Grundlagen
„Internetprotokolle“

Zusammengestellt von Alexander Scheib
Stand: 24.08.02

Geschichte und Entwicklung

Die TCP/IP-Protokollfamilie hat seit der Erfindung des *World Wide Web* (WWW) weltweit sehr stark an Bedeutung zugenommen und sich als De-facto-Standard im Bereich der Kommunikation zwischen Rechnern verschiedener Hersteller und Firmen durchgesetzt.

ARPANET

Ende der 60-er Jahre erkannte man beim *Department of Defence* (DoD) die Wichtigkeit, offene Netze für den Datenaustausch zu haben. 1969 entwickelte deshalb die *Defense Advanced Research Project Agency* (DARPA) im Auftrag des DoD ein Computernetz mit dem Namen ARPANET . Das Netzwerk sollte einen Datenaustausch ermöglichen, welcher nicht zentral an einzelnen Punkten geführt wird, sondern durch mehrere verschiedene mögliche Wege ausfallsicherer werden sollte.

Das erste Protokoll, welches im ursprünglichen ARPANET eingesetzt wurde, hiess *Network Control Protocol* (NCP, Netzsteuerungsprotokoll). Das Protokoll gab Nachrichtenpakete an die Vermittlungsschicht weiter und nahm an, dass sie korrekt am Ziel ankamen (verbindungsloses Protokoll). 1972 wurde das ARPANET öffentlich präsentiert.

Bald schon entwickelte sich das ARPANET weiter und dehnte sich auf mehrere verschiedenartige Subnetze (Paketfunk, Satellitenkanäle, LANs) mit insgesamt ca. 750 angeschlossenen Computern aus. Damit wandelte sich auch der Name von ARPANET zu ARPA Internet. Die Ende-zu-Ende-Zuverlässigkeit verringerte sich zusehends. Deshalb waren die Entwickler gezwungen, grössere Veränderungen an der Transportschicht vorzunehmen. Dies führte dann 1974 zu einem neuen Protokoll für die Transportschicht: das *Transmission Control Protocol* (TCP, Übertragungssteuerungsprotokoll). Entsprechend dem TCP wurde auch ein neues Protokoll für die Vermittlungsschicht entworfen: das *Internet Protocol* (IP). Damit waren die Grundzüge der TCP/IP Protokolle und der Netzwerkarchitektur niedergelegt:

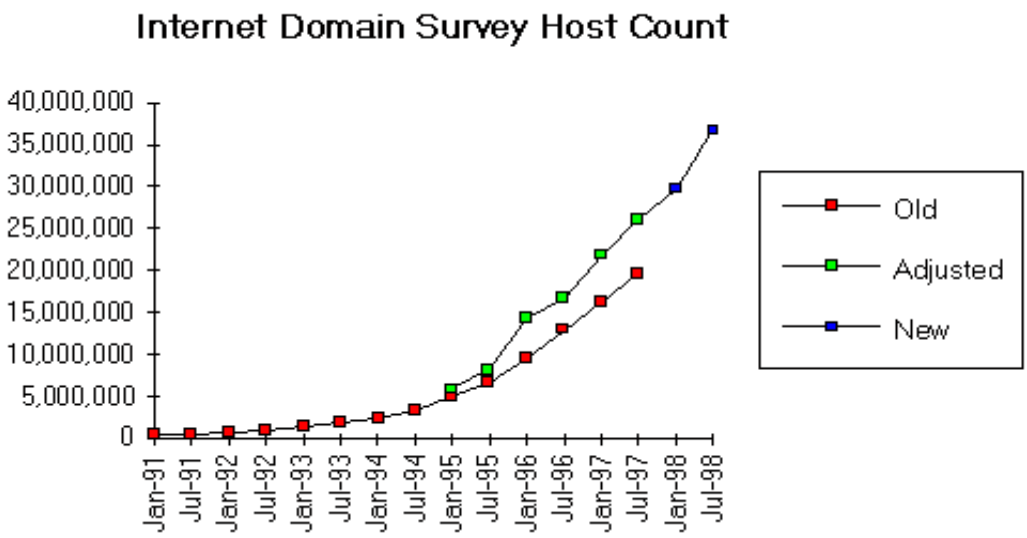
- Unabhängigkeit von der verwendeten Netzwerktechnologie und den Rechnersystemen
- Universelle Verbindungsmöglichkeit im ganzen Netz
- End- zu End Quittungen (Verbindungskontrolle)
- Standardisierte Anwendungsprotokolle
- Sichere Übertragung auch über instabile Verbindungswege

Mit der Zeit schlossen sich verschiedene Universitäten und Forschungseinrichtungen an das Netz an. An den Universitäten wurde die Protokollfamilie übernommen. Die Universität von Berkley implementierte die TCP/IP Protokollfamilie auf ihrem Unix. Mit der Verbreitung der Berkley Software Distribution UNIX 4.2 ab 1983 fand TCP/IP auch ausserhalb von Regierungsprojekten immer mehr Verbreitung und wurde rasch zum Kommunikationsstandard zwischen Unix-Systemen.

Heute findet man Implementationen von TCP/IP für praktisch sämtliche Betriebssysteme, wie VAX/VMS, IBM MVS, AOS/VS, OS/2, MS-DOS, Windows, usw. PCs, Workstations, Minis und Mainframes lassen sich mit Hilfe der TCP/IP-Protokollfamilie vernetzen. Weil die Protokolle bewusst auf mehrere Netzwerkarchitekturen (Ethernet, Tokenring, X.25, FrameRelay, usw.) ausgelegt sind, ist es möglich, praktisch sämtliche "Systemwelten" miteinander zu vernetzen. So wird heute vermehrt in Unternehmungen aus Kostengründen und zur Verringerung der Abhängigkeit von einzelnen Lieferanten und Spezialisten der Einsatz von nur noch einer Protokollfamilie (TCP/IP) angestrebt.

Internet

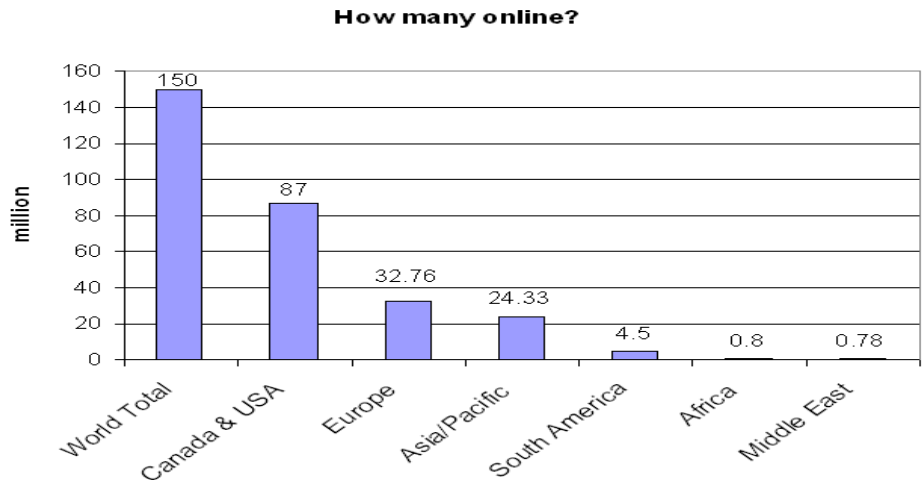
Im Jahre 1990 bestand das Internet noch aus etwa 3000 lokalen Netzwerken (LAN) mit 200'000 eingebundenen Computern. 1992 betrug die Zahl der angeschlossenen Rechner bereits 727'000. Abbildung 1 zeigt deutlich das exponentielle Wachstum des Internets. Heute werden bereits gegen 40 Mio. Hosts auf dem Internet gezählt.



Anzahl Internet Hosts

Dazu kommen noch die unzählbar vielen Rechner, welche über Dial Up zu einem Provider auf das Internet zugreifen.

Es ist nicht möglich, eine genaue Zahl anzugeben, wieviele Benutzer auf der ganzen Welt Internet nutzen. Grund dafür sind einerseits die vielen Benutzer, welche über Dial Up auf das Internet zugreifen und andererseits die Rechner, welche hinter Firewalls liegen und mit einem Suchprogramm nicht entdeckt werden. Die Internet-Beratungsfirma Nua hat versucht mit einer Vielzahl von Umfragen und Messungen eine 'gute Schätzung' zu machen. Dabei kam sie auf die Zahl von ca. 150 Mio. Benutzern weltweit (Abbildung 2).



geschätzte Anzahl Online-User auf dem Internet im November 1998

Weitere Statistiken können unter folgender Adresse gefunden werden:
http://www.isoc.org/internet/stats/

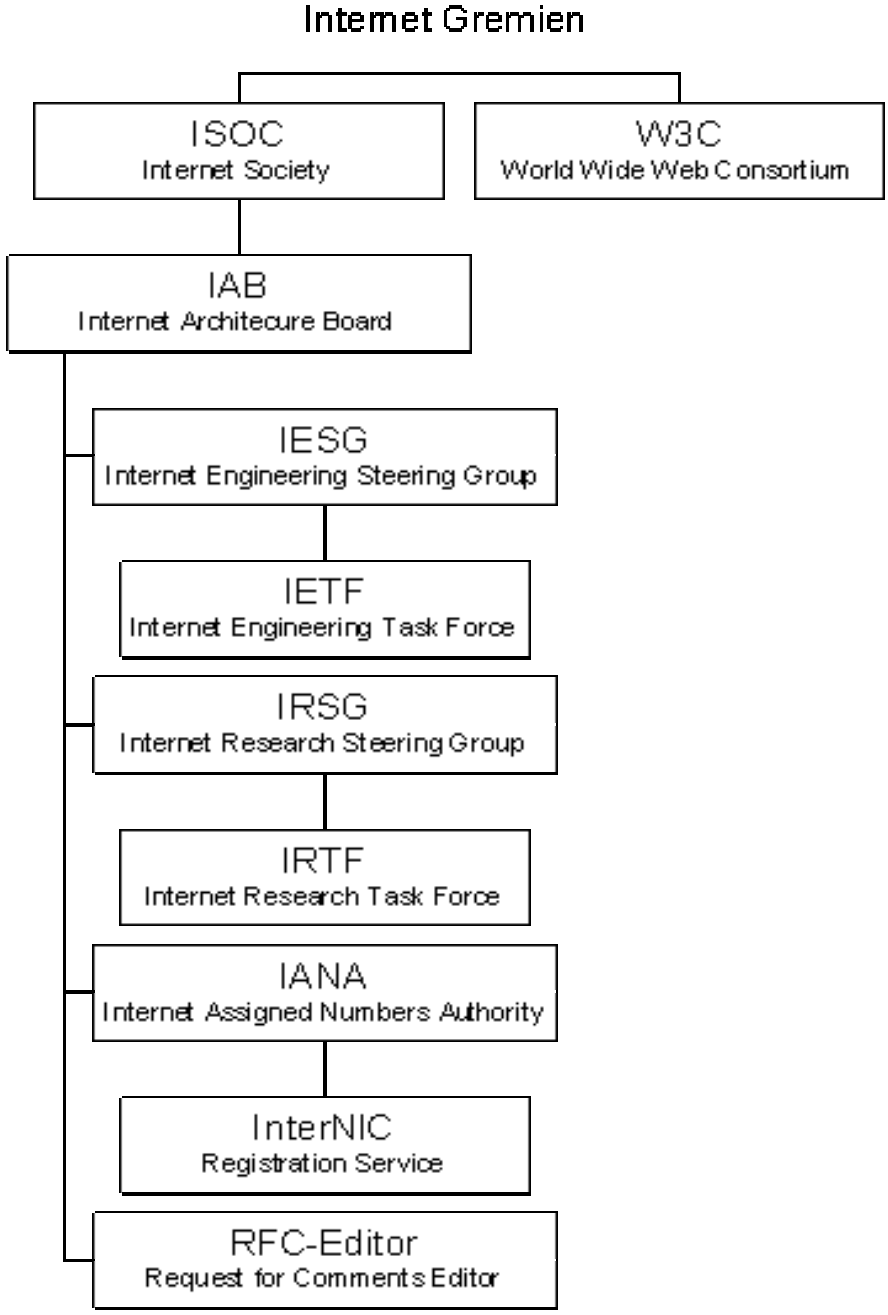
Folgende Abbildung zeigt die grössten Internet Provider in Europa und deren Verbindungen untereinander:

Intranet

Mit der Einführung von TCP/IP in Unternehmungen sind die Unternehmensnetze zu kleinen Internets geworden. Diese firmeninternen Internets werden Intranet genannt. Um die Firmendaten vor Angriffen und unberechtigten Lesern zu schützen, werden diese Firmen-Internets mittels einem speziellen Router vom Internet getrennt. Dieser Router trennt also das Intranet vom Internet.

Internet Gremien

Das Internet besteht aus einer grossen Menge von Netzen, welche von verschiedenen Organisationen und Firmen betreut werden. Es gibt keine Organisation, die für das gesamte Internet als Einheit verantwortlich ist. Trotzdem bestehen Organisationen, die bestimmte Aufgaben für das ganze Netz übernehmen.



Hierarchien und Zusammenhang der Internet Gremien

Internet Society (ISOC)
Die *Internet Society* (ISOC, Internet Gesellschaft/Gemeinschaft) ist eine berufsmässige Gesellschaft, die sich befasst mit:
dem Wachstum und der Entwicklung des weltweiten Internet
der Art, wie das Internet benutzt wird und werden kann
sozialen, politischen, und technischen Fragen im Zusammenhang mit dem Internet
Die ISOC ernennt die Mitarbeiter des IAB, die der Nominierungs-Ausschuss des IETF vorschlägt.

Internet Architecture Board (IAB)
Das *Internet Architecture Board* (IAB, früher "Internet Activity Board") ist das Koordinationskomitee für die Planung, den Aufbau und das Management des gesamten Internet. Alle Entscheidungen, die das IAB trifft, werden in den sog. *Request for Comments* (RFC) veröffentlicht.
Das IAB umfasst zwei spezialisierte Untergruppen, die *Internet Engineering Task Force* (IETF) und die *Internet Research Task Force* (IRTF). Diese beiden Gremien sind für die Weiterentwicklung und Definition der kurz- und mittelfristigen Internet-Protokolle und -Architekturen für die Bereiche Applikationen, Host- und Benutzerservice, Internet-Service, Routing, Adressierung, Netzwerkmanagement, Operations, Security und OSI-Integration verantwortlich. Sie erarbeiten die für das IAB zum Entscheid notwendigen Grundlagen.
Das IAB ist eine technische Beratungsgruppe der *Internet Society* (ISOC). Seine Verantwortungen schliessen ein:

IESG Auswahl: Das IAB ernennt den IESG Vorsitzenden und die Mitarbeiter der IESG, die der Nominierungs-Ausschuss der IETF vorschlägt.

Architekturübersicht: Das IAB stellt eine Übersicht der Architektur der Protokolle und Verfahren bereit, die vom Internet benutzt werden.
Massstäbe für Entwicklung und Dokumentation: Das IAB stellt Verfahren für das Erarbeiten von Internet Standards bereit. Es nimmt auch Klagen wegen schlechter Umsetzung der Standards entgegen.
RFC Serien und IANA: Das IAB ist verantwortlich für das redaktionelle Management und die Veröffentlichung der *Request for Comments* (RFC) Dokument-Serien, und für die Verwaltung der verschiedenen *Internet Assigned Numbers* (IAN, Zuordnung der IP-Adressen).
Externe Verbindungen: Das IAB handelt als Vertreter der Interessen der *Internet Society* an Besprechungen mit anderen Organisationen betreffend Standards und anderen technischen und organisatorischen Fragen im Zusammenhang mit dem weltweiten Internet.
Beratung der ISOC: Das IAB berät das ISOC (freie Mitarbeiter und Vorstandsmitglieder der Internet Gesellschaft) in Fragen rund um das Internet, die technischer, architektonischer, verfahrensmässiger, und (wo angemessen) politischer Natur sind.

Internet Engineering Steering Group (IESG)
Die *Internet Engineering Steering Group* (IESG) ist verantwortlich für das technische Management der IETF Aktivitäten und für die Internet Standards. Im Rahmen des ISOC, verwaltet es den Prozess gemäss den Regeln und Verfahren, die von den ISOC Trustees ratifiziert worden sind. Das IESG ist direkt verantwortlich für die Handlungen, die mit Eintragung in und Ausführung der Internet Standards verbunden werden, einschliesslich abschliessender Zustimmung von Spezifikationen als Internet Standards.

Internet Engineering Task Force (IETF)
Die *Internet Engineering Task Force* (IETF) eine grosse offene internationale Gemeinde von Netzdesignern, -Operatoren, -Verkäufern, -Entwicklern, die sich für die Entwicklung der Internet Architektur und das gute funktionieren des Internets interessieren. Sie ist jedem Interessierten frei

zugänglich. Die IETF ist zuständig für das Internet Protokoll Engineering und die anschliessend notwendige Standardisierung.
Das IETF wird in folgende acht funktionale Gebiete geteilt::

1. Anwendungen (Applications)
2. Internet
3. IP: Next Generation
4. Netzwerkmanagement (Network Management)
5. Betriebliche Auflagen (Operational Requirements)
6. Routing
7. Sicherheit (Security)
8. Transport- und Anwenderdienste (Transport and User Services)

Jedes Gebiet hat ein oder zwei Gebiet-Direktoren. Die Gebiet-Direktoren zusammen mit dem IETF/IESG Vorsitzenden , bilden die IESG.
Jedes Gebiet hat mehrere *Working Groups* (Arbeitsgruppen). Eine Arbeitsgruppe ist eine Gruppe Leute, die auf ein ganz bestimmtes Ziel hin arbeiten. Das Ziel kann ein informatives Dokument, die Schöpfung einer Protokoll-Spezifizierung, oder der Entschluss von Problemen im Internet sein. Die meisten Arbeitsgruppen haben eine endliche Lebenszeit. Das heisst, sobald eine Arbeitsgruppe ihr Ziel erreicht hat, wird sie aufgelöst. Wie in der IETF, gibt es keine offizielle Mitgliedschaft für Arbeitsgruppen. Inoffiziell ist ein Mitglied einer Arbeitsgruppe meistens jemand, der die Mailing-Liste der Arbeitsgruppe abonniert hat; es ist jedoch jedermann erlaubt eine Arbeitsgruppensitzung zu besuchen.
Gebiete können auch *Birds of a Feather* (BOF) *sessions* haben (ad hoc Arbeitsgruppensitzungen). Sie haben die gleichen Ziele wie Arbeitsgruppen allgemein, ausgenommen , dass sie kein spezielles Ziel haben und sich gewöhnlich nur einmal oder zweimal treffen. BOFs werden oft gehalten, um zu entscheiden, ob genug Interesse zum Bilden einer Arbeitsgruppe vorhanden ist.

Die IETF wird im RFC 1718 vollständig beschrieben.

Internet Research Steering Group (IRSG)

Die *Internet Research Steering Group* (IRSG) "steuert" die Arbeiten der IRTF. Die Mitgliedschaft der IRSG beinhaltet den Vorsitzenden des IRTF , die Vorsitzenden der verschiedenen Forschungsgruppen und möglicherweise einzelne weitere Personen der Forschungsgemeinschaft ("*members at large*").

Internet Research Task Force (IRTF)

Ziel der *Internet Research Task Force* (IRTF) ist die Förderung der Forschung im Hinblick auf das Wachstum und die Zukunft des Internets. Dazu werden kleine langfristige Forschungsgruppen gebildet, die auf gezielten Forschungsgebieten, wie Internet Protokolle, Applikationen, Architektur und Technologie arbeiten. Folgende Forschungsgruppen sind derzeit gebildet:

- End-to-End
- Information Infrastructure Architecture
- Privacy and Security
- Internet Resource Discovery
- Routing
- Services Management
- Reliable Multicast

Die IRTF wird im RFC 2014 vollständig beschrieben.

Internet Assigned Numbers Authority (IANA) [neu ICANN]

Die *Internet Assigned Numbers Authority* (IANA) ist die zentrale Koordinationsstelle

für die Zuweisung von eindeutigen Parameterwerten für Internet-Protokolle. Das heisst, sie ist für die Vergabe von IP-Adressen und Domain-Namen verantwortlich. In der Vergangenheit, wurden die Aktivitäten der IANA durch die Regierung der USA unterstützt. Dies ist nun nicht mehr möglich oder angemessen. Die USA hat nicht vor, in der Zukunft weiter (Geld-)Mittel bereitzustellen und so muss sich die IANA auf ihre Organisationen verlassen können. Das Internet ist von einem Forschungsnetz für Hochschulen zu einem multimedialen Netz für die Allgemeinheit geworden, das von Leuten über alle Welt benutzt wird. Diese Veränderung in Funktion und in der Anwenderbasis fordert nach einer neuen standfesten Institution mit einer breiten Gemeinschaftsvertretung aller Benutzer.
Dieses neue IANA ist zur Zeit gerade im Begriff zu entstehen. Das alte IANA stand noch unter der Leitung von Dr. Jon Postel vom University of Southern California's Information Sciences Institute. Das neue IANA heisst *Internet Corporation for Assigned Names and Numbers* (ICANN) und wird als Non-Profit-Organisation von einem internationalen Verwaltungsrat bestehend aus Repräsentanten einer Wählerschaft und weiteren internationalen Repräsentanten geführt. Das ICANN hat den Betrieb am 6. November 1998 offiziell aufgenommen.

Request for Comments Editor

Arbeitsdokumente im Internet heissen *Request For Comments* (RFC). Sie können Informationen, Diskussionsgrundlagen, aber auch Normen und Standards enthalten. Jeder kann RFCs schreiben. Um ein Dokument als RFC zu publizieren, muss es allerdings beim "RFC-Chefredaktor" eingereicht werden. Dieser "RFC-Chefredaktor" heisst im Internet *RFC-Editor* und ist unter http://www.rfc-editor.org erreichbar.
Der *RFC-Editor* ist Mitglied des IAB.
Mehr darüber im Kapitel über die *Request for Comments* (RFC),

InterNIC Registration Service

Die *InterNIC Registration Service* (NIC = Network Information Center) ist die ausführende Organisation, welche die Verwaltung der Internet-Adressen und Domainnamen im Auftrag der IANA organisiert.
Mehr darüber im Kapitel über das *Domain Name System* (DNS) des Internets.

World Wide Web Consortium (W3C)

Das *World Wide Web Consortium* (W3C) ist eine der jüngsten Organisationen des Internets. Es wurde im Oktober 1994 gegründet um das World Wide Web durch die Entwicklung allgemein gültiger Standards in seinem vollen Potential zu nutzen, seine Entwicklung zu fördern und seine Interoperabilität sicherzustellen. Das W3C ist ein internationales Industriekonsortium, gemeinsam vertreten im Internet durch folgende drei Organisationen:

- *Massachusetts Institute of Technology Laboratory for Computer Science* [MIT/LCS] in den USA
- *Institut National de Recherche en Informatique et en Automatique* [INRIA] in Europa
- *Keio University Shonan Fujisawa Campus* in Japan.

Dienste, welche das Konsortium anbietet:

- Informationen über das WWW und Standards für Entwickler und Benutzer
- Referenzcode und -Implementationen, um die Standards zu umzusetzen und zu befördern
- verschiedene Prototypen und Beispiel-Anwendungen, um den Gebrauch der neuen

Technologie zu demonstrieren
Das Konsortium wird geführt durch Tim Berners-Lee, Direktor und Erfinder des World Wide Web, und Jean-François Abramatic, als Chairman. W3C wird durch Mitgliedsorganisationen unterstützt und ist Verkäufer-Neutral. Es erarbeitet mit der Web-Gemeinschaft gemeinsam Produktespezifikationen und Referenzsoftware, welche weltweit frei zur Verfügung gestellt wird.

TCP/IP Standards: RFC

Arbeitsdokumente im Internet heissen *Request For Comments* (RFC). Ursprünglich waren RFCs genau das, was der Name bereits andeutet: Bitte um Kommentar. Die ersten RFCs waren Mitteilungen zwischen den Entwicklern des ARPANET, wo sie Problemlösungen diskutierten. Über die Jahre hinweg wurden die RFCs immer förmlicher. Bis zu dem Punkt, wo sie als Standard zitiert wurden, auch wenn es keine waren.
Deshalb werden nun die RFCs in zwei Untergruppen eingeteilt: *For Your Information* (FYI, Zu Ihrer Information) und *Standard* (STD, Standart). Die FYI-RFCs dienen als Dokument-Übersichten und für einleitende Themen. So gibt zum Beispiel RFC 1600 eine Übersicht über alle RFCs von 1 bis 1599 und dient damit als Inhaltsverzeichnis für RFCs < 1600. Häufig werden FYIs von der IETF Arbeitsgruppe *User Services Area* geschrieben. Die STD-RFCs identifizieren die RFCs, welche die wirklichen Internet Standards spezifizieren.
Jedem RFC (auch FYIs und STDs) ist eine Nummer zugeordnet, mit welcher es indiziert wird und über welche es gesucht werden kann. Wird ein RFC revidiert, bekommt es eine neue Nummer, die alte Nummer bleibt erhalten um die Referenzierung sicherzustellen. RFCs sind für jedermann zugänglich:

- http://ds.internic.net/rfc
- http://sunsite.cnlab-switch.ch

Die meisten RFCs sind Arbeitsdokumente, welche von der IETF verfasst worden sind. Diese werden publiziert, nachdem sie von der IESG genehmigt. Nicht nur spezielle Internet-Organisationen können RFCs schreiben, sondern jede Person. Um ein RFC publizieren zu lassen, müssen sie beim Chefredaktor für RFC-Dokumente, dem RFC-Editor (rfc-editor@rfc-editor.org), eingereicht werden. Dem Dokument wird dann eine offizielle Nummer zugewiesen. So hat zum Beispiel das RFC, welches den Standard für IP definiert, die Nummer 791. Bevor jedoch das erste RFC geschrieben wird, sollte man das RFC 2223 "Instructions to RFC-Authors" durchlesen.

TCP/IP Grundlagen

Die Hauptmerkmale der TCP/IP Architektur sind:

- Verbindungsloses (Connectionless) Protokoll auf der Vermittlungsebene (Netzschicht): IP
- Netzknoten als Paketvermittlungsrechner (Router)
- Transportprotokoll mit Sicherungsfunktionen: TCP
- Einheitlicher Satz von Anwendungsprogrammen: Telnet, FTP

TCP/IP im Vergleich mit zum ISO/OSI-Referenzmodell

Die TCP/IP-Architektur basiert auf vier Schichten gegenüber den 7 Schichten des ISO/OSI-Referenzmodells.

Dabei ist die unterste Schicht, der Network-Layer nicht weiter festgelegt. Dies ermöglicht aber die Unterstützung von einer ganzen Anzahl verschiedener Netzprotokolle (Ethernet, Tokenring, FDDI, X.25, FrameRelay, ATM, usw.). Dabei kann es aber vorkommen, das einzelne Funktionen eines Netzprotokolls (X.25 stellt z.B. die fehlergesicherte Übertragung von Daten über virtuelle Verbindungen sicher) durch TCP/IP-Protokolle (unnötigerweise) verdoppelt werden. Dadurch werden redundante Protokollinformationen und damit Overhead erzeugt. Dem gegenüber steht eine grösstmögliche Flexibilität bei der Gestaltung der Netze.

Die Kommunikation zwischen den Schichten (Transport, Internet, Network) erfolgt über bestimmte Primitives. Die Anwendungsschicht bildet mit den Anwendungen (FTP, Telnet, SMTP) die Schnittstelle zum Benutzer.

<table>
<tr><td></td><td>OSI-Layer</td><td></td><td></td><td></td><td>TCP/IP-Layer</td><td></td><td colspan="8">TCP/IP-Protocols</td></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>7</td><td>Application</td><td></td><td rowspan="5"></td><td></td><td rowspan="3">Application (Anwendung)</td><td></td><td rowspan="3">Telnet</td><td rowspan="3">FTP</td><td rowspan="3">SMTP</td><td rowspan="3">DNS</td><td rowspan="3">SNMP</td><td>NFS</td></tr>
<tr><td>6</td><td>Presentation</td><td></td><td></td><td></td><td>XDR</td></tr>
<tr><td>5</td><td>Session</td><td></td><td></td><td></td><td>RPC</td></tr>
<tr><td>4</td><td>Transport</td><td></td><td></td><td>Transport (Rechner)</td><td></td><td colspan="3">TCP</td><td colspan="4">UDP</td></tr>
<tr><td>3</td><td>Network</td><td></td><td rowspan="4">Router</td><td></td><td rowspan="2">Internet</td><td></td><td colspan="8">RIP, OSPF, EGP</td></tr>
<tr><td rowspan="2">2</td><td rowspan="2">Data Link</td><td rowspan="2"></td><td rowspan="2"></td><td colspan="8">IP, ICMP, ARP, RARP</td></tr>
<tr><td></td><td></td><td>Network (Netzzugang)</td><td></td><td>Ethernet</td><td>Tokenring</td><td>FDDI</td><td>X.25</td><td>FrameRelay</td><td>ATM</td><td>SLIP</td><td>PPP</td></tr>
<tr><td>1</td><td>Physical</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
</table>

TCP/IP-Architektur im Vergleich zum ISO/OSI Referenzmodell

Telnet	Virtual Terminal Protocol
FTP	File Transfer Protocol
SMTP	Simple Mail Transfer Protocol
DNS	Domain Name Server
SNMP	Simple Network Management Protocol
NFS	Network File System
XDR	Exchange Date Representative Protocol
RPC	Remote Procedure Call
TCP	Transmission Control Protocol
UDP	User Datagram Protocol
IP	Internet Protocol
ICMP	Internet Control Message Protocol
ARP	Address Resolution Protocol
RARP	Reverse Address Resolution Protocol
RIP	Routing Information Protocol
OSPF	Open Shortest Path First Protocol
EGP	Exterior Gateway Protocol
Ethernet	802.3
Tokenring	802.5
FDDI	Fiber Distributed Data Interface
X.25	
FrameRelay	
ATM	Asynchronous Transfer Mode
SLIP	Serial Line over IP
PPP	Point to Point Protocol

Die wichtigsten Protokolle im TCP/IP-Protokollstapel

Die folgenden Protokolle bilden den Kern des TCP/IP-Protokollstapels.

Internet Protocol (IP)

Beschreibt eine Datagramm-Prozedur (verbindungslos) zur Kommunikation über ein oder mehrere Netze. Der Standard geht von nicht fehlergesicherten Protokollen auf der Netzwerkschicht aus.

Transmission Control Protocol (TCP)

Ein verbindungsorientiertes End-zu-End-Protokoll zur fehlergesicherten Übertragung von Daten.

User Datagram Protocol (UDP)

UDP stellt eine schnelle, verbindungslose Transportverbindung zwischen den Endsystemen zur Verfügung.

File Transfer Protocol (FTP)

Einfaches Protokoll zum Austausch von ASCII- und Binärdateien.

Simple Mail Transfer Protocol (SMTP)

E-Mail Übertragungsprotokoll. Ermöglicht das Senden und Empfangen von E-Mail.

Telnet Protocol (Telnet)

Beschreibt ein *Network Virtual Terminal* (NVT) als Grundlage für die Einbindung von Endgeräten (*Remote Terminal Access*) in ein Netzwerk. Emulationen zum Beispiel für VT100/VT220 oder TN3270 (IBM 3270).

Der Network Layer
Die TCP/IP-Architektur beschreibt kein eigenes Protokoll im Network Layer. Es unterstützt aber als ein von der darunterliegenden Schicht unabhängiges Protokoll eine Vielzahl von Netzwerktechnologien. Es gibt eine ganze Reihe von RFCs, welche Empfehlungen geben, wie IP-Pakete über Netzwerkprotokolle transportiert werden sollen. Abbildung 8 gibt einen Überblick über diese netzwerkspezifischen Standards.

IP über serielle Leitungen
Eine einfache und kostengünstige Verbindung zwischen Rechner und IP-Netzen oder zwischen zwei IP-Netzen kann über eine Punkt-Punkt-Verbindung realisiert werden. Dazu wird eine asynchrone RS-232/V.24-Schnittstelle für die physikalische Verbindung vorausgesetzt. Die IP-Pakete werden mit dem *Serial Line Internet Protocol* (SLIP) oder dem *Point-to-Point Protocol* (PPP) übertragen.

Serial Line Internet Protocol (SLIP)
Serial Line IP (SLIP) ist ein de facto Standard, welcher TCP/IP Punkt-Punkt-Verbindungen über serielle Datenleitungen beschreibt. Es entstand in den frühen 1980'er Jahren und wurde 1988 im RFC 1055 beschrieben.
SLIP definiert eine Zeichenfolge, welche ein IP-Paket auf einer seriellen Leitung einrahmen. Das ist auch schon alles. Es bietet keine Adressierung, keine unterschiedlichen Pakettypen, keine Fehlererkennung und -korrektur und keine Kompressionsmechanismen. Es handelt sich um ein sehr einfaches Protokoll, welches auch sehr einfach zu implementieren ist. Der RFC selbst enthält den C-Source für eine einfache Implementierung als Beispiel.
SLIP wird vor allem auf dedizierten seriellen Verbindungen und manchmal auch für Dial-Up Zwecke verwendet. Dabei verwendet es Durchsatzraten von 1200bps bis 19.2kbps.

SLIP ist fast gänzlich vom universellen *Point-to-Point Protocol* (PPP) verdrängt worden.

Point-to-Point Protocol (PPP)
SLIP dient dazu, Daten zwischen TCP/IP-Rechnern über serielle WAN-Strecken zu transportieren. Da für andere Protokolle (DECnet, AppleTalk, IPX, usw.) keine allgemein gültigen Standards zur Verfügung standen, mussten diese Informationen über eigene Mechanismen auf der Schicht 2 übertragen werden. Als Ausweg aus diesem Dilemma zeichnet sich nun durch die Verfügbarkeit des *Point-to-Point Protocols* (PPP) eine Lösung ab.

PPP ermöglicht die Übermittlung von Daten über synchrone (Bit-seriell) und asynchrone (Start/Stop-Betrieb) Wahl- und Standleitungen. PPP ist dadurch in der Lage, unaghängig vom jeweiligen physischen Interface (RS-232-C, RS-422, RS-423, X.21, V.24, V.35) zu arbeiten. Die einzige Voraussetzung, die gefordert wird, besteht in einer vollkommen transparenten, voll-duplex -fähigen Datenleitung. Als Datenformat sind bei PPP 8 Bit und keine Parität festgelegt. Ausserdem wird ein Flusskontroll-Mechanismus über die Verbindung unterstützt.
PPP basiert auf folgenden drei Hauptkomponenten:

- Data Encapsulation
- *Link Control Protocol* (LCP)
- eine Familie von *Network Control Protocols* (NCP)

Data Encapsulation

Das bekannte *High Level Data Link Control Protocol* (HDLC) wurde bei PPP als Basis zum transportieren von Datenrahmen auf der Schicht zwei spezifiziert. HDLC ist seit Mitte der siebziger Jahre weltweit standardisiert und wurde im ISO-Standard ISO3309 veröffentlicht.

Das PPP-Datenformat:

Flag	Address	Control	Protocol	Data	FCS	Flag

Flag Sequence Jeder PPP-Datenrahmen wird durch ein Flag mit dem binären Wert 01111110 eröffnet und beendet.

Address PPP unterstützt momentan noch keinen Adressierungsmechanismus, der das Adressieren von individuellen Stationen ermöglicht. Daher wird ausschliesslich mit der All-Station (Broadcast)-Adresse mit dem binären Wert 11111111 gesendet.

Control Das Control-Feld definiert immer das *Unnumbered-Information* (UI)-Kommando (binär 00000011), bei dem das Poll/Final-Bit auf den Wert 0 gesetzt ist. Datenrahmen mit anderen Werten werden verworfen.

Protocol Das zwei Byte lange Protokoll-Feld definiert, nach welchen Regeln die nachfolgenden Daten zu behandeln sind. Die Werte des Protokollfeldes werden in den Assigned Numbers in den jeweiligen RFCs publiziert.

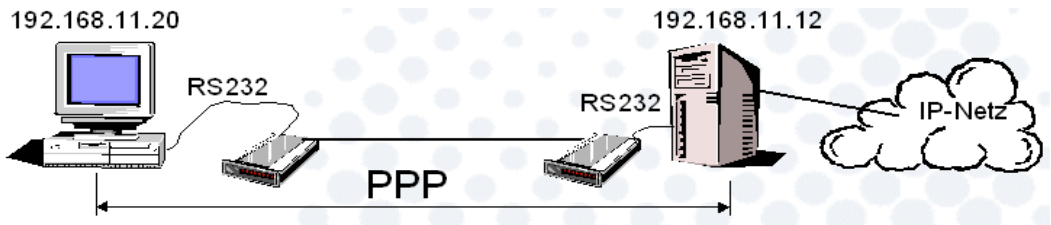
Data Das Data-Feld enthält protokollspezifische Informationen (Header und Daten) des im Protokollfeld definierten Network-Layer Protokolls. Die Defaultlänge beträgt 1500 Byte. Zwischen den Kommunikationspartnern kann die maximale Rahmenlänge jederzeit ausgehandelt werden.

FCS Die 16 Bit lange *Frame Check Sequence* (FCS) ermöglicht die Fehlerkontrolle über die übermittelten Datenpakete.

Link Control Protocol (LCP)

Das *Link Control Protocol* (LCP) sorgt für den ordnungsgemässen Aufbau, die Konfiguration, den Test und den Aufbau einer PPP-Verbindung. Bevor die eigentlichen Datenrahmen über eine PPP-Verbindung übermittelt werden, sendet jedes der beteiligten PPP-Interfaces eine Reihe von LCP-Datenrahmen auf die Leitung. Das *Link Control Protocol* durchläuft dabei die folgenden fünf Phasen:

1		Link Dead (verbindungsloser Zustand)
2		Link Establishment (Verbindungsaufbau-Phase)
3		Authentification (optionaler Modus für die Authentifizierung)
4		Network Layer Protocol Configuration (Konfiguration der Verbindung)
5		Link Termination (Verbindungsabbau)



Point-to-Point Protocol (PPP) / Dial-Up zum ISP

Der Internet Layer

Im Network Layer, wie auch im Application Layer der TCP/IP-Architektur sind eine grosse Anzahl von Protokollen vorgesehen. Der Internet Layer verbindet diese beiden Schichten. Dabei kommen die folgenden Protokolle zum Einsatz:

- *Internet Protocol* (IP)
- *Internet Control Message Protocol* (ICMP)
- *Routing Information Protocol* (RIP)
- *Open Shortest Path First Protocol* (OSPF)
- *Address Resolution Protocol* (ARP)
- *Reverse Address Resolution Protocol* (RARP)

Dabei ist das *Internet Protocol* (IP) der grosse Dreh- und Angelpunkt.

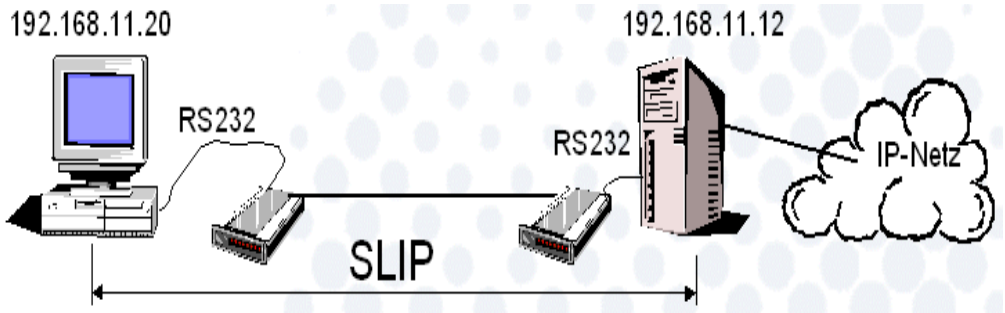
Das Internet Protokoll (IP)

Das Internet Protocol (IP) bildet zusammen mit dem Transmission Control Protocol (TCP) das zentrale Protokollpaar der Internetarchitketur. Die Hauptaufgabe des Internet Protokolls ist das Adressieren von Rechnern sowie das Fragmentieren von Paketen der darüberliegenden Schicht. IP stellt also die Endsystemverbindung zwischen Partnerrechnern her. Der darüberliegenden Ebene bietet IP einen sog. unzuverlässigen und verbindungslosen Dienst an. Wenn also eine zuverlässige Übertragung gefordert wird (z.B. Dateitransfer), dann ist es Aufgabe eines der übergeordneten Protokolle, die Zuverlässigkeit zu gewährleisten.

IP-Pakete sind das Fundament der TCP/IP Protokollfamilie. Jedes Paket ist zusammengesetzt aus einem Header und einem Rumpf. Der Header enthält Quell- und Zieladresse und der Rumpf enthält die eigentlichen Nutzdaten. Ein typisches IP-Paket umfasst einige 100 Byte.

IP sorgt für die Aufteilung der Daten in kleine Päckchen, und für den Verbindungsaufbau zum Empfänger. Darauf setzen das Internet Control Message Protokoll (ICMP), sowie das Address Resolution Protocol (ARP) auf.

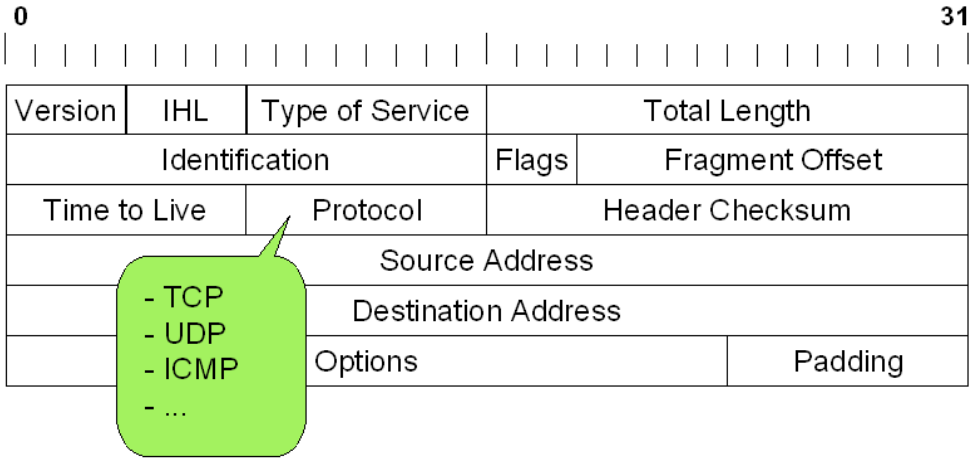
IP ist ein ungesicherter Dienst. Pakete können verlorengehen, mehrfach zugestellt werden, einander überholen. Auch die Integrität der Nutzdaten wird nicht überprüft. Selbst die Korrektheit der Quelladresse ist nicht garantiert. Eine Authentifikation



muss auf höheren Schichten stattfinden.

IP-Header Format

Ein IP-Datagramm besteht aus einem Kopf- und einem Textteil. Der Kopfteil hat einen festen Teil mit 20 Bytes und einen optionalen Teil von variabler Länge. Das Format des Nachrichtenkopfes wird in Abbildung 13 gezeigt.



IP-Header Format

Bedeutung der einzelnen Felder:

Version Das Versionfeld gibt die verwendete Version des IP-Headers an. Momentan wird mit der IP-Version 4 gearbeitet.

Internet Header Length

Die Internet Header Length gibt die gesamte Länge des IP-Headers in 32-Bit-Einheiten an. Die IHL ist notwendig, da das Optionsfeld eine variable Länge aufweist.

Type of Service

Das Type of Service Feld definiert die geforderten Dienste eines IP-Datagramms. Dieses Feld ermöglicht, den Rechnern bei der Übertragung über weite Netzwerke die gewünschte Dienstart (vorrangige Behandlung, sowie optimierte Übertragung bezüglich Durchsatz, Verzögerung oder Zuverlässigkeit) anzugeben.

Leider wird der Type of Service in den meisten kommerziell erhältlichen Produkten nicht unterstützt oder ist nicht vollständig implementiert.

Total Length

Die Total Length gibt die Gesamtlänge des Datagramms, einschliesslich des IP-Headers und des Datenteils der höheren Protokolle in Anzahl der darin enthaltenen Bytes an. Rechner müssen in der Lage sein, Datenpakete mit einer Länge von 576 Bytes zu empfangen. Grössere Datagramme werden durch den Transport-Layer vor dem Senden fragmentiert.

Identification

Kennwert zur Zuordnung von Fragmenten zu einem Datagramm.

Flag Dieses Feld enthält die Informationen "Don't fragment", falls die Fragmentierung nicht unterstützt wird und "More fragments", zur Anzeige, dass noch weitere zum Datagramm des oberen

Layers gehörende Fragmente folgen. "Last Fragment" gibt an, das mit dem Datagramm das letzte Fragment des Datagramms des oberen Layers transportiert wird.

Fragment

Offset Dieses Feld gibt die Lage der Fragmentdaten relativ zum Anfang des Datenblocks im ursprünglichen Datagramm an.

Time-to-Live

Das Time-to-Live-Feld definiert die verbleibende Lebensdauer eines Datagramms im Netz. Fällt der Wert auf Null, so muss das Datagramm verworfen werden. Dieser Wert wird durch jeden

durchlaufenen Router um mindestens eine Einheit herabgesetzt. RFC1700 spezifiziert einen Defaultwert von 64.

Protocol

Im Protokollfeld wird definiert, welches höhere Protokoll dem Datenteil vorangestellt ist [RFC1700]:

1 = ICMP Internet Control Message Protocol
2 = IGMP Internet Group Management Protocol
6 = TCP Transmission Control Protocol
8 = EGP Exterior Gateway Protocol
17 = UDP User Datagram Protocol
29 = ISO TP4 ISO Transport Class 4 Protocol
88 = IGRP Interior Gateway Routing Protocol (Cisco)
89 = OSPFIGP Open Shortest Path First Interior Gateway Protocol

Header

Checksum Enthält eine Prüfsumme, die nur den IP-Header auf Fehler überprüft.

Source

Address Enthält die IP-Adresse des Rechners, der das Datagramm erzeugt und abgesendet hat.

Destination

Address Enthält die IP-Adresse des Rechners, für den das Datagramm bestimmt ist.

Options

Die Dienste des Internet-Protokolls können durch Optionen an die speziellen Anforderungen der höheren Protokolle angepasst werden. Die Feldlänge hängt von der Art und der Anzahl der Optionen ab, die mit einem

Datagramm übertragen werden. Für die Optionen wurden folgende zwei Formate definiert [RFC1700]:

1. Ein 1-Byte-Optionentyp
2. Ein String variabler Länge:

enthält ein Optionen-Typ-Byte, ein Optionen-Längen-Byte und
die Optionsdaten.

Leider werden die IP-Optionen in den meisten heute kommerziell erhältlichen Produkten nicht oder nur unvollständig unterstützt.

Padding

Dies ist ein "Lückenfüller". Das Padding -Feld stellt durch Auffüllen des Headers mit Nullen sicher, dass der Header vollständig aus 32bit-Worten besteht.

Adressierung

Jeder Rechner, der am Internet angeschlossen wird, braucht eine eindeutige Kennung. Diesen Zweck erfüllt die Internet-Adresse (IP-Adresse). Die IP-Adresse ist eine 32-bit breite Zahl. Der besseren Lesbarkeit halber fasst man jeweils ein Oktett zu einer Dezimalzahl zusammen:
11000000 10101000 00010110 00101111 wird zu 192.168.22.47
Die 32 Bit werden in Worte zu jeweils 8 Bit zusammengefasst, denen dann die entsprechende Dezimalzahl zugeordnet wird. Diese sog. Oktetts sind durch Punkte voneinander getrennt (dotted quad notation): 192.168.22.47
Der Wertebereich für jedes Oktett liegt zwischen 0 und 255. Wobei der Wert 0 für die Adressierung eines Netzwerks und der Wert 255 für Broadcasts reserviert ist.
Jede IP-Adresse besteht aus zwei Teilen: der Netzwerkadresse und der Systemadresse (Host-ID).

Netz-Klassifizierung

Die IP-Adressen werden in fünf Klassen eingeteilt. Jede Klasse ist dabei für einen bestimmten Verwendungszweck bzw. für bestimmte Netzwerkgrössen vorgesehen:

Class A

Netzwerkadressen: 1.0.0.0 - 126.0.0.0
Anzahl Netze: 126
Anzahl Hosts pro Netzwerk: max. 16'777'214 Hosts pro Netz

Class B

Netzwerkadressen: 128.1.0.0 - 191.254.0.0
Anzahl Netze: 16'382
Anzahl Hosts pro Netzwerk: max. 65'534 Hosts pro Netz

Class C

Netzwerkadressen: 192.0.1.0 - 223.255.254.0
Anzahl Netze: 2'097'150
Anzahl Hosts pro Netzwerk: max. 254 Hosts pro Netz

Class D

Netzwerkadressen: 224.0.0.1 - 239.255.255.254

Class E

Netzwerkadressen: 240.0.0.1 - 254.255.255.254

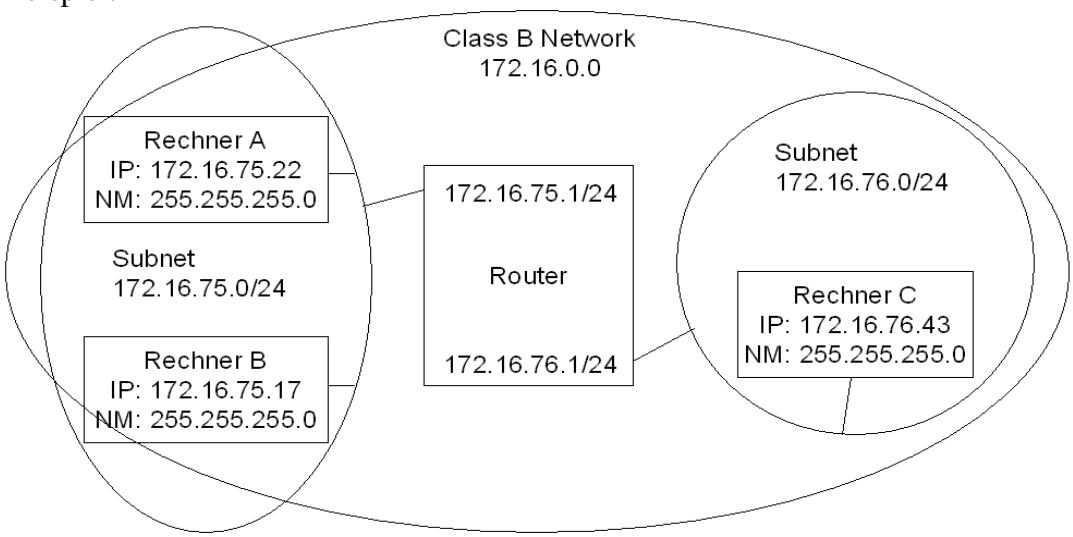
Loopback

Adressen beginnend mit 127 (127.0.0.0 - 127.255.255.255) werden nur Maschinenintern für Loopback verwendet.

Subnetz-Adressen

Mit der Subnettierung wird erreicht, dass der Host-Id-Bereich weiter unterteilt werden kann. Mittels einer Subnet-Mask werden einige Bits von der Host-Id abgetrennt. Diese Bits werden dann zur Adressierung der einzelnen Subnetze benutzt.
Eine Class B Adresse könnte zum Beispiel wie folgt aufgeteilt werden:

Die dazu verwendete Subnet-Mask lautet: 255.255.255.0
Beispiel:



Class B Network mit Subnetz

Rechner A will ein IP-Paket an Rechner C senden.
Um zu wissen, ob Rechner C im gleichen Subnet ist (und er das Paket gleich direkt senden kann) oder ob Rechner C in einem anderen Subnet ist (und er das Paket an den Router senden muss) vergleicht er seine Subnet-Adresse mit der Subnet-Adresse von Rechner C.
Seine eigene Subnet-Adresse erhält er durch eine logische UND-Verknüpfung seiner IP-Adresse mit seiner Netmask:

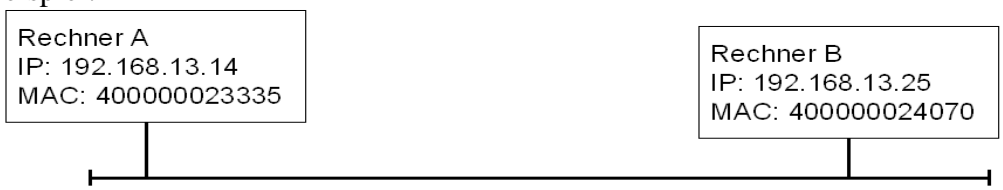
```
          10101100 00010000 01001011 00010110 172. 16. 75. 22
UND       11111111 11111111 11111111 00000000 255.255.255. 0
        = 10101100 00010000 01001011 00000000 172. 16. 75. 00
```

Nun prüft er, ob der Empfänger-Rechner im gleichen Subnet ist. Das Vorgehen ist gleich wie vorher:

```
          10101100 00010000 01001100 00101011 172. 16. 76. 43
UND       11111111 11111111 11111111 00000000 255.255.255. 0
        = 10101100 00010000 01001100 00000000 172. 16. 76. 00
```

Ist 172.16.75.00 gleich 172.16.76.00?
Nein. Also sendet Rechner A sein IP-Paket an den Router. Der Router prüft nun wiederum anhand der Netmask nach obigem Vorgehen, an welches Subnet er das Paket senden muss.

Address Resolution Protocol (ARP)
Das *Address Resolution Protocol* (ARP) sorgt für die Umsetzung der logischen IP Adressen in Adressen der darunterliegenden Schicht (Network Layer).
ARP arbeitet wie folgt:
Vor der Übertragung von IP-Paketen überprüft das Internet Protokoll (IP) das Vorhandensein eines Eintrages für die Zieladresse in der ARP-Adresstabelle (ARP Cache), bei Ethernet auch Internet-to-Ethernet-Translation-Table genannt. ARP vergleicht die vorhandenen Adresstabellen mit der Anfrage des Internet-Protokolls. Wird kein Eintrag im ARP-Cache gefunden, so wird bei allen Rechnern am Netz die gewünschte Adresse mit Hilfe einer IP-Broadcast-Meldung erfragt. Nur der Rechner mit einem Eintrag zu dieser IP-Adresse antwortet auf diese Anfrage. Die Antwort (ein ARP-Reply) auf den ARP-Request wird im ARP-Cache gespeichert. Dieser Eintrag verbleibt für zwanzig Minuten in der Tabelle, bis er nach Ablauf des ARP-Timers automatisch wieder gelöscht wird.
Beispiel:



- Rechner A möchte mit Rechner B Daten austauschen.
- Er kennt aber nur die IP-Adresse von Rechner B.
- Rechner A sendet per IP-Broadcast einen ARP-Request auf die Reise: „Derjenige mit der IP-Adresse 192.168.13.25 melde mir bitte seine MAC-Adresse!“
- Rechner B erkennt den Request und sendet Rechner A seine IP-Adresse.

Abbildung 15: Beispiel mit ARP

Sicherheitsproblematik
Dieses Verfahren kann missbraucht werden. ARP ist nur solange sicher, als dass nur vertrauenswürdige Systeme am Netzstrang angeschlossen sind. Es ist möglich, dass ein Angreifer ARP-Pakete fälscht und so eine Umleitung der Daten auf sein System erreichen kann. ARP kann bei erhöhten Sicherheitsanforderungen auch abgeschaltet werden, dann müssen die jeweiligen Tabellen jedoch fixiert werden und dementsprechend auch geschützt werden.

Reverse Address Resolution Protocol (RARP)
Wie erfährt ein Diskless-Rechner seine eigene IP-Adresse?
Beim Einsatz von "Diskless" Workstations in einem TCP/IP-Netzwerk besteht das Problem, dass diesen Rechnern keine IP-Adresse direkt zugeordnet werden kann. Da die Hersteller von Diskless-Workstations keine individuelle IP-Adresse in den Initialisierungscode einfügen können, muss ein Mechanismus vorgesehen werden, der es ermöglicht, mit einem File-Server Kontakt aufzunehmen. Von diesem File-Server wird die spezifische IP-Adresse für die jeweilige Diskless Workstation dynamisch abgefragt. Die Zuordnung von IP-Adressen zu Diskless Workstation

erfolgt in der TCP/IP-Welt mit folgenden drei höchst unterschiedlichen Protokollen:

- *Reverse Address Resolution Protocol* (RARP)
- *Bootstrap Protocol* (BOOTP)
- *Dynamic Host Configuration Protocol* (DHCP)

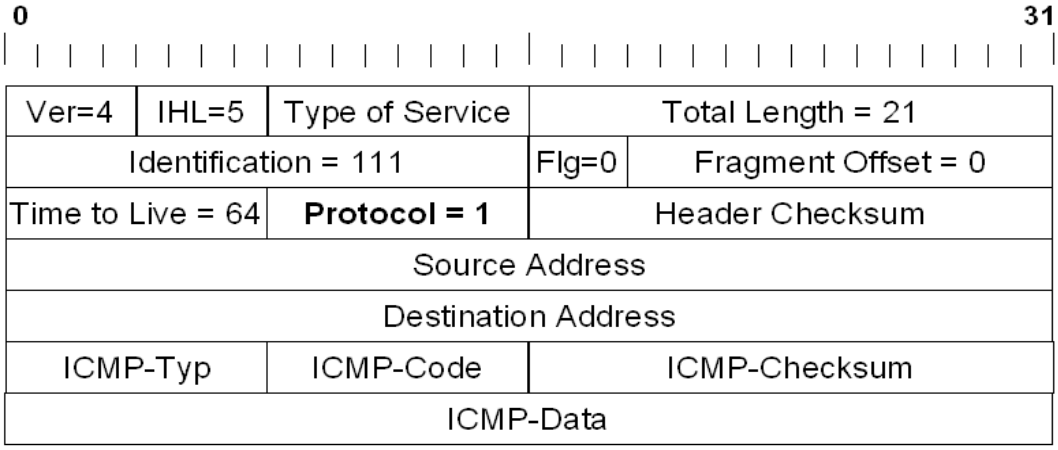
Das *Reverse Address Resolution Protocol* (RARP) ist ein sehr einfaches Protokoll. RARP wurde vom *Address Resolution Protocol* (ARP) abgeleitet.
RARP arbeitet wie folgt:
Zur Abfrage seiner eigenen IP-Adresse sendet der Sender einen RARP-Request als IP-Broadcast an alle Rechner. In diesem Broadcast-Paket trägt der Sender seine eigene Hardware-Adresse im Source-Adressfeld ein. Dieser Broadcast-Request wird von allen am Netz aktiven Rechnern empfangen. Beantwortet werden diese Requests aber nur von Rechnern, die den RARP-Service aktiviert haben und über einen entsprechenden Eintrag in der Konfigurationsdatei (auf Unix-System in der Regel in /etc/ethers) verfügen. Findet der RARP-Server einen entsprechenden Eintrag, so wird der RARP-Request durch einen RARP-Reply an die Hardware-Adresse des Requesters beantwortet. Im RARP-Reply-Header wird die IP-Adresse der suchenden Maschine mitgegeben. Befinden sich mehrere RARP-Server im Netz, welche eine Antwort senden, so reagiert der RARP-Requester nur auf die erste eintreffende Meldung.
Beispiel:

- Rechner B ist ein Gerät ohne Disk (Diskless Workstation) und kennt nach dem Aufstarten seine eigene IP-Adresse nicht. Die MAC-Adresse ist fest „eingebrannt".
- Nach dem Aufstarten sendet Rechner B einen RARP-Request per IP-Broadcast an alle Rechner: „Ich bin 400000024070. Wer kann mir meine IP-Adresse mitteilen?"
- Rechner A ist als RARP-Server dazu in der Lage und meldet in einem RARP-Reply an Rechner B die IP-Adresse 192.168.13.25 zurück.

Beispiel mit RARP

Internet Control Message Protocol (ICMP)

Das *Internet Control Message Protocol* (ICMP) ist ein verbindungsloser Dienst, der nur zum Senden von Kontrollmeldungen auf dem Internet Layer dient, wird zum Beispiel ein Adressat nicht erreicht, so sorgt dieses Protokoll für die Vernichtung der überflüssigen Daten.
Mit ICMP lässt sich das Verhalten von TCP Verbindungen beeinflussen. Es dient dazu, Hosts günstigere Routen zu einem Ziel bekanntzugeben, über Routing-Probleme zu informieren oder Verbindungen wegen Problemen im Datennetz abzubrechen. Auf ICMP basieren die Kommandos *ping* und *traceroute*.



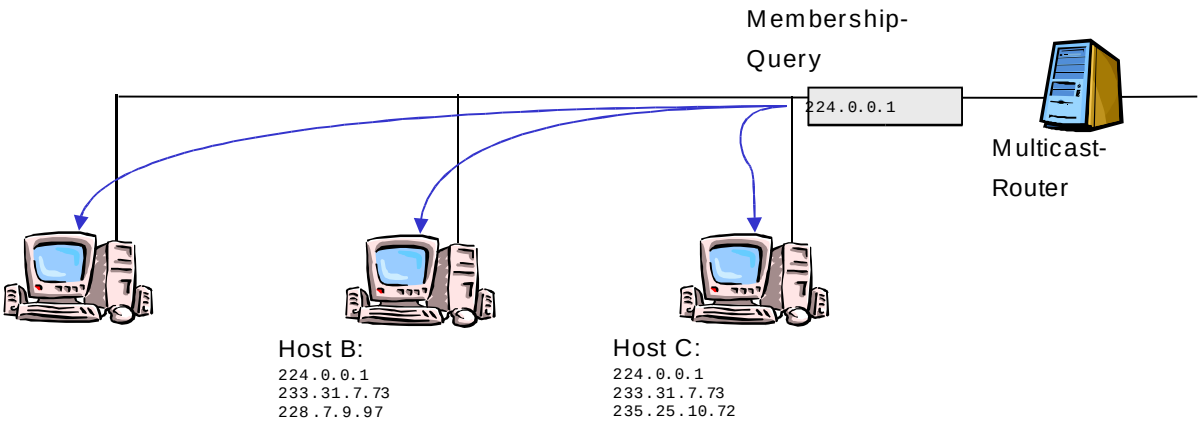
Struktur und Beispiel eines ICMP-Internet-Datagramms

Der ICMP-Header besteht aus einem ICMP-Typen-feld, ICMP-Code-Feld und einem ICMP-Checksummen-Feld. Anschliessend an den Header folgt das ICMP-Daten-Feld.
Die wichtigsten ICMP-Pakettypen:

Typ	Funktion
0	Echo Reply
3	Destination Unreachable
8	Echo Request
11	Time Exceeded for a Datagram
30	Traceroute

Sicherheitsproblematik
Mit ICMP-Nachrichten können z.T. alle Verbindungen zwischen zwei Systemen unterbrochen werden, wenn nur innerhalb einer Verbindung "Destination unreachable" gemeldet wird (bezieht sich auf alte Implementationen von ICMP).
Mit Redirect-Nachrichten kann Verkehr zu bestimmten Zielen umgeleitet werden. Nur Hosts sollten Redirect Nachrichten Glauben schenken, niemals Router. Router sollten fest an Ihre Routingtabellen glauben und nicht durch Angreifer aus dem Tritt gebracht werden.

Nachricht an alle:
Multicast auf dem Network-Layer



Routing
Um Daten von einem Subnet in ein anderes übertragen zu können, benötigt man einen Router. Dies ist im einfachsten Fall ein Rechner mit zwei oder mehr Netzwerksinterfaces.
Im Gegensatz zu einem Repeater oder zu einer Bridge speichert ein Router das ganze Datenpaket zwischen, bevor er es auf dem anderen Subnetz wieder ausgibt.
Ebenso kann er, wenn sich die Paketlänge auf den beiden Subnetzen unterscheidet, grosse Datenpakete in kleinere Päckchen fragmentieren bzw. Die Fragmente wieder zusammensetzen.
Will ein Rechner an einen anderen Rechner, welcher sich nicht im gleichen Subnetz befindet, Daten schicken, kann er diese nicht direkt absenden, sondern muss sie zu einem Router senden. Befindet sich der Zielrechner an einem an diesem Router angeschlossenem Subnetz, so kann die Daten der Router dort abliefern. Ansonsten muss er die Daten zum nächsten Router weiterreichen. Wohin die Daten weitergereicht werden wird anhand von Routing-Tabellen bestimmt.

Routing-Methoden
Grundsätzlich können Routing-Methoden in folgende Kategorien eingeteilt werden:
Statisches Routing Die benötigten Daten für die Verkehrslenkung (Routing) der Datagramme werden manuell oder durch externe Rechner vorbereitet und dann nach Eingabe fest im Router aktiviert. Nach Bedarf wird der Knoten auf dieselbe Art neu konfiguriert, wobei auch einzelne Daten im Betrieb eines Routers geändert werden können.

Bei UNIX-Rechnern kann die statische Routing-Tabelle mit dem Kommando /usr/etc/netstat -r abgefragt werden.
C:\>netstat -r
Routing-Tabelle
Aktive Routen:
Netzwerkadresse Subnet Mask Gateway-Adresse Schnittstelle Anzahl
0.0.0.0 0.0.0.0 147.78.78.129 147.78.78.170 1
127.0.0.0 255.0.0.0 127.0.0.1 127.0.0.1 1
147.78.78.128 255.255.255.192 147.78.78.170 147.78.78.170 1
147.78.78.170 255.255.255.255 127.0.0.1 127.0.0.1 1
147.78.255.255 255.255.255.255 147.78.78.170 147.78.78.170 1
224.0.0.0 224.0.0.0 147.78.78.170 147.78.78.170 1
255.255.255.255 255.255.255.255 147.78.78.170 147.78.78.170 1

Adaptiv, dynamisch Die Netzkonfiguration und der Zustand des Netzes wird in kürzeren Abständen automatisch erfasst und berücksichtigt. Je nach Art der Erfassung unterscheidet man:

Zentralisiertes Routing: Erfassung und Verteilung der Information über das Netz mit einem zentralen Rechner, dem sogenannten *Routing Control Center* (RCC).Jeder Knoten meldet dem RCC periodisch die für das Routing wichtige Information, z.B. über ausgefallene Leitungen. Das RCC ermittelt aus diesen Daten für jeden Knoten die auszuwählenden Wege zu jedem anderen Knoten und verteilt die Daten in der Form von Routingtabellen an alle Knoten. Beim zentralisierten Routing wird im allgemeinen der Zustand des ganzen Netzes erfasst und bei der Festlegung der Wege berücksichtigt.
Vorteile:
Ganzer Netzzustand kann berücksichtigt werden.
Die Zustandsangaben sind umfangreich, sie müssen nur an ein Zentrum übermittelt werden und nicht an alle anderen Knoten.
Nachteile:
Die Berechnung erfordert viel Rechenkapazität, die wohl nur im Zentrum bereitgestellt werden

kann.
Wenn die Verkehrssituation berücksichtigt werden soll, muss der Zyklus von Datenerfassung, Auswertung und Übermittlung neuer Routingtabellen in Sekundenschnelle ablaufen.
Daraus resultiert aber ein häufiges Übermitteln von Daten, was eine grosse Bandbreite erfordert.
Ausfälle von Leitungen auf dem Weg zum RCC können sich gravierend auswirken.
Der Ausfall des RCC führt zum Ausfall des ganzen Netzes.
Die Routingtabellen werden nicht genau gleichzeitig an alle Knoten übermittelt. Daher können zeitweise in verschiedenen Knoten inkonsistente Tabellen wirksam sein, was zu Umwegen der Pakete führen kann.
Für jedes Ziel muss mehr als ein Leitweg bekannt sein, da andernfalls durch einen einfachen Ausfall die Kommunikationsmöglichkeit zwischen Hosts und RCC unterbrochen werden kann.

Verteiltes Routing: Die Knoten (Router) tauschen untereinander eigens für das Routing verwendete Informationen in besonderen Paketen aus (mittels Routing Protokollen).
Beispiele:
Routing Information Protocol (RIP) [RFC1058]
Interior Gateway Routing Protocol (IGRP) [Cisco]

Hierarchisch Bei grossen Netzen ist es zu aufwendig, für jeden Knoten des Netzes einen Eintrag in den Routing-Tabellen zu haben: die Tabellen werden umfangreich, und Änderungen der Konfiguration des Netzes ziehen Änderungen in den Routing-Tabellen der Knoten nach sich.
Man bildet Regionen, die für das Routing von aussen gesehen eine Einheit bilden, d.h. die Knoten ausserhalb der Region senden jedes für die Region bestimmte Paket an einen bestimmten Knoten der Region. Dieser sorgt dann für die weitere Verteilung innerhalb der Region.
Damit die Zugehörigkeit eines Knotens zu einer bestimmten Region einfach bestimmt werden kann, muss die Region auf einfache Art aus der Adresse des Knotens ermittelt werden können.
Beispiele:
Open Shortest Path First (OSPF) [RFC1247]
Border Gateway Protocol (BGP) [RFC1265 bis 1268]

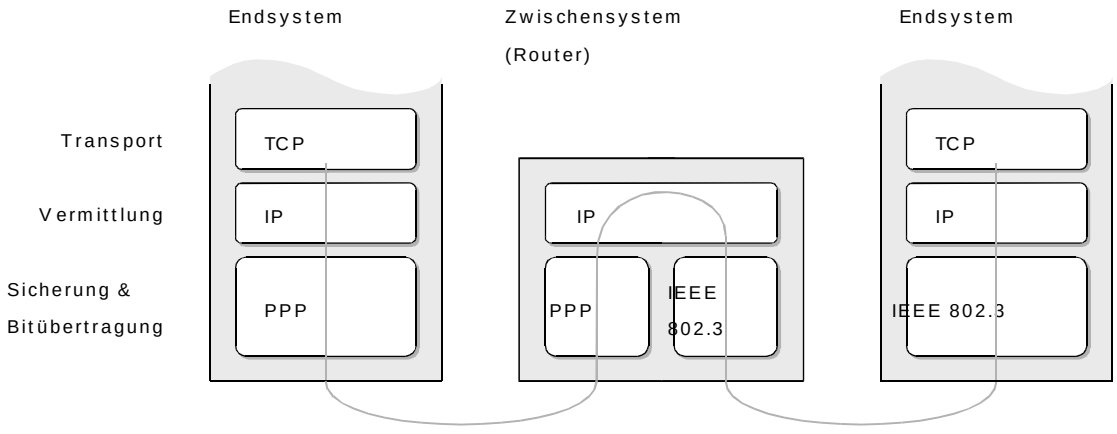


Abb: Schema eines Internetrouters

Der Transport Layer

Die Aufgaben der Transportschicht werden in der TCP/IP-Architektur durch die beiden Protokolle TCP und UDP wahrgenommen.

Transmission Control Protocol (TCP)

Das *Transmission Control Protocol* (TCP) ist für die Ende zu Ende Kontrolle der Nutzdaten zuständig. Analog zur Post kann man sagen, dass die Datenpakete erst in einen IP-Umschlag kommen, der die Adresse enthält, und dann in einen weiteren Umschlag, der Daten über den Inhalt enthält. Das *User Datagram Protocol* (UDP) kann man in diesem Zusammenhang als einfache Briefsendung betrachten, bei der man nicht erfährt, ob der Brief angekommen ist, TCP entspräche dann einem Einschreiben mit Rückschein.

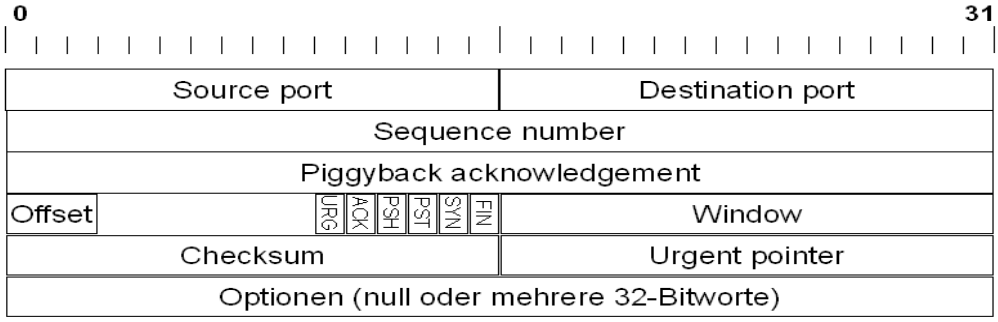
Im Einzelnen leistet TCP folgendes:

· Flusskontrolle und Zeitüberwachung einer Verbindung
· Überwachung der richtigen Reihenfolge der Pakete
· Überwachung der Zuverlässigkeit und Sicherheit einer Verbindung
· Steuerung der Prioritäten der Vorrangdaten

Dieser Verwaltungsaufwand macht sich in einem deutlich grösseren Control-Overhead bemerkbar. TCP braucht dafür 20 Byte, UDP im Vergleich dazu lediglich 8.

Das TCP stellt über dem IP die gesicherte, virtuelle Verbindung her. IP-Pakete werden wieder in der richtigen Reihenfolge zusammengesetzt. Verlorene oder unvollständige Pakete werden nochmals übermittelt. TCP dient als Basis für Anwendungen wie Telnet oder ftp, bei denen eine zuverlässige Übertragung der Daten gefordert wird.

Der Header, der beim Zusammenbau eines Datensegmentes den Daten aus der Anwendungsschicht vorangestellt wird, wird durch RFC 793 definiert.



TCP-Header Format

Jede TCP Nachricht enthält die Bestandteile

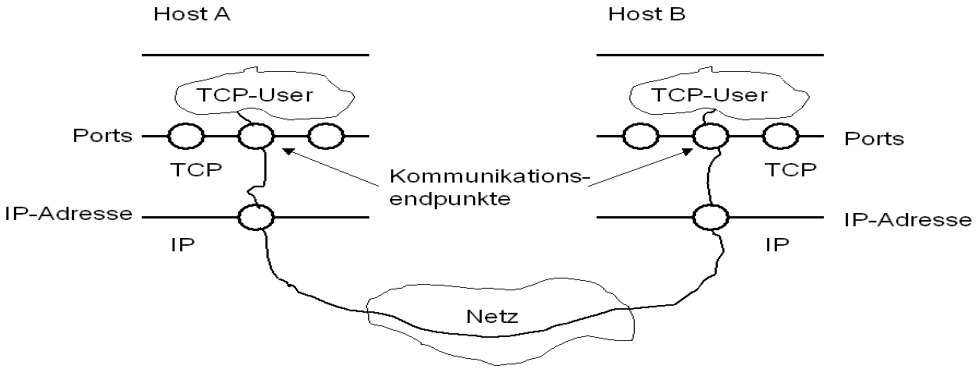
- Quellsystem
- Quellport
- Zielsystem
- Zielport
- Zusätzlich enthalten sind:
- Sequenznummer Sequenznummer des ersten Datenbytes
- Quittungsnummer Nächste, vom Sender erwartete Sequenznummer
- Daten-Offset Länge des TCP-Kopfes in 32-Bit-Wörtern

Reserviert sind folgende Nutzdateninhalte:

URG Urgent-Frame: Zeigt auf wichtige, sofort weiterzuleitende Daten
SYN Sync Sequence-Numbers: Für Verbindungsaufbau zuständig Kontrollfelder
ACK Acknowledgement-Frame: Zeigt an, dass Quittungsnummer gültig ist
RST Reset Connection: Zurücksetzen einer Verbindung
PSH Push (weiter ohne resync): Sofortige Auslieferung der Daten
FIN Finish/End-Frame: Für Verbindungsabbau zuständig

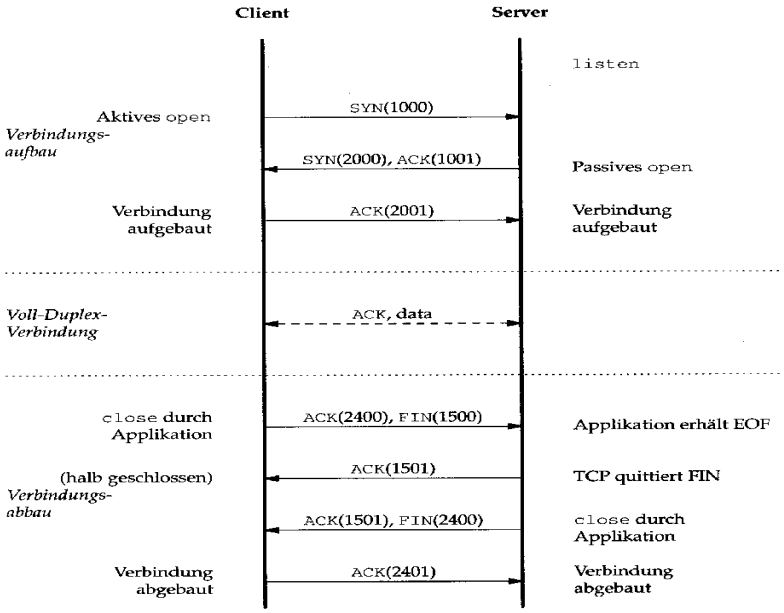
Ist ein Datenpaket beim Zielrechner angekommen, so muss es an eines der auf dem Zielrechner

laufenden Programme vermittelt werden. Diese Zuordnung geschieht anhand der Portnummer, einer 16-Bit-Zahl, die im Header des Transportlayerprotokolls enthalten ist. Jedes Programm, das Daten vom Netz empfangen will, muss sich mindestens eine Portnummer zuordnen lassen. Abbildung 20 zeigt die Beziehung zwischen Port-Nummern und IP-Adressen:



TCP-Sockets

TCP erbringt der Anwendungsschicht einen zuverlässigen, verbindungsorientierten Dienst. Der Ablauf einer TCP-Sitzung sieht dabei wie folgt aus:



Ablauf einer TCP-Sitzung

Das Transmission Control Protocol stellt einen zuverlässigen Datenübertragungsservice zwischen zwei Netzwerksrechnern dar. Im TCP-Protokoll werden alle zu übertragenden Datensegmente mit einer Sequence Number versehen, um Pakete, die aufgrund unterschiedlicher Übertragungswege vertauscht wurden, wieder in die richtige Reihenfolge zu bringen und um verlorengegengene Pakete erneut anfordern zu können.

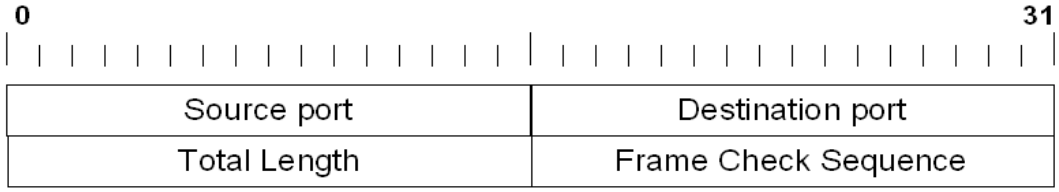
Das TCP-Protokoll ist ein verbindungsorientiertes, zustandsbehaftetetes Datenübertragungsprotokoll, d.h. zwei Netwerksanwendungsprogramme bauen eine Verbindung auf und übertragen darüber ihre Daten. Erst wenn die beiden Programme keinen Kontakt mehr wünschen wird, die Verbindung wieder aufgegeben.

Bevor mit der eigentlichen Datenübertragung begonnen werden kann, wird ein sogennanntes 3-Weg Handshake ausgeführt. Im ersten Datensegment schickt der Rechner, der die Verbindung aufbauen will, die erste Sequence Number und setzt das *Synchronize sequence number* (SYN) Bit. Der die Verbindung annehmende Rechner antwortet mit einem Datensegment, das seine erste Sequence Number beinhaltet und zusätzlich zum SYN-Bit das Acknowledgment (ACK) Bit gesetzt ist. Diese Dateneinheit transportiert die initiale Sequenz-Nummer i für den Datenfluss von Instanz A zu Instanz B (i wird zeitlich gesteuert, um sicherzugehen, dass keine alten Duplikate von Dateneinheiten mehr im Netz kursieren.)

Der initiierende Rechner sendet nun ein Datensegment, in dem nur mehr das ACK-Bit gesetzt hat, um den Erhalt des zweiten Datensegmentes zu bestätigen, begleitet durch das Setzen der Sequenz-Nummer für den Datenaustausch von Instanz B nach Instanz A.
Sollte es trotz zeitlich gesteuerter Sequenznummer doch dazukommen, dass ein Duplikat eines Verbindungsaufbau-Paketes verspätet eintrifft, so empfängt Instanz B die veraltete SYN-Dateneinheit mit Sequenznummer j und bestätigt diese normal.
Instanz A empfängt sie zwar, verweigert aber die Verbindung, weil sie sich nicht im SYN-Zustand (Verbindungsaufbauzustand) befindet, und signalisiert das durch versenden einer RST-Dateneinheit.
Ein weiterer möglicher Fehler beim Verbindungsaufbau kann durch das verspätete Eintreffen einer SYN\ACK-Dateneinheit hervorgerufen werden. Instanz A hat eine SYN-Dateneinheit weggesendet und erhält danach eine verspätete SYN\ACK-Dateneinheit der Instanz B. Sie weist diese jedoch durch Versenden einer RST-Dateneinheit sofort zurück, und akzeptiert erst die danach empfangene, korrekte SYN\ACK-Dateneinheit. Bei Instanz B wird der Verbindungsaufbau durch das Eintreffen der RST-Dateneinheit nicht beeinträchtigt.
Netzwerksanwendungen, die das TCP-Protokoll verwenden können sich darauf verlassen, dass die dem TCP übergebenen Daten auf der anderen Seite der Netzwerksverbindung unverändert ankommen.
Für jede solche Verbindung wird die Reihenfolge also durch eine Laufnummer bestimmt, die beim Verbindungsaufbau von der Gegenseite quittiert werden muss.
Kann ein Angreifer die Startlaufnummer vorhersagen, kann er eine vertrauenswürdige Verbindung vortäuschen. Mit Authentifikationsprotokollen, die sich auf IP-Quelladressen abstützen kann dann in das Zielsystem eingedrungen werden. Dies ist als Laufnummerattacke bekannt.

User Datagram Protocol (UDP)

Das *User Datagram Protocol* (UDP), ebenfalls ein Protokoll der Transportschicht, stellt die Dienste von IP praktisch transparent der Applikation zur Verfügung. Die Sicherheitsvorkehrungen von TCP entfallen. Es findet kein Verbindungsaufbau statt. Das User Datagramm Protocol stellt einen direkten Zugang zur Datenübertragung ins Internetprotokoll der Netzwerksschicht dar. Es werden beim UDP-Protokoll keine unnötigen Kontrolldaten übertragen, dafür kann aber auch keine verlustfreie Übertragung garantiert werden. Ebenso kann die Reihenfolge der Datensegmente vertauscht werden. Der Header, der beim Zusammenbau eines Datensegmentes den Daten aus der Anwendungsschicht vorangestellt wird, hat das in RFC 768 definierte Aussehen.



UDP-Header Format

Das UDP-Protokoll wird meist dann angewendet, wenn kleine Mengen von Daten schnell ausgetauscht werden sollen, ohne dass vorher bekannt ist, wann die Übertragung benötigt wird. Werden nur geringe Anforderungen an die Sicherheit der Übertragung gemacht, so kommt oft UDP zum Einsatz, da es keine Überprüfung der Daten vornimmt.
UDP-Pakete sind damit leichter zu fälschen. Es gibt keine Quittungs- und keine Laufnummern.

Port-Nummern (Services)

Für die bekannteren Programme sind Portnummern festgelegt und den Ports sind Namen zugeordnet. Bei UNIX-Rechnern findet sich diese Liste der Portnummern und Namen in der Datei /etc/services. Eine Aufstellung aller offiziell vergebenen Portnummern ist dem RFC 1700 zu entnehmen.
Die Kommunikation erfolgt auf der Transportebene über sogenannte Portnummern. Diese ermöglichen das Ansprechen unterschiedlicher Dienste. Zusammen mit den IP-Nummern bilden die

Portnummern Kommunikationsendpunkte.
Die Portnummern bestehen aus 16 bit, so dass von einem Rechner maximal 65.535 unterschiedliche Ports realisiert sein können. Für einige Standarddienste werden von der IANA (Internet Assigned Numbers Authority) sog. Well-Known-Portnummern vergeben. Dies sind Vereinbarungen, Portnummern auf welche Ports bestimmte Dienste hören.
FTP-Servern ist der Port mit der Nummer 21 zugeordnet, Telnet hört auf Port 23, Gopher-Server auf den Port mit der Nummer 70.
Portnummern sind auf einen Rechner beschränkt, sie werden für TCP und UDP getrennt vergeben, d.h. die Portnummer 4711 für UDP spezifiziert einen anderen Dienst als die Portnummer 4711 für TCP. Unter UNIX erfolgt die Zuordnung vom Dienst zum Portnummer/Protokoll-Paar in der Datei /etc/services. Um bei der Analogie zum Telefonsystem zu bleiben, stehen Portnummern auf der gleichen Stufe wie Nebenstellenanlagen. Die Netzadresse entspricht dabei der Ortsvorwahl, die Hostadresse der Rufnummer, und der Port entspricht der Durchwahl. Mit diesen Angaben kann eine Verbindung zu einem bestimmten Dienst aufgebaut werden. Well-Known-Ports können nun mit der Auskunft, der Störungsstelle oder der Zeitansage verglichen werden, also Dienstleistungen, die unter einer allgemein verbreiteten Nummer erreichbar sind.
Multiuser/Multitasking-Umgebungen wie UNIX oder Windows NT erlauben die quasi gleichzeitige Bearbeitung von Aufgaben. Serverprozesse sind oft Hintergrundprozesse (daemon = Disk And Execution MONitor). Diese bauen einen Kommunikationspunkt oder Socket auf. Ein Client koppelt sich an einen Socket und beantragt beim Server eine Verbindung. Dieser nennt dem Client einen anderen Punkt über den die Kommunikation stattfinden kann.
Aus Sicherheitsgründen werden Portnummern kleiner als 512 nur an Programme vergeben, die unter Superuserrechten laufen. Damit wird verhindert, dass ein unbefugtes Benutzerprogramm sich als ein offizieller Dienst ausgibt, und dann vom ahnungslosen Nutzer, der eigentlich den offiziellen Dienst kontaktieren wollte, den Benutzernamen und das Passwort abfragt, um damit Missbrauch zu betreiben.
Damit nicht für jeden Netzwerkservice, den ein Unix-Rechner anbietet, ein eigener Prozess gestartet werden muss, nur um den Port zu reservieren und ihn abzuhören, gibt es im UNIX System einen Prozess inetd , der sich unter allen in der Datei /etc/inetd.conf genannten Ports registrieren lässt. Der inetd wartet dann unter diesen Portnummern auf Daten, und erst wenn Daten ankommen, ruft er den in /etc/inetd.conf zugeordneten Serverprozess auf und übergibt ihm die Daten.

Vergleich zwischen TCP- und UDP-Diensten

Funktion	TCP	UDP
Verbindungsaufbau vor der Datenübertragung	ja	nein
End-zu-End-Kontrolle	ja	nein
Zeitüberwachung der Verbindung	ja	nein
Spezialfunktionen	ja	nein
Flusskontrolle über das Netz hinweg	ja	nein
Zuverlässige Datenübertragung	ja	nein
Geschwindigkeit	normal	hoch
Connectionless Service	ja	ja
Erkennung von Datagramm-Duplikaten	ja	nein
Reihenfolgerichtige Übertragung	ja	nein
Multiplexen von Verbindungen	ja	ja

Der Application Layer

Der Application Layer definiert die Dienste, welche im Transport Layer als TCP- und UDP-Services bezeichnet werden. In diesem Kapitel werden die wichtigsten Dienste genauer

beschrieben.

Domain Name System (DNS)

Das Domain Name System (DNS) wurde geschaffen, um Rechnern zusätzlich zu den IP-Adressen auch logische Namen zuordnen zu können. Man braucht sich daher nicht komplizierte Zahlencodes als Adresse eines Hosts zu merken, sondern, sehr viel intuitiver, seinen Namen.

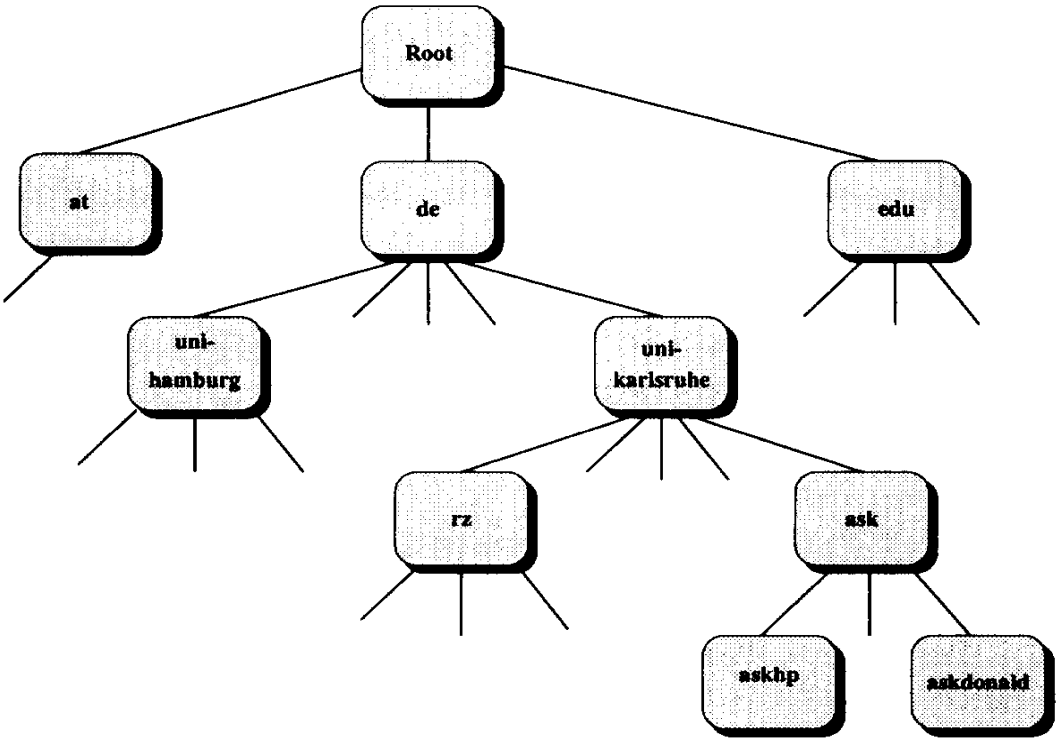
Parallel zu den numerischen Adressen haben praktisch alle Rechner, die dauerhaft an das Internet angeschlossen sind, einen Namen - und zwar in der Form: host.domain. Solche Namen lassen sich wesentlich leichter merken und werden von den sogenannten *Name Servern* in die numerischen IP-Adressen "übersetzt".

Die Zuordnung der Host-Namen zu den IP-Adressen erfolgte in den Anfängen des Internet über eine zentral gehaltene Datei (/etc/hosts auf UNIX Systemen) des *Network Information Centers* (NIC), die an alle Rechner jeder Domain regelmässig mittels ftp verschickt wurde und die jeder IP-Adresse eindeutig einen Namen zuordnete.

Damals war der Adressraum (= die Anzahl der benutzten Adressen) noch klein. Im Laufe der Zeit wurde ein leistungsfähigeres, aber auch aufwendigeres System nötig. Als das Internet wuchs, war eine mittels ftp verschickte hosts-Datei für jeden Rechner nicht mehr möglich. Das Aktualisieren einer solchen Datei wäre zu aufwendig und würde das Netzwerk zu sehr belasten. Die Hosts-Datei ist mittlerweile zu einer grossen Datenbank angewachsen und wird in Zonen aufgeteilt und von dedizierten Rechnern, den *Domain Name Servern* (DNS-Server), innerhalb der Zone verwaltet. Es handelt sich also um eine dezentralisierte Datenbank.

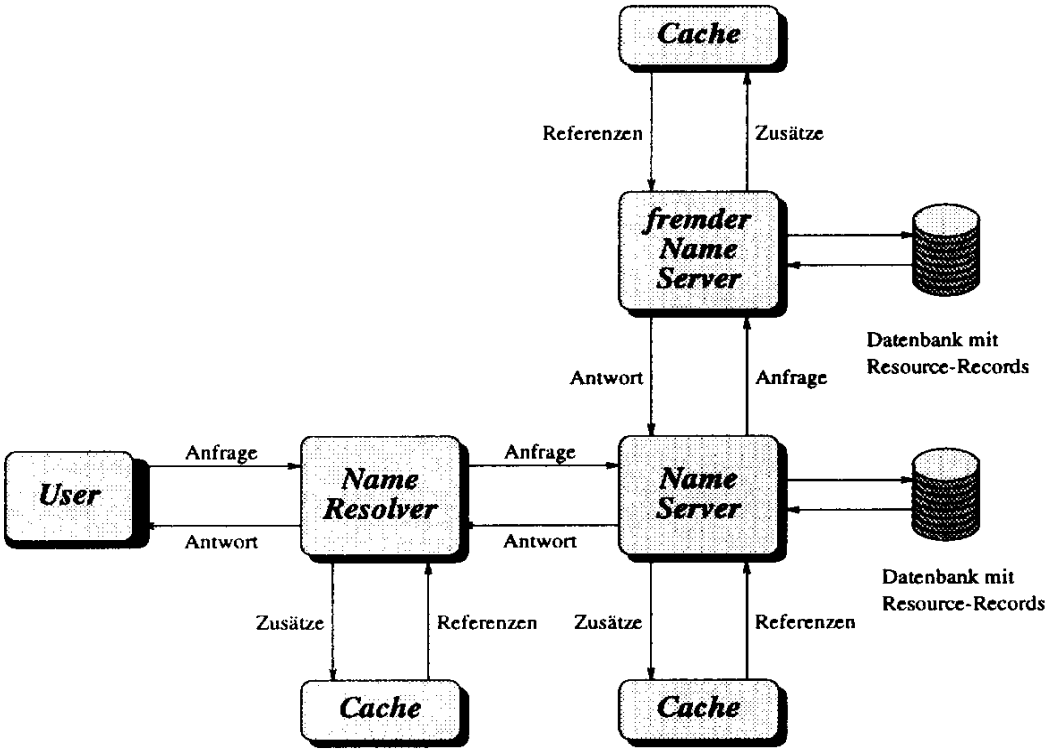
Jeder *Domain Name Server* verwaltet in seinen Tabellen nur einen Teil des inzwischen riesigen *Domain Name Space*. Insgesamt existieren drei Komponenten, aus denen sich das DNS-Verwaltungssystem zusammensetzt.

1. Der *Domain Name Space* ist ein baumartiger, hierarchisch strukturierter Namensraum, der die Resource Records enthält. Das sind Datensätze, die den Knoten zugeordnet sind.
2. *Name-Server* sind Rechner im Internet, die die Informationen über die Struktur des Domain Name Space speichern und zugänglich machen. Ein Name-Server hat normalerweise nur einen Teil des Domain Name Space zu verwalten.
3. *Resolver* sind Programme, die für den Client Anfragen an den Name Server stellen. Resolver sind einem Name-Server zugeordnet. Bei Anfragen, die er nicht beantworten kann, kann er aufgrund von Referenzen andere Name-Server kontaktieren, um die Information zu erhalten.



Domain Name Space

Der Domain Name Space ist als baumartige Struktur angelegt. Ausgehend von der Wurzel (root) folgen die Top Level Domains wie com, edu oder ch. Diese spalten sich in weitere Unterdomains auf. Die Name-Server verwalten also Zonen, die einen Knotenpunkt im DNS-Baum und alle darunterliegenden Zweige beinhalten. Durch die Existenz von Name-Servern auf verschiedenen Tiefen des DNS-Baumes überlappen sich die Zonen der verschiedenen Name-Server. Ein Name-Server kennt jeweils seinen nächsthöheren und nächsttieferen Nachbarn. In jeder Zone gibt es aus Sicherheitsgründen mindestens zwei aktive Name-Server (primary und secondary), die beide dieselben Informationen liefern. Für den Bereich ch wird der Primary Server vom CH-NIC, der Switch AG (http://www.switch.ch), betrieben.
Damit das DNS überhaupt funktioniert muss jedes Netzwerk, das an das Internet angeschlossen ist über einen sogenannten Name Server verfügen, der die logischen Namen des DNS den IP Nummern zuordnet.



Ablauf einer DNS-Abfrage

Diese Name Server sind hierarchisch miteinander verbunden, so dass nicht jeder Name Server alle Adressen kennen muss. In kleineren Netzen, ohne Anschluss an das Internet reicht es jedoch die einzelnen Rechner untereinander über eine Konfigurationsdatei bekannt zu machen. Soll eine Verbindung zu einem Rechner im Internet hergestellt werden so wird erst bei dem zugehörigen Name Server die Adresse erfragt, und dieser schickt dann die entsprechende IP zurück, sodass dann ein Verbindungsaufbau stattfinden kann. Ein Rechner kann auch mehrere Aliasse haben, so ist der Rechner 139.6.57.5 sowohl als www.w1.com als auch als ftp.ftp1.com erreichbar.

Well Known Port Numbers:

domain 53/tcp Domain Name Server

domain 53/udp Domain Name Server

Was ist IPv6 ?

Jeder Computer im Internet hat seine eigene Adresse, eine sogenannte IP (Internet Protocol). Das in den 80'er Jahren konzipierte Format und Protokol für diese Adressen heisst IPv4. Damals ging man verständlicher Weise noch nicht davon aus, dass es keine 10 Jahre später ein weltumspannendes Internet) geben wird, in dem Millionen von IPs zu vergeben werden müssen. Ein Problem, welches mit der Zeit mehr an Bedeutung gewinnen wird ist die begrenzte Anzahl an IP-Adressen. Auf die Tragweite dieser Problematik wurde man erst Mitte der 90'er Jahre aufmerksam, als es zu einem explosionsartigem Anstiegt der privaten Internetbenutzer kam. Zu den Anfangszeiten des Internet war es nicht üblich, sich von zu hause via Modem in das Internet einzuwählen, da damals die Verbindungen sehr langsam und meist auch extrem teuer waren. So wurde das Internet primär als eine Art "Forum für Universitäten" genutzt. Erst, als Mitte der 90'er Jahre die Verbindungen schneller, und die Preise niedriger wurden, gewann das "Web" auch für die Allgemeinheit an Bedeutung. Bis heute, benutzen mehrere Millionen Menschen diese Dienste. Aber im 21. Jahrhundert wird das Internet nicht nur vom heimischen Rechner aus genutzt werden, sondern auch mobil (Handy, Handhelds, etc.). Da jeder Rechner, eine eben solche IP-Adresse braucht, ist es nicht schwer festzustellen, dass der unter IPv4 zur Verfügung stehende Adressraum (32 Bit) nicht lange ausreichen wird. Spätestens wenn jeder Kühlschrank und jede Waschmaschine einen Internetzugang haben, was durchaus denkbar wäre, würde es zu IP-Engpässen kommen.

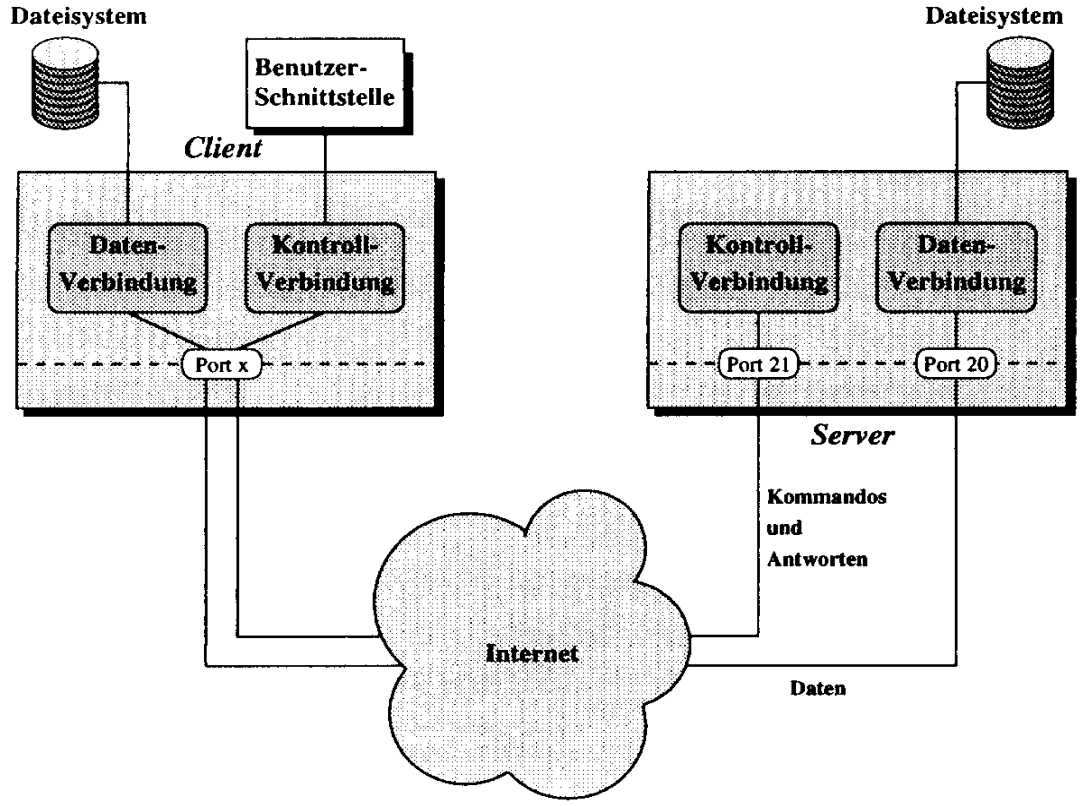
Die "IP-Knappheit" dürfte mit IPv6, auch "IP Next Generation" genannt (IPNG), wohl vorrüber sein. So ist es mit IPv6 theoretisch möglich 655,570,793,348,866,943,898,599 Adressen pro Quadrat-Meter Erde, zu vergeben. Auch (D)DoS Attacken, wie zum Beispiel auf Yahoo!, eBay, etc. gehören mit der Einführung von IPv6 wohl der Vergangenheit an, da durch das neue Format kriminelle Handlungen wie z.B. das IP-Spoofing enorm erschwert werden. Bis dato unterstützen besonders europäische und asiatische Institutionen und Firmen die Entwicklung und Verbreitung von IPv6. Das ist wohl mit der Tatsache, dass etwa 75% des IPv4-Adressraums den USA zugeteilt wurde, zu erklären. Im Moment unterstützen zwar nur wenige Dienste das Internet Protokol der Zukunft, aber gerade bei der Entwicklung neuer Dienste in diesem Bereich wird es in den nächsten Jahren einen enormen zuwachs geben.

Da zum Thema IPv6 gerade in der deutschen "Internetöffentlichkeit" doch eher magere Informationen zur Verfügung stehen, hoffen wir mit dieser Website dem "Versierten-Internet-User" eines der wichtigsten Themen der Zukunft des Internets etwas näher zu bringen.

Bei weiteren Fragen, Anregungen, können Sie uns gerne via E-Mail kontaktieren. Aber auch über Postings auf dem offiziellen IPv6-Webboard würden wir uns sehr freuen

File Transfer Protocol (FTP)

Wie in RFC 959 detailliert beschrieben, hat das *File Transfer Protocol* (FTP) eine sehr lange Geschichte. Es gibt mehrere verschiedene Clientprogramme (xftp, ncftp, netscape,...), die eine graphische Benutzeroberfläche haben. Das Programm ftp ist das älteste Clientprogramm und wird von der Kommandozeile aus bedient.
Im Allgemeinen wird ftp mit dem Namen des Zielrechners aufgerufen, mit dem Dateien ausgetauscht werden sollen. Das Clientprogramm eröffnet daraufhin eine Verbindung zum Port 21 des Zielrechners. Dort wird vom inetd der ftpd gestartet. Um sich am Zielrechner als Benutzer zu identifizieren, fragt das Clientprogramm nach dem Benutzernamen, unter dem man am Zielrechner bekannt ist, und nach dem Passwort des Benutzers. Beide werden dann an den Zielrechner weitergereicht. War die Identifikation erfolgreich, so kontrolliert von nun an das Clientprogramm die Verbindung, indem es Kommandos an den ftpd schickt, der mit Resultcodes antwortet. Die Resultcodes sind dreistellig Zahlen und dienen als Antwort für das Clientprogramm, der Rest der Zeile ist eine Beschreibung des Resultcodes für den menschlichen Leser. Für die eigentliche Übertragung der Daten aus den zu transferierenden Dateien handeln die beiden Programm bei jeder Datei die Portnummern für eine zweite Verbindung aus, über die der Dateiinhalt transportiert wird.
FTP ist das am häufigsten benutzte Dateitransferprotokoll. Es unterstützt den Transport und die Zeichensatzumkodierung von Text- und Binärdateien.



Funktionsweise von FTP

Anonymous ftp ist der wichtigste Mechanismus zur Verteilung von Programmen und Daten. Normalerweise muss man sich mit einem Benutzernamen und -Passwort bei einem FTP-Server authentifizieren. Anonymous ftp ist ein spezieller Benutzeraccount, der es jedem Benutzer ermöglicht sich (anonym) auf dem FTP-Server einzuloggen. Dazu wird als Benutzernamen 'anonymous' angegeben und als Passwort die eigene E-Mail-Adresse.

Trivial File Transfer Protocol (TFTP)
Das Trivial File Transfer Protocol (TFTP) ist eine sehr vereinfachte Form des FTP. Es kann keine Verzeichnisse auflisten und bietet keine Benutzerauthentifikation an. Es dient lediglich der Übertragung von Dateien. TFTP basiert im Gegensatz zu FTP auf den Diensten von UDP. Es wird benutzt um diskless-Workstations zu booten und Router-Konfigurationen zu laden.

Terminal-Emulationen (Telnet)
Das am weitesten verbreitete Protokoll für eine einfache Terminalemulation ist das TELNET Protokoll. Neben der Terminalemulation ermöglicht TELNET die Aushandlung von Optionen (z.B. Zeichensatz, Line-Mode) zwischen zwei Kommunikationspartnern. Es gibt eine grosse Anzahl verschiedener Telnet-Implementationen, welche unterschiedliche Systeme emulieren.
Telnet bietet einen einfachen Terminalzugang zu einem System. In der Regel rufen Telnet-Dämonen zum Authentifizieren und Initialisieren einer Sitzung Login auf, welches dann einen Benutzernamen und meist auch ein Passwort erhält.
tinu@speedy:/home/tinu > telnet speedy
Trying 192.168.11.11...

HyperText Transfer Protocol (HTTP)
Das *Hyper Text Transfer Protocol* (http), welches erst im Mai 1996 in RFC 1945 festgelegt wurde[20], ist eine Weiterentwicklung des Gopher-Protokolls. Bei http nimmt der Server die Verbindung an, ohne eine Begrüssung auszugeben. Der Client kann nun eine Zeile senden. Will der Client Daten vom Server erhalten, so beginnt die Zeile mit GET und beinhaltet dann den Zugriffspfad zu der gewünschten Information. Diese Information wird dann vom Server an den Client geschickt und die Verbindung wird wieder abgebaut.
Eine http-Verbindung kann auch mit Telnet simuliert werden. Dazu startet man eine telnet-Verbindung auf den Port 80 des Zielrechners
tinu@speedy:/home/tinu > **telnet speedy 80**
Trying 192.168.11.11...

Will der Client Informationen wie z.B. den Inhalt eines vom Benutzer ausgefüllten Formulars an den Server übermitteln, so beginnt die vom Client gesendete Zeile mit dem Wort ***PUT*** gefolgt von dem Zugriffspfad des vom Server abzuarbeitenden Programmes, welches die in den folgenden Zeilen vom Client zu sendenden Informationen verarbeiten soll. Die von diesem Programm als Reaktion auf diese Informationen ausgegebenen Antworten werden vom Server an den Client zurückgegeben.
Well Known Port Numbers:
www-http 80/tcp World Wide Web HTTP
www-http 80/udp World Wide Web HTTP

Simple Mail Transfer Protocol (SMTP)
Eine der interessantesten Möglichkeiten des Internet ist die elektronische Post. E-Mail im Internet wird meistens über *Simple Mail Transport Protocol* (SMTP) transportiert. SMTP ist ein einfaches Protokoll zum Transport von ASCII-Zeichen.
Voraussetzung für die weltweite Kommunikation über E-Mail sind eindeutige Mailadressen. Im Internet ist durch das DNS die Eindeutigkeit von Rechnernamen gewährleistet. Da auf Rechnern Benutzeraccounts eindeutig sein müssen, liegt es nahe, die Kombination Benutzer/Domain als eindeutige Identifizierung für Mailadressen zu verwenden
SMTP funktioniert über eine ASCII-Zeichen basierte Steuerung. Obwohl es heute für die unterschiedlichsten Ansprüche Mail-Applikationen gibt, kann man auch mit telnet eine Mail über SMTP absenden:
tinu@speedy:/home/tinu > **telnet speedy 25**
Trying 192.168.11.11...

Post Office Protocol Version 3 (POP3)
Um Mail nur mit SMTP zu verwalten, ist ein ständig aktiver Mailserver (*Message Transport System* MTS) und eine stehende Leitung zum Internet auf der lokalen Workstation notwendig. Auf kleineren Internet-Knoten, wie z.B. ein PC zuhause, macht dies keinen Sinn oder ist schlicht zu teuer.
Trotzdem möchte man auch auf kleineren Knoten die Möglichkeit haben, Mail zu empfangen und zu verwalten. In genau diese Lücke von SMTP springt das *Post Office Protocol Version 3* (POP3). POP3 ermöglicht einem PC Mail von einem Mail-Server abzuholen. Es gestattet nicht sehr viel weitere Operationen:
Mail vom Server holen, aber nicht löschen
Mail nach dem Download löschen
Nur neue Mail vom Server holen
POP3 dient nur dem Empfangen von Mail. (Internet-)Mail-Programme auf PCs benutzen zum Empfangen von Mail POP3 und zum Senden SMTP.
POP3 funktioniert nach dem gleichen Prinzip wie auch SMTP und HTTP. Es werden zeilenweise Kommandos und Antworten gesendet. Auch POP3 kann man im Prinzip mit telnet testen:
tinu@speedy:/home/tinu > **telnet speedy 110**
Trying 192.168.11.11...

Internet Message Access Protocol Version 4 (IMAP4)
Das *Internet Message Access Protocol Version 4* (IMAP4) erlaubt einem Client den Zugriff auf und die Manipulation von Mail auf einem Server. Es erlaubt das Erstellen von entfernten Mail-Verzeichnissen (Mailboxes), gleich wie man es von einem komfortablen POP3-Mail-Programm gewohnt ist. Einem IMAP4-Client ist es möglich, sich mit dem IMAP4-Server zu synchronisieren. So kann man zum Beispiel im Büro und zuhause mit einem IMAP4-Mailprogramm auf den gleichen IMAP4-Server zugreifen und hat jederzeit Zugriff auf alle Mail.
IMAP4 ermöglicht eine Menge verschiedener Operationen:
Erstellen, Löschen und Umbennen von Verzeichnissen (Mailboxes)
Suchen nach neuen Mails
Mails permanent löschen
Flags setzen und entfernen
Suchen nach Inhalt und Attributen
Selektiver Download
IMAP4 definiert wie POP3 keine Funktionen zum Senden von Mail. Diese Funktion wird durch SMTP abgedeckt.