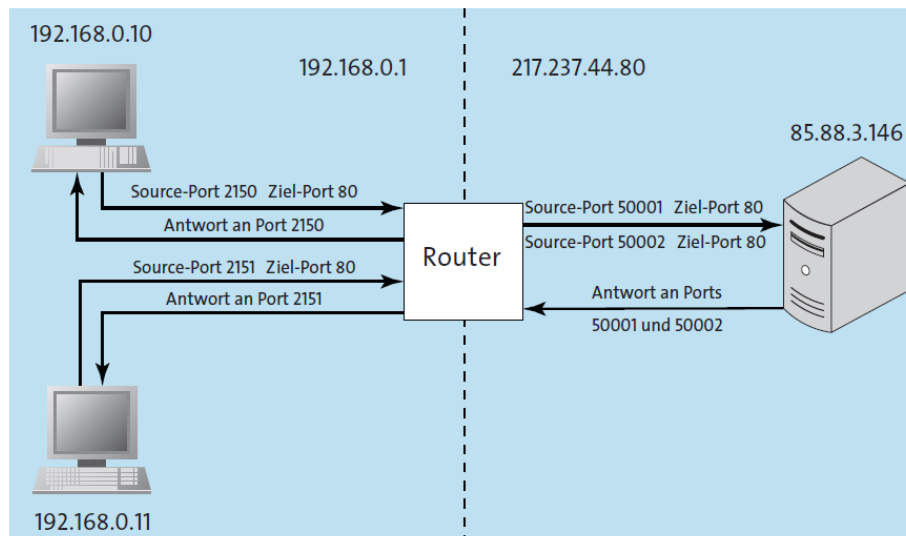


Abbildung 6.16 Internetzugriff mit PAT

Durch dieses Verfahren bleiben Ihre Rechner im Allgemeinen recht gut hinter dem Router verborgen. Sie können diesen Umstand in Ihr Sicherheitskonzept mit einbeziehen, solange Sie IPv4-Adressen benutzen.

Möchten Sie von außen einen oder mehrere Ihrer Rechner in Ihrem privaten Netz erreichen, benötigen Sie hierzu die *Network Address Translation (NAT)*. Diese setzt IP-Adressen 1:1 um. Solange Sie nur für einen Rechner den Netzzugriff von außen benötigen, genügt Ihnen Ihre einzige IPv4-Adresse vom Internet-Provider. Wenn Sie mehrere Rechner »ans Netz« bringen wollen, würden Sie auch mehrere Adressen Ihres Internet-Providers benötigen. In modernen Einwahl-Routern finden Sie dieses Problem gelöst. Diese arbeiten mit einer Mischform von PAT und NAT, und Sie kommen mit einer einzigen Adresse aus. Sie müssen allerdings mit einem kleinen Kompromiss leben: Sobald Sie zweimal denselben Dienst (HTTP, HTTPS, FTP, SSH) über den Router mit dem Internet verbinden möchten, benötigt einer der lokalen Rechner hierfür eine abweichende Port-Nummer (Abbildung 6.17).

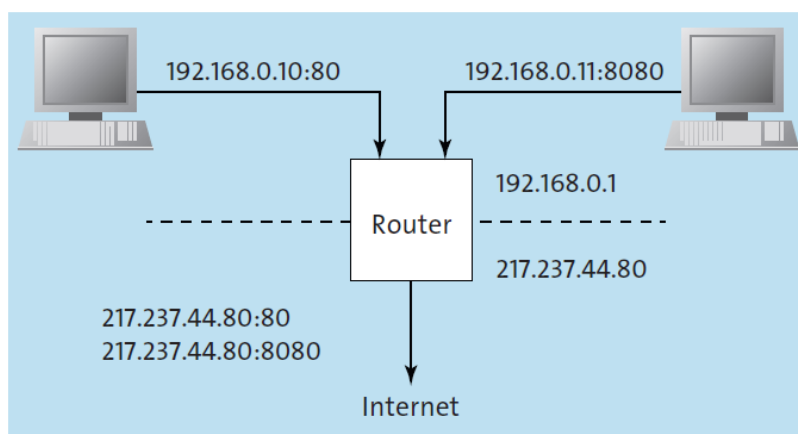


Abbildung 6.17 PAT/NAT zur Verbindung mit dem Internet

6.7 Praxis

In den nächsten Abschnitten stelle ich Ihnen einige konkrete Anwendungsbeispiele vor.

6.7.1 Verbindungsaufbau zu einem Dienst mit geänderter Port-Nummer

Solange Sie über Standard-Ports mit einem Rechner per Netzwerk Verbindung aufnehmen, geben Sie nur den Rechnernamen oder die IP-Adresse ein. Die Anwendung benutzt die Standard-Port-Nummer. Abweichende Port-Nummern müssen Sie der jeweiligen Anwendung mitteilen:

- in der Kommandozeile mit einer Option (hier: -p)
- in URLs nach dem Rechnernamen: <NAME>:<PORT>
- in URLs mit IPv4-Adressen: <IP-ADRESSE>:<PORT>
- in URLs mit IPv6-Adressen: [<IP-ADRESSE>]:<PORT>
- in entsprechenden Eingabefeldern

In Abbildung 6.18 finden Sie einige Beispiele hierzu.

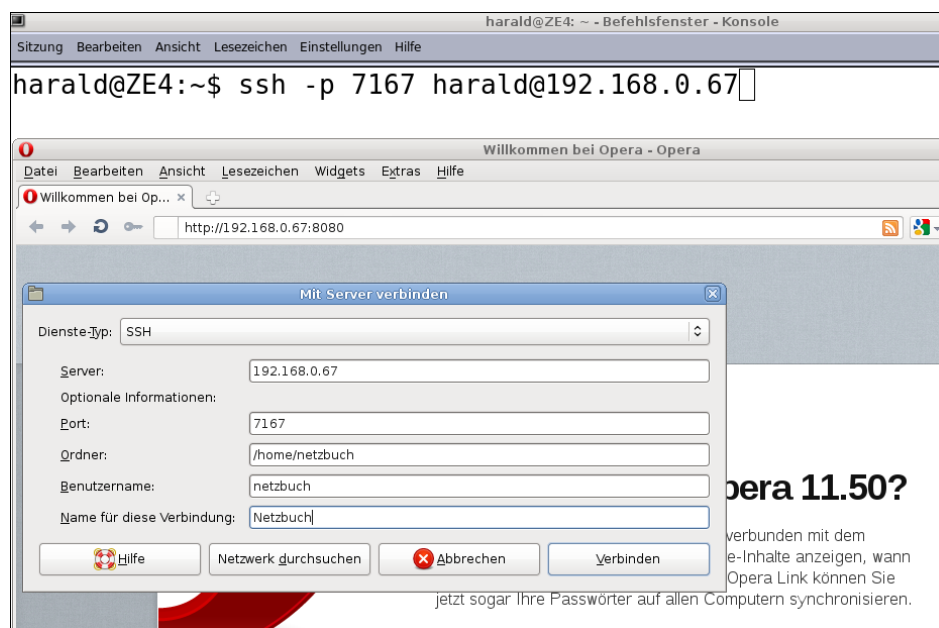


Abbildung 6.18 Angabe von abweichenden Port-Nummern

6.7.2 Durchführen von Portscans zum Austesten von Sicherheitsproblemen

Zunächst die Paragrafen: Scannen Sie keine Systeme, die Sie nicht scannen dürfen! Das kann arbeits- oder strafrechtliche Folgen haben.

Der Portscanner `nmap` spürt offene Ports auf. Einen Schnelldurchlauf erhalten Sie mit

```
nmap <IP-Adresse>
```

oder mit

```
nmap <Rechnername>
```

Abbildung 6.19 zeigt Ihnen den Ablauf. Der Scan fand einen offenen Port (25).

Wenn Sie wissen wollen, wie viel Ihr System nach außen über sich preisgibt und welche Ports geöffnet sind, geben Sie

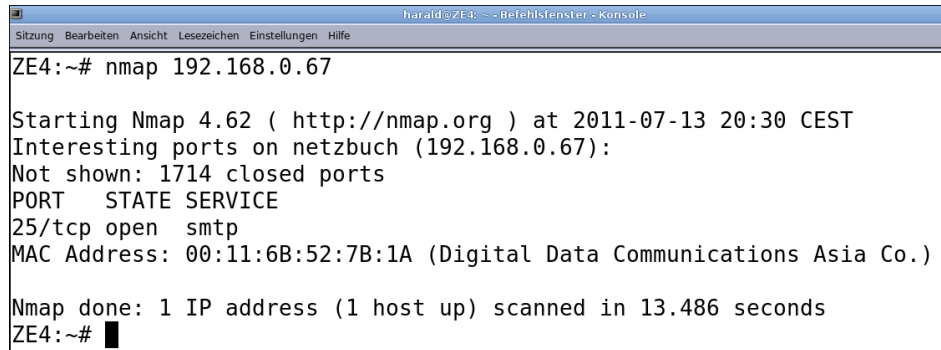
```
nmap -A -T4 <IP-Adresse>
```

oder

```
nmap -A -T4 <Rechnername>
```

ein.

Abbildung 6.20 zeigt Ihnen eine detailliertere Auskunft. Der Testrechner schwieg sich eisern über sein Betriebssystem und die Uptime aus. Dafür wurde das Standard-Gateway als ein Hop erkannt.



```

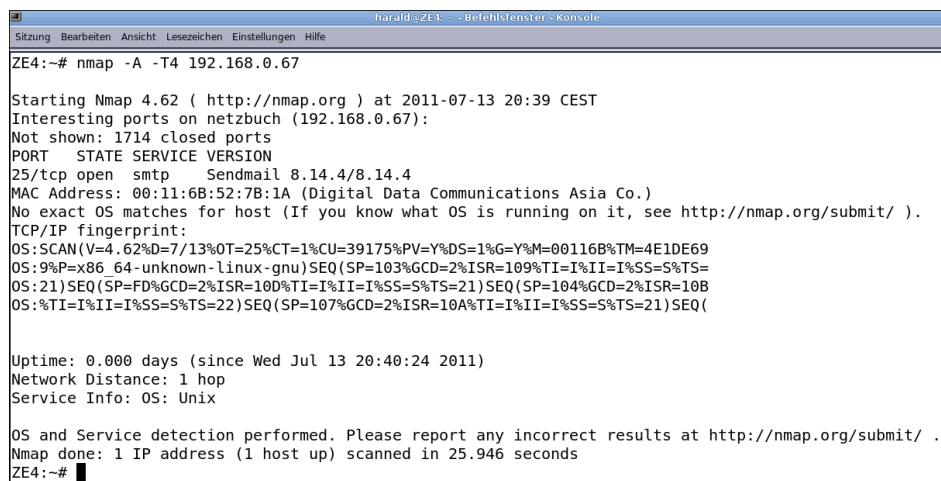
ZE4:~# nmap 192.168.0.67

Starting Nmap 4.62 ( http://nmap.org ) at 2011-07-13 20:30 CEST
Interesting ports on netzbuch (192.168.0.67):
Not shown: 1714 closed ports
PORT      STATE SERVICE
25/tcp    open  smtp
MAC Address: 00:11:6B:52:7B:1A (Digital Data Communications Asia Co.)

Nmap done: 1 IP address (1 host up) scanned in 13.486 seconds
ZE4:~#

```

Abbildung 6.19 Portscan mit »nmap«, Schnelldurchlauf



```

ZE4:~# nmap -A -T4 192.168.0.67

Starting Nmap 4.62 ( http://nmap.org ) at 2011-07-13 20:39 CEST
Interesting ports on netzbuch (192.168.0.67):
Not shown: 1714 closed ports
PORT      STATE SERVICE VERSION
25/tcp    open  smtp      Sendmail 8.14.4/8.14.4
MAC Address: 00:11:6B:52:7B:1A (Digital Data Communications Asia Co.)
No exact OS matches for host (If you know what OS is running on it, see http://nmap.org/submit/ ).
TCP/IP fingerprint:
OS:SCAN(V=4.62%D=7/13%OT=25%CT=1%CU=39175%PV=Y%DS=1%G=Y%M=00116B%TM=4E1DE69
OS:9%P=x86_64-unknown-linux-gnu)SEQ(SP=103%GCD=2%ISR=109%TI=I%II=I%SS=S%TS=
OS:21)SEQ(SP=FD%GCD=2%ISR=10D%TI=I%II=I%SS=S%TS=21)SEQ(SP=104%GCD=2%ISR=10B
OS:TI=I%II=I%SS=S%TS=22)SEQ(SP=107%GCD=2%ISR=10A%TI=I%II=I%SS=S%TS=21)SEQ(

Uptime: 0.000 days (since Wed Jul 13 20:40:24 2011)
Network Distance: 1 hop
Service Info: OS: Unix

OS and Service detection performed. Please report any incorrect results at http://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 25.946 seconds
ZE4:~#

```

Abbildung 6.20 Ausführlicherer Portscan mit »nmap«

Weitere Ausführungen zu nmap finden Sie in Abschnitt 10.7.3, »Hosts und Ports mit »nmap« finden«.

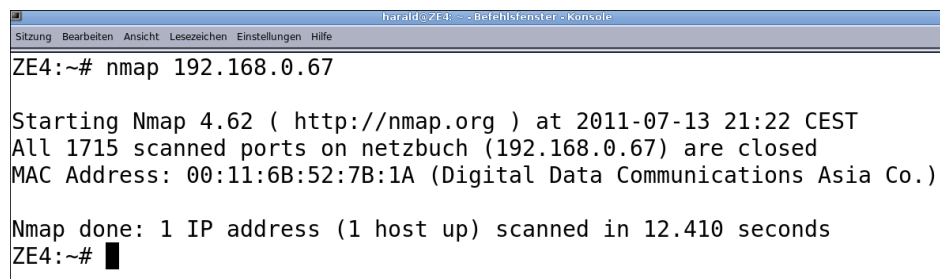
6.7.3 Schließen von Ports

Ports werden als »offen« bezeichnet, wenn sich dahinter eine aktive Anwendung verbirgt. Sie »lauscht« im Netzwerk und wartet darauf, angesprochen zu werden.

Ports

- ▶ **Offen zur Außenwelt:** Wenn Sie einen offenen Port nach außen hin benötigen, muss er auch »sichtbar« sein. Die Anwendung (Webserver, FTP-Server ...) muss immer auf dem neuesten Stand gehalten werden, damit eventuelle Sicherheitslücken keine Einbruchmöglichkeit in Ihr System schaffen. Achten Sie auf eine gute Konfiguration Ihrer Netzanwendung in puncto Sicherheit.
- ▶ **Offen zur Außenwelt, aber (halbwegs) sicher:** Geben Sie einem notwendigerweise offenen Port gegebenenfalls eine Nummer im Bereich von 49152 bis 65535 oder wenigstens über 1023. Standard-Scans erreichen meist die hohen Nummern nicht. Verlassen Sie sich aber trotzdem nicht darauf.
- ▶ **Gefiltert für die Außenwelt:** Mit einer Firewall verbergen Sie den aktiven Port vor Zugriffen aus dem Internet. Portscanner können unterscheiden, ob ein Port mittels Firewall »gefiltert« wurde oder geschlossen ist.
- ▶ **Geschlossen:** Es ist keine Anwendung aktiv, die auf einen bestimmten Port zugreift.

In unserem Beispielfall wurde der SMTP-Daemon auf Port 25 (Sendmail) gestoppt. Der Portscan zeigt nunmehr keine offenen Ports auf dem Testsystem an (Abbildung 6.21).



```

harald@ZE4: ~$ nmap 192.168.0.67

Starting Nmap 4.62 ( http://nmap.org ) at 2011-07-13 21:22 CEST
All 1715 scanned ports on netzbuch (192.168.0.67) are closed
MAC Address: 00:11:6B:52:7B:1A (Digital Data Communications Asia Co.)

Nmap done: 1 IP address (1 host up) scanned in 12.410 seconds
ZE4:~#
  
```

Abbildung 6.21 Negativer Portscan mit »nmap«

6.8 Prüfungsfragen

Hier können Sie Ihr Wissen zu einigen Einzelthemen dieses Kapitels überprüfen. Die Auflösungen finden Sie in Anhang B, »Auflösungen zu den Prüfungsfragen«.

6.8.1 TCP-Protokoll

1. Mit welchem Flag weisen Sie Verbindungswünsche ab?
2. Welchem Zweck dienen die Sequenznummern?

6.8.2 Ports und Sockets

1. Der Rechner mit der IP-Adresse 192.168.7.100 verfügt über einen Webserver, der seine Dienste auf dem Port 8182 anbietet. Wie geben Sie dies in Ihren Webbrowser ein?
2. Sockets sind eine Kombination aus der IP-Adresse und der Port-Nummer. Trifft dies nur für TCP, nur für UDP oder für beide zu?
3. Was entnehmen Sie folgender Programmausgabe?

```

harald@ze8:~$ nmap 192.168.0.67
Starting Nmap 5.00 ( http://nmap.org ) at 2016-09-15 15:06 CEST
Interesting ports on netzbuch (192.168.0.67):
Not shown: 999 closed ports
PORT STATE SERVICE
80/tcp open  http

Nmap done: 1 IP address (1 host up) scanned in 6.85 seconds
  
```

4. Wie spricht ein Portscanner einen Rechner an?
 5. Sie erhalten bei einem Portscan verschiedene Antworten:
 - keine Antwort (ist hier auch eine Antwort)
 - SYN/ACK-Flag-Paket
 - RST-Flag-Paket
- Was bedeuten diese im Einzelnen?

6.8.3 Firewall

Welches Sicherheitsproblem kann eine gewöhnliche Firewall nicht lösen?