

Einführung

Das *Internet* ist in seiner heutigen Form ein weltumspannendes Rechnernetz, das sich selbst aus einer Menge einzelner, voneinander unabhängiger Netzwerke zusammensetzt. Der Begriff Internet ist dabei als Kurzschreibweise für *Interconnected Networks*, also miteinander verbundene Netzwerke, entstanden. Angefangen bei einzelnen Rechnersystemen, die zu einem *Local Area Network (LAN)* zusammengeschlossen sind, werden mehrere LANs über größere Distanzen hinweg zu einem *Wide Area Network (WAN)* verknüpft. Der verschachtelte Verbund vieler derartiger WANs ergibt schließlich das Internet.

Natürlich hatte zu den Anfangszeiten des Internets niemand die kühne Vision, ein Rechnernetz bestehend aus knapp 400 Millionen¹ miteinander vernetzter Computersysteme zu initiieren. Vielmehr sollte gegen Ende der 60'er Jahre im Auftrag der *Advanced Research Projects Agency (ARPA)*, die damals für das US-amerikanische Verteidigungsministerium Forschungsprojekte förderte, ein Netzwerk entstehen, über das die damals knappen Rechenkapazitäten der Hochschulen durch den Austausch von Daten besser ausgenutzt werden sollten. Das resultierende Netzwerk hieß zu dieser Zeit auch noch nicht Internet, sondern *ARPANET*, und wurde Ende 1969 von der *University of California, Los Angeles*, der *University of California, Santa Barbara*, der *University of Utah* und dem *Stanford Research Institute* in Betrieb genommen.² Das Netz verband zunächst genau vier Rechner der vier beteiligten Universitäten.

Um eine möglichst hohe Ausfallsicherheit zu erreichen, ersannen die beteiligten Forscher ein *paketvermitteltes Netzwerk*. In paketvermittelten Netzen werden die zu übertragenden Daten vom Absender in einzelne Pakete zerlegt und vom Empfänger nach dem Eintreffen wieder zusammengesetzt. In einem komplexen Netzwerk können die einzelnen Pakete bei der Übertragung

¹ Stand: Januar 2006, siehe <http://www.isc.org/ds/>

² Der erste Datenaustausch soll laut [ZEI01] am 29. Oktober 1969 stattgefunden haben, um die Buchstabenkombination *LOG* zu übermitteln. Von den gleichzeitig telefonierenden Technikern werden die Worte „Hast du das L?“ – „Ja!“ – „Hast du das O?“ – „Ja!“ – „Hast du das G?“ überliefert, dann sei der Rechner abgestürzt.

durchaus unterschiedliche Wege gehen. Absender und Empfänger sind also, im Gegensatz zu *leitungsvermittelten Netzwerken* mit fester Verbindung, lediglich lose miteinander verbunden.

Weitere Forschungsarbeiten führten bis 1974 zur Entwicklung von TCP/IP, einer Familie von Netzwerkprotokollen, die bis heute die Basis für den Datenaustausch über das Internet bilden. Um die Akzeptanz von TCP/IP zu forcieren, wurde die *University of California at Berkeley* damit beauftragt, TCP/IP in *Berkeley Unix* zu integrieren. Mit der zunehmenden Verbreitung von TCP/IP wuchs gleichzeitig die Anzahl der Rechner im ARPANET rapide an. Der große Verbund von Netzen, den das ARPANET nun langsam aber sicher darstellte, wurde schließlich als *das Internet* bekannt. TCP/IP wird seitdem auch als Internet-Protokoll-Familie bezeichnet.

1.1 TCP/IP-Grundlagen

Um die komplexen Zusammenhänge, die in einem Rechnernetz bestehen, besser strukturieren und beschreiben zu können, wurden sogenannte Referenzmodelle eingeführt. Das bekannteste dieser Modelle ist das von der *International Organization for Standardization (ISO)* standardisierte *OSI-Referenzmodell* (*Open Systems Interconnection Reference Model*). Nachdem die Internet-Protokoll-Familie bereits vor dem OSI-Referenzmodell entwickelt wurde, liegt der TCP/IP-Architektur ein anderes, vereinfachtes Referenzmodell zugrunde. Die Erfahrungen mit dem TCP/IP-Referenzmodell sind dann bei der Ausarbeitung des OSI-Referenzmodells eingeflossen.

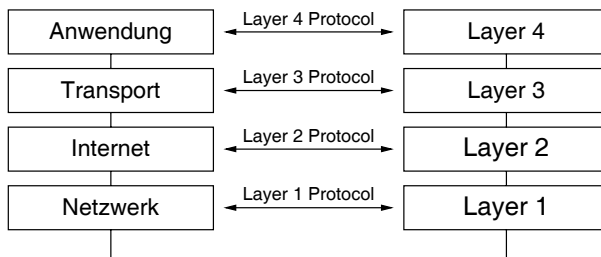


Abb. 1.1. Kommunikation im TCP/IP-Referenzmodell

Das TCP/IP-Referenzmodell unterteilt die einzelnen Aufgabenstellungen der Netzwerkkommunikation, angefangen bei der physikalischen Übertragung der Datensignale über die Vermittlung von Datenpaketen bis hin zu speziellen, anwendungsspezifischen Aufgaben, in einen Stapel von vier Schichten. Die einzelnen Schichten des Modells sind in Abb. 1.1 zu sehen. Jede der Schichten im Referenzmodell definiert bestimmte Funktionen, die in Form von Diensten und Protokollen implementiert werden:

- Jede Schicht kann die Dienste der darunterliegenden Schicht nutzen, ohne dabei konkrete Kenntnisse darüber besitzen zu müssen, wie die in Anspruch genommenen Dienstleistungen genau erbracht werden bzw. implementiert sind. Sämtliche Daten, die von einer Anwendung über das Netzwerk verschickt werden, durchlaufen von der Anwendungs- bis zur Netzwerkschicht alle Schichten des Modells (und auf dem empfangenden System wieder in der umgekehrten Reihenfolge).
- Die gleichnamigen Schichten zweier kommunizierender Systeme kooperieren miteinander jeweils über spezielle, schichtspezifische Protokolle. Dazu werden den von einer Anwendung verschickten Daten in jeder Schicht protokollspezifische Informationen hinzugefügt (und von der gleichnamigen Schicht auf dem empfangenden System wieder entnommen).

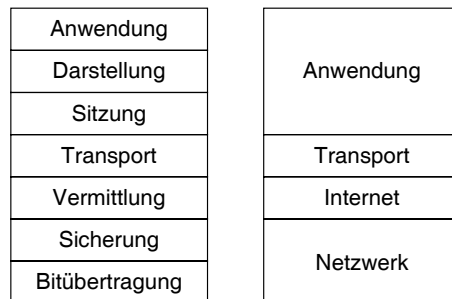


Abb. 1.2. OSI- und TCP/IP-Referenzmodell im Vergleich

Im direkten Vergleich zwischen OSI- und TCP/IP-Referenzmodell (vgl. dazu Abb. 1.2) zeigt sich die etwas feinere Untergliederung des OSI-Modells in einen Stapel von insgesamt sieben Schichten zusammen mit der korrespondierenden Aufteilung des TCP/IP-Modells.

1.1.1 Netzwerkschicht

Die beiden untersten Schichten des OSI-Referenzmodells repräsentieren grob gesprochen die Netzwerk-Hardware und die zugehörigen Gerätetreiber. Sie kapseln die eingesetzten Netzwerktechnologien wie z. B. *Ethernet*, *Token Ring* oder *FDDI* und liefern den übergeordneten Schichten eine davon unabhängige Schnittstelle zur Kommunikation. Diese beiden Schichten sind in der *Netzwerkschicht* des TCP/IP-Referenzmodells zusammengefaßt und spielen bei der Entwicklung netzwerkfähiger Client-/Server-Programme im Allgemeinen keine Rolle.

1.1.2 Internet-Schicht

Der Vermittlungsschicht des OSI-Modells entspricht im TCP/IP-Modell die *Internet-Schicht*. Diese Schicht kümmert sich in paketvermittelten Netzwerken wie dem Internet um die Weitervermittlung der einzelnen Datenpakete. Die Internet-Schicht etabliert damit eine Rechner-zu-Rechner-Verbindung, unabhängig von der zugrundeliegenden Netzwerkstruktur (bestehend aus zahlreichen LANs und WANs). Dadurch befreit die Schicht die übergeordneten Schichten von den Details der Datenübertragung.

Internet Protocol (IP)

Die Internet-Schicht wird bei TCP/IP vom *Internet Protocol (IP)* besetzt. Die wesentliche Aufgabe dieses in RFC 791 [IP81] spezifizierten Protokolls ist die Adressierung von Netzteilnehmern über *IP-Adressen* und die Datenübertragung von und zu anderen Teilnehmern. Die Daten werden dabei in einem vom Internet Protocol definierten Paketformat als sogenannte *Datagramme* an die tieferliegenden Schichten übergeben und über das Netzwerk übertragen. Bei Bedarf werden die Datagramme vom Internet Protocol fragmentiert, d. h. vor dem Versand in kleinere Fragmente zerlegt und nach dem Empfang wieder zusammengesetzt.

Beim Internet Protocol handelt es sich um ein *verbindungsloses Protokoll*, d. h. es existiert auf der IP-Ebene keine Ende-zu-Ende-Verbindung zwischen den kommunizierenden Anwendungen. Die einzelnen IP-Datagramme werden unabhängig von den jeweils anderen Datagrammen zugestellt, insbesondere kann sich dabei die Empfangsreihenfolge der Datagramme von der Reihenfolge beim Versand unterscheiden. Darüber hinaus wird der Empfang der einzelnen IP-Datagramme nicht bestätigt. Das Internet Protocol bietet also bezüglich der Datenübertragung keine Zuverlässigkeitsgarantie, weshalb das Protokoll auch oft als *nicht zuverlässig* oder sogar als *unzuverlässig* bezeichnet wird.³ Das Internet Protocol kann demnach die tatsächliche Zustellung der verschickten Datagramme nicht garantieren, die einzelnen IP-Datagramme werden lediglich bestmöglich verschickt.

Internet Control Message Protocol (ICMP)

Das in RFC 792 [Pos81] spezifizierte *Internet Control Message Protocol (ICMP)* ist ebenfalls auf der Internet-Schicht angesiedelt. Das Protokoll ist

³ Das Attribut *unzuverlässig* bedeutet in diesem Fall nicht, daß beim Internet Protocol ständig etwas schief läuft und man sich deshalb generell nicht darauf verlassen kann. Die Bezeichnung weist vielmehr darauf hin, daß das Protokoll selbst keine Mechanismen zur Erkennung und Behebung von Paketverlusten besitzt, so daß diese Aufgaben bei Bedarf von höheren Schichten übernommen werden müssen.

integraler Bestandteil jeder IP-Implementierung und transportiert Status-, Fehler- und Diagnoseinformationen für das Internet Protocol. Die ICMP-Pakete werden dazu als IP-Datagramme über das Netzwerk übertragen.

1.1.3 Transportschicht

Die *Transportschicht* ist die erste (niedrigste) Schicht, die den übergeordneten anwendungsorientierten Schichten, also den Schichten 5–7 des OSI-Modells bzw. Schicht 4 des TCP/IP-Modells, eine vollständige Ende-zu-Ende-Kommunikation zwischen zwei Anwendungen zur Verfügung stellt. Die Transportschicht leitet den Datenfluß von der Anwendung zur Internet-Schicht und umgekehrt. Die Adressierung der Anwendung erfolgt dabei über einen sogenannten *Port*, an den die Anwendung zuvor gebunden wurde.

User Datagram Protocol (UDP)

Genau wie das zugrundeliegende Internet Protocol ist auch das *User Datagram Protocol (UDP)* ein verbindungsloses Transportprotokoll ohne Zuverlässigkeitsgarantie.⁴ Die Spezifikation des Protokolls findet sich in RFC 768 [Pos80]. Das User Datagram Protocol erweitert die bereits vom Internet Protocol erbrachten Leistungen lediglich um die Portnummern der sendenden und empfangenden Anwendung. Die von UDP transportierten Pakete werden deshalb in Anlehnung an IP auch *UDP-Datagramme* genannt. UDP hat demzufolge auch nur einen minimalen *Protokoll-Overhead*, was das Protokoll v. a. für Anwendungen, die nur wenige Daten übertragen müssen, interessanter als das nachfolgend beschriebene TCP macht.

Soll auf Basis von UDP eine zuverlässige, reihenfolgetreue Datenübertragung erfolgen, so muß sich eine der übergeordneten Schichten (sprich: die Anwendung selbst) um diese Aufgabe kümmern.

Transmission Control Protocol (TCP)

Das ursprünglich in RFC 793 [TCP81] spezifizierte *Transmission Control Protocol (TCP)* ist, im Gegensatz zu UDP, ein verbindungsorientiertes und zuverlässiges Transportprotokoll. Das Protokoll stellt je zwei Kommunikationspartnern eine virtuelle Ende-zu-Ende-Verbindung (im Sinne einer festen Punkt-zu-Punkt-Verbindung in einem leitungsvermittelten Netzwerk) zur Verfügung. Die Datenübertragung erfolgt bei TCP deshalb in drei Phasen: dem Verbindungsaufbau, der eigentlichen Datenübertragung und dem

⁴ Bezüglich der (fehlenden) Zuverlässigkeitsgarantie gelten die selben Anmerkungen wie beim Internet Protocol (vgl. dazu Abschnitt 1.1.2).

abschließenden Verbindungsabbau. Die von TCP übermittelten Datenpakete werden *TCP-Segmente* genannt. Die Reihenfolge der über eine solche virtuelle Ende-zu-Ende-Verbindung verschickten TCP-Segmente bleibt im Rahmen der Übertragung erhalten. Nachdem das Protokoll zudem den Empfang eines TCP-Segments gegenüber dem Sender quittiert und der Sender beim Ausbleiben einer solchen Empfangsbestätigung die Übertragung des betreffenden Segments wiederholt, wird das Transmission Control Protocol als zuverlässiges Transportprotokoll bezeichnet.

TCP nimmt Daten von den übergeordneten Schichten als *Datenstrom* an und teilt diesen Datenstrom auf einzelne TCP-Segmente auf. Jedes dieser TCP-Segmente wird dann über die Internet-Schicht als IP-Datagramm verschickt. Umgekehrt werden die einer TCP-Verbindung zugeordneten IP-Datagramme bzw. TCP-Segmente auf der Empfängerseite wieder zum ursprünglichen Datenstrom zusammengesetzt. Nachdem das zugrundeliegende Internet Protocol bezüglich der Datenübertragung keine Zuverlässigkeitsgarantie übernimmt, muß sich das Transmission Control Protocol selbst um diese Aufgabe (wiederholtes Senden verlorengegangener Pakete, Einhalten der Reihenfolge) kümmern. Die TCP-Segmente erhalten dazu sogenannte *Sequenz-* und *Bestätigungsnummern*, mit deren Hilfe die Segmente quittiert, ggf. neu übertragen und in der korrekten Reihenfolge wieder zu einem Datenstrom zusammengesetzt werden können.

1.1.4 Anwendungsschicht

Die oberste Schicht des TCP/IP-Referenzmodells ist die *Anwendungsschicht*. Sie faßt die Schichten 5–7 des OSI-Modells in einer Schicht zusammen. Für diese Schicht ist inzwischen eine Fülle von Anwendungsprotokollen spezifiziert, über welche heute die gängigen Internetdienste miteinander kommunizieren. Stellvertretend seien an dieser Stelle die weithin bekannten Anwendungsprotokolle *Hypertext Transfer Protocol (HTTP)* für das *World Wide Web (WWW)* oder das *Simple Mail Transfer Protocol (SMTP)* zum Austausch *elektronischer Post (E-Mail)* genannt.

Die Anwendungsprotokolle bzw. die Anwendungen, die diese Anwendungsprotokolle implementieren, greifen ihrerseits über die *Socket-API* auf die Dienste der Transportschicht zu. Im weiteren Verlauf dieses Buchs kümmern wir uns nun im Wesentlichen um die Funktionalität der Socket-API und ihre Anwendung im Rahmen eigener Client-/Server-Programme.

1.2 Internet-Standards

Die sogenannten *RFCs (Request for Comments)*, von denen wir in diesem Kapitel bereits mehrere zitiert haben, spielten und spielen bei der Entstehung

und Weiterentwicklung des Internets eine gewichtige Rolle. Dabei handelt es sich um eine fortlaufend ergänzte Reihe von technischen und organisatorischen Dokumenten, über welche die Standards für das Internet festgelegt werden. Sie beschreiben die Dienste und Protokolle des Internets und legen Regeln und Grundsätze für dieses Netzwerk fest. Die Sammlung aller RFCs wird an zahlreichen Stellen im Internet publiziert, u. a. auf der Homepage des *RFC-Editors*.⁵

Neue RFC-Dokumente werden von einer Arbeitsgruppe bzw. dem RFC-Editor geprüft und durchlaufen dabei bestimmte Reifestufen. Ein einmal veröffentlichter RFC wird nie wieder verändert oder aktualisiert. Stattdessen wird er bei Bedarf durch einen neuen RFC ersetzt und erhält den Zusatz, daß er durch den neuen RFC abgelöst wurde. Der neue RFC enthält seinerseits einen Hinweis auf den RFC, den er abgelöst hat.

Neben ihrer verantwortungsvollen Aufgabe als eine Art Standardisierungsgremium beweisen RFC-Autoren mitunter auch sehr feinen Humor, bevorzugt am 1. April jeden Jahres. Stellvertretend für viele amüsante Exkursionen sei an dieser Stelle auf den lesenswerten und durchaus zu diesem Buch passenden RFC 1925 [Cal96], *The Twelve Networking Truths*, hingewiesen.

1.3 Unix-Standards

Die Entstehungsgeschichte von Unix reicht wie die Geschichte des Internets ins Jahr 1969 zurück. Ken Thompson begann damals bei den Bell Laboratories mit der Entwicklung des neuen Betriebssystems *UNICS*, einer in Assembler geschriebenen, abgespeckten Version des Betriebssystems *MULTICS*.⁶ Der Name UNICS wandelte sich im weiteren Projektverlauf zu Unix.⁷ Das Unix-System wurde dann Anfang der 70'er Jahre in der, im Rahmen der Unix-Entwicklung entstandenen, Programmiersprache C neu implementiert und wenig später zusammen mit einem C-Compiler kostenfrei an verschiedene Universitäten verteilt. In der Folgezeit entstanden viele variierende Systemlinien von Unix mit jeweils unterschiedlichen Kommandos, Kommandooptionen und Systemschnittstellen. So wurde z. B. von der *University of California at Berkeley* neben anderen Anpassungen auch TCP/IP in das dort entstandene *Berkeley Unix* integriert.

⁵ <http://www.rfc-editor.org/>

⁶ Übrigens: Der Name UNICS enthält gleich ein zweifaches Wortspiel: Zum einen dokumentiert *UNI* im Gegensatz zu *MULTI*, daß es sich bei UNICS um ein abgespecktes MULTICS handelt. Zum anderen wird UNICS in etwa so ausgesprochen wie das Wort *Eunuchs*, was UNICS als kastriertes MULTICS darstellt.

⁷ Streng genommen steht die Schreibweise *UNIX* für Unix-Systeme, deren Implementierung auf den ursprünglichen Unix-Quellen der Bell Laboratories basiert, während durch die Schreibweise *Unix* zusätzlich auch Unix-ähnliche Systeme, wie z. B. Linux, mit eingeschlossen werden.

Die vielen verschiedenen Unix-Derivate mit ihren gegenseitigen Inkompatibilitäten führten schließlich zu den ersten Standardisierungsbemühungen und es entstanden die sogenannten *POSIX-Standards*.⁸ Die aktuelle Version [SUS02], auf die sich die Ausführungen des vorliegenden Buchs beziehen, ist unter den drei Namen *IEEE Std 1003.1-2001*, *ISO/IEC 9945:2002* und *Single UNIX Specification, Version 3* bekannt. Wir werden im weiteren Verlauf den Namen IEEE Std 1003.1-2001 oder einfach nur kurz POSIX-Standard verwenden.

⁸ Das Akronym *POSIX* steht für *Portable Operating System Interface*, ergänzt um die Unix-typische Endung *IX*.

Unix-Netzwerkprogrammierung mit Threads, Sockets
und SSL

Zahn, M.

2006, XV, 434 S., Hardcover

ISBN: 978-3-540-00299-4