

# Vorwort

## Einführung

Dieses Buch beschreibt die TCP/IP-Protokollsuite, aber aus einer anderen Perspektive als andere Texte zu TCP/IP. Statt lediglich die Protokolle und Ihre Funktionsweise zu erläutern, setzen wir ein gängiges Diagnosetool ein, um die Protokolle bei der Arbeit zu beobachten. Die Beobachtung der Protokolle in verschiedenen Szenarien bietet eine tiefer gehende Einsicht in ihre Funktionsweise und hilft uns zu verstehen, warum manche Design-Entscheidungen getroffen wurden. Außerdem können wir die Implementierung der Protokolle untersuchen, ohne uns durch Tausende Zeilen Quellcode durchwälzen zu müssen.

Während der Entwicklung der Netzwerkprotokolle zwischen den 60er und 80er Jahren waren teure, dedizierte Hardwarekomponenten erforderlich, um die Übertragung von Paketen über das Medium zu beobachten. Außerdem mussten sich die Ingenieure damals sehr gut mit den Protokollen auskennen, um die Pakete zu verstehen, die von dieser Hardware dargestellt wurden. Die Funktionalität dieser Hardwareanalysegeräte war außerdem beschränkt auf das, was die Hardwarehersteller für notwendig erachteten.

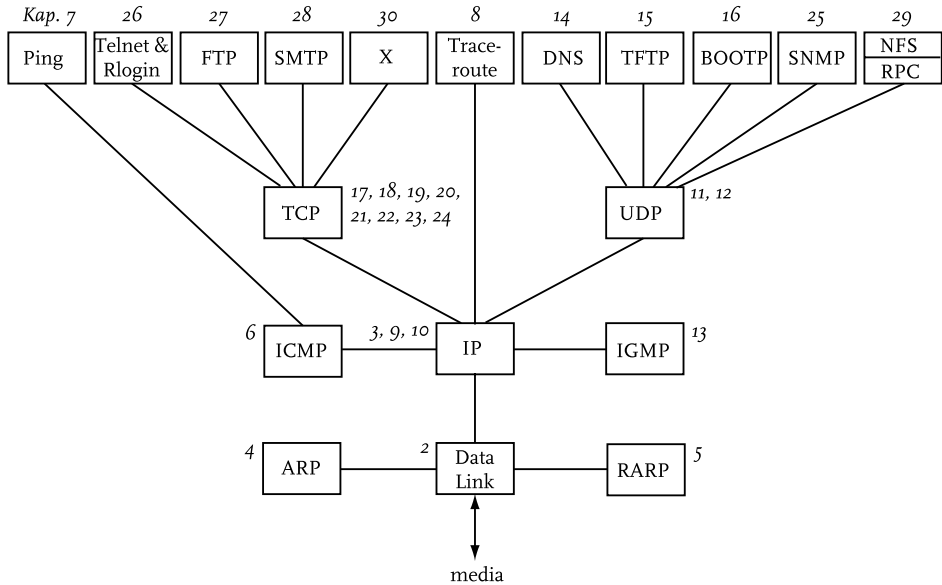
Heute hat sich die Situation stark geändert, weil allgegenwärtige Workstations in der Lage sind, ein Local Area-Netzwerk zu überwachen [Mogul 1990]. Schließen Sie eine Workstation an Ihr Netzwerk an, führen Sie eine öffentlich verfügbare Software aus, die in Anhang A beschrieben wird, und beobachten Sie die Pakete während der Übertragung über das Netzwerkmedium. Obwohl viele Leute diese Tools für die Diagnose von Netzwerkproblemen einsetzen, sind sie mindestens genauso gut geeignet, wenn es darum geht, die Funktionsweise der Netzwerkprotokolle zu verstehen – und das ist das Ziel dieses Buchs.

Dieses Buch ist für jeden geeignet, der verstehen möchte, wie die TCP/IP-Protokolle funktionieren: Programmierer, die Netzwerkanwendungen schreiben wollen, Systemadministratoren, die für den Betrieb von Computersystemen und Netzwerken auf der Basis von TCP/IP zuständig sind, aber auch für Benutzer, die im Alltag mit TCP/IP-Anwendungen zu tun haben.

## Aufbau dieses Buchs

Die folgende Abbildung zeigt die verschiedenen Protokolle und Applikationen, die wir besprechen. Die Nummer in Schrägschrift neben jedem Kästchen verweist auf

das entsprechende Kapitel, in dem das Protokoll oder die Applikation besprochen wird.



(In dieser Abbildung fehlen natürlich viele Details, auf die wir in den entsprechenden Kapiteln eingehen werden. Zum Beispiel benutzen sowohl DNS als auch RPC TCP, was hier nicht gezeigt wird).

Wir werden TCP/IP von der Pike auf kennen lernen. Nach einer allgemeinen Einführung in TCP/IP in Kapitel 1 beginnen wir in Kapitel 2 mit der Sicherungsschicht (Link Layer) und arbeiten uns nach oben im Protokoll-Stack. Damit erhalten Leser, für die TCP/IP oder Netzwerke im Allgemeinen neu sind, eine solide Basis für die Besprechung der Applikationen in den späteren Kapiteln.

Dieses Buch geht funktionsbezogen vor, statt sich stur von unten nach oben durch Schichtenmodelle zu bewegen. Zum beispielsweise werden in Kapitel 3 die IP-Schicht und der IP-Header beschrieben. Allerdings enthält der IP-Header viele Felder, die am besten im Kontext einer Applikation beschreiben lassen, die entweder ein bestimmtes Feld nutzt, oder von diesem Feld beeinflusst wird. Zum Beispiel kann man die Fragmentierung am besten im Kontext von UDP verstehen (Kapitel 11), dem Protokoll, das am häufigsten von der Fragmentierung betroffen ist. Das Time-to-Live-Feld wird im Kontext der Beschreibung des Traceroute-Programms in Kapitel 8 komplett ausführlich untersucht, weil dieses Feld die Basis für die Funktionalität von Traceroute bildet. Analog dazu werden viele Merkmale von ICMP in den späteren Kapiteln beschrieben, wenn wir erfahren, wie eine bestimmte ICMP-Nachricht von einem Protokoll oder einer Applikation benutzt wird.

Wir wollen die guten Sachen aber nicht alle bis zum Schluss aufheben und daher gehen wir, sobald das zum Verstehen dieser Applikation erforderliche Basiswissen gesichert ist, auf TCP/IP-Applikationen ein. Ping und Traceroute werden direkt im Anschluss an IP und ICMP beschrieben. Die Applikationen, die auf UDP aufbauen (Multicasting, DNS, TFTP und BOOTP) werden sofort nach der Untersuchung von UDP besprochen. Die TCP-Applikationen müssen wir jedoch genau wie das Netzwerk-Management bis zum Schluss aufheben, d.h. bis nach der detaillierten Besprechung von TCP. Der Text konzentriert sich auf die Frage, wie diese Applikation die TCP/IP-Protokolle nutzen: Einzelheiten zum Betrieb der Applikationen finden Sie hier jedoch nicht.

## Leser

Dieses Buch setzt keine weiteren Quellen und keine spezifischen Kenntnisse im Bereich Netzwerke oder TCP/IP voraus. Leser, die sich jedoch für weitere Einzelheiten zu bestimmten Themen interessieren, finden allerdings Querverweise.

Es gibt viele Arten, dieses Buch zu nutzen. Sie können es als Referenz zum Selbststudium nutzen; Leser, die sich für alle Einzelheiten der TCP/IP-Protokollsuite interessieren, können es am besten von vorne bis hinten durchlesen. Leser, die bereits über TCP/IP-Hintergrundwissen verfügen, können vielleicht sofort mit der Lektüre von Kapitel 7 beginnen und sich dann auf bestimmte Kapitel konzentrieren, für die sie sich besonders interessieren. Am Ende von jedem Kapitel finden Sie Übungen und für einen Großteil dieser Übungen finden Sie die Lösungen im Anhang D. Dieses Merkmal dient dazu, das Nutzen dieses Texts als Referenz zum Selbststudium zu maximieren.

Wenn dieses Buch als Teil eines ein oder zwei Semester dauernden Kurses über Computernetzwerke eingesetzt wird, sollte man sich auf IP (Kapitel 3 und 9), UDP (Kapitel 11) und TCP (Kapitel 17 bis 24) sowie auf einige der Kapitel zu Applikationen konzentrieren.

Im gesamten Text finden Sie Querverweise, außerdem enthält das Buch einen ausführlichen Index, was die Lektüre einzelner Kapitel ebenfalls ermöglicht. Eine Liste der Akronyme, die im gesamten Text vorkommen, mit Erläuterungen finden Sie auf der Innenseite des Einbands vorne und hinten.

Wenn Sie selbst Zugriff auf ein Netzwerk haben, können Sie sich am besten die Software beschaffen, die wir in diesem Buch einsetzen (siehe Anhang F) und selbst experimentieren. Hands-on-Experimente mit den Protokollen liefern die meisten Erkenntnisse (und machen auch viel mehr Spaß).

## Testsysteme

Jedes Beispiel in diesem Buch wurde in einem tatsächlich existierenden Netzwerk nachvollzogen und die Ergebnisse in einer Datei gespeichert, die im Text abgebildet wird. Abbildung 1.11 zeigt eine Abbildung der verschiedenen Hosts, Router und

Netzwerke, die wir dabei benutzt haben (diese Abbildung finden Sie außerdem auf der Innenseite des Einbands vorne, sodass sie bei der Lektüre leicht auffindbar ist). Die Ansammlung von Netzwerken ist einfach genug gehalten, um eine unnötige Komplexität der Beispiele zu vermeiden und die Tatsache, dass vier Systeme als Router benutzt werden, ermöglicht uns die Darstellung von Routerfehlermeldungen.

Die meisten Systeme haben Namen, aus denen auf die dort eingesetzte Software geschlossen werden kann: `bsd`, `svr4`, `sun`, `solaris`, `aix`, `slip` und so weiter. Auf diese Art können wir die Art der Software, mit der wir es zu tun haben, durch die Untersuchung des Systemnamens im gedruckten Output erkennen.

Wir nutzen eine große Vielfalt an Betriebssystemen und TCP/IP-Implementierungen:

- BSD/386 Version 1.0 von Berkeley Software Design, Inc., auf den Hosts `bsd` und `slip`. Dieses System wurde von der BSD Networking Software Release 2.0 abgeleitet (wir zeigen die Abstammung der verschiedenen BSD-Releases in Abbildung 1.10).
- UNIX System V/386 Release 4.0 Version 2.0 von U.H. Corporation auf dem Host `svr4`. Es handelt sich um ein Standard-SVR4-System mit einer Standard-TCP/IP-Implementierung von Lachmann Associates, die mit den meisten Versionen von SVR4 zum Einsatz kommt.
- SunOS 4.1.3 von Sun Microsystems auf dem Host `sun`. Die SunOS 4.1.x-Systeme haben wahrscheinlich die größte Verbreitung aller TCP/IP-Implementierungen. Der TCP/IP-Code stammt von 4.2BSD und 4.3BSD ab.
- Solaris 2.2 von Sun Microsystems auf dem Host `solaris`. Solaris 2.x-Systeme haben eine andere TCP/IP-Implementierung als die früheren SunOS 4.1.x-Systeme und SVR4 (dieses Betriebssystem ist in Wirklichkeit SunOS 5.2, aber wird typischerweise Solaris 2.2. genannt).
- AIX 3.2.2 von IBM auf dem Host namens `aix`. Diese TCP/IP-Implementierung basiert auf dem 4.3BSD Reno-Release.
- 4.4BSD von der Computer Systems Research Group an der University of California at Berkeley auf dem Host `vangogh.cs.berkeley.edu`. Dieses System hat die neueste Berkeley TCP/IP-Release (dieses System wird nicht im Diagramm auf der Innenseite des Umschlags vorne abgebildet, ist aber über das Internet ansprechbar).

Obwohl es sich bei diesen Systemen um UNIX-Systeme handelt, ist TCP/IP betriebssystemunabhängig und ebenfalls für fast alle gängige Nicht-UNIX-Systeme verfügbar. Viele Einzelheiten in diesem Text gelten genauso für Nicht-UNIX-Implementierungen, obwohl manche Applikationen (wie beispielsweise `Trace-route`) unter Umständen nicht auf allen Systemen zur Verfügung stehen).

## Typografische Konventionen

Wenn interaktive Ein- und Ausgaben dargestellt werden, wird die Eingabe **fett** dargestellt und die Ausgabe wie folgt. *Kommentare stehen in Kursivschrift.*

<b>bsdi % telnet svr4 discard</b>	<i>Verbindung zum discard-Server aufbauen</i>
<b>Trying 140.252.13.34...</b>	<i>diese Zeile und die</i>
	<i>nächste wurden vom Telnet-</i>
<b>Connected to svr4.</b>	<i>Client ausgegeben</i>

Außerdem wird der Name des Systems grundsätzlich als Shell-Prompt angezeigt (bsdi im vorherigen Beispiel), um zu zeigen, auf welchem Host der Befehl ausgeführt wurde.

Im gesamten Text kommen Notizen vor, die wie diese historische Punkte oder Einzelheiten der Implementierung hervorheben.

Manchmal beziehen wir uns auf die vollständige Beschreibung eines Befehls im UNIX-Manual wie beispielsweise `ifconfig(8)`. Diese Schreibweise mit dem Befehlsnamen gefolgt von einer Zahl in Klammern ist die typische Art, UNIX-Befehle zu bezeichnen. Die Zahl in Klammern bezeichnet den Abschnitt im UNIX-Manual für die *manpage* des Befehls, auf der weitere Informationen zu finden sind. Leider organisieren nicht alle UNIX-Systeme ihre *manpages* gleich, was die Abschnittsnummern der verschiedenen Befehlsgruppierungen betrifft. In diesem Buch nutzen wir BSD-typische Abschnittsnummern (die für BSD-Derivate wie SunOS 4.1.3 ebenfalls gelten), aber Ihre *manpages* können anders organisiert sein.

## Danksagungen

Obwohl nur der Name des Autors auf dem Einband des Buchs zu lesen ist, sind die gemeinsamen Bemühungen vieler Menschen erforderlich, um ein qualitativ hochwertiges Lehrbuch zu erzeugen. Vor allem Danke an meine Familie, die sich mit den langen und ungewöhnlichen Stunden abfinden mussten, die mit dem Schreiben eines Buchs einhergehen. Noch einmal Danke Sally, Bill, Ellen und David.

Mein beratender Lektor Brian Kernighan ist zweifelsohne der beste dieser Branche. Er hat als erste die verschiedenen Entwürfe dieses Manuskriptes gelesen und mit seinem schier endlosen Vorrat an Rotstiften bearbeitet. Sein Auge für Details, seine ständige Nachfrage nach lesbarem Material und seine gründliche Kritiken des Manuskripts sind ein wichtiger Beitrag für den Autor.

Die technischen *Reviewer* lieferten eine andere Betrachtungsweise und sorgten dafür, dass der Autor ehrlich geblieben ist, indem sie technische Fehler aufgedeckt haben. Ihre Kommentare, Vorschläge und (was noch wichtiger ist) Kritiken waren wichtige Beiträge zum fertigen Produkt. Mein Dank gilt Steve Bellovin, Jon Crowcroft, Pete Haverlock und Doug Schmidt für ihre Rezensionen des kompletten

Manuskripts. Genauso wertvoll waren die Kommentare zu bestimmten Abschnitten des Manuskripts von Dave Borman, Tony DeSimone, Bob Gilligan, Jeff Gitlin, John Gulbenkian, Tom Herbert, Mukesh Kacker, Barry Margolin, Paul Mockapetris, Burr Nelson, Steve Rago, James Risner, Chris Walquist, Phil Winterbottom und Gary Wright. Mein besonderer Dank gilt Dave Borman für seine ausführliche Überprüfungen aller TCP-Kapitel und Bob Gilligan, der als Co-Autor für Anhang E aufgeführt werden sollte.

Ein Autor kann nicht in völliger Isolierung arbeiten, und daher möchte ich mich bei den folgenden Menschen für die vielen kleinen Gefallen bedanken, vor allem für die Beantwortung meiner E-Mail-Fragen: Joe Godsil, Jim Hogue, Mike Karels, Paul Lucchina, Craig Partridge, Thomas Skibo und Jerry Toporek.

Dieses Buch ist das Ergebnis vieler Fragen zu TCP/IP, die mir gestellt wurden, und die ich nicht sofort und schnell beantworten konnte. Diese Fragen brachten mich zur Erkenntnis, dass man die Antworten am besten erhält, indem man kleine Tests durchführt, bestimmte Bedingungen erzwingt und die Ergebnisse einfach beobachtet. Ich bedanke mich bei Pete Haverlock für seine kniffligen Fragen und bei Van Jacobson für die Bereitstellung eines Großteils der öffentlich verfügbaren Software, die für die Beantwortung vieler Fragen in diesem Buch eingesetzt wurde.

Ein Buch über Netzwerke braucht ein echtes Netzwerk und Zugriff auf das Internet, wenn es glaubwürdig erscheinen soll. Ich bedanke mich in diesem Zusammenhang bei den National Optical Astronomy Observatories (NOAO), vor allem bei Sidney Wolff, Richard Wolff und Steve Grandi, die Konten für Ihre Netzwerke und Hosts bereit gestellt haben. Besonderen Dank gilt Steve Grandi, der viele Fragen beantwortet und Konten auf verschiedenen Hosts bereit gestellt hat. Danke auch an Keith Bostic und Kirk McKusick an der U.C. Berkeley CSRG für den Zugriff auf das neueste 4.4BSD-System.

Nicht zuletzt ist es der Verlag, der alle Fäden zusammenzieht und eben das Notwendige unternimmt, um das endgültige Produkt an die Leser zu bringen. Und in diesem Fall laufen alle Fäden beim Redakteur, John Wait, zusammen – in meinen Augen, der beste seiner Zunft. Die Arbeit mit John und den anderen Fachleuten von Addison-Wesley war ein Vergnügen. Ihre Professionalität und ihr Auge für das Detail spiegeln sich im Endergebnis wider.

Eine reprofähige Vorlage für das Buch wurde vom Autor selbst – einem der letzten Troff-Anhänger – mithilfe des Groff-Pakets von James Clark hergestellt. E-Mail von allen Lesern mit Kommentaren, Vorschlägen und Bugfixes ist willkommen.

*Tucson, Arizona*

*Oktober 1993*

*W. Richard Stevens*

`rstevens@noao.edu`

`http://www.noao.edu/~rstevens`