Wie funktioniert das Internet?

Stand: 13. März 2003

Kontakt:
schule@barmetler.de

Inhaltsverzeichnis

1 Anschauliche Betrachtung des Internet

Nachfolgend werden die prinzipiellen Schritte beim Erstellen, Veröffentlichen und Abrufen einer Webseite im Internet erläutert (vgl. Abbildung auf Seite 6).

Ein Entwickler möchte eine Webseite auf einer eigenen Homepage veröffentlichen.

1. Dazu sucht er sich zunächst einen Internet Service Provider (ISP). Dieser bietet seinen Kunden verschiedene Dienste wie einen Webserver, Mailserver, Datenbanken, DNS-Server, ... Üblicherweise sind diese Dienstleistungen kostenpflichtig. Als Alternative wird oft die Einblendung von Werbebannern auf der Homepage des Kunden angeboten.

   Anschließend muss sich der Entwickler eine Adresse (=Domain, siehe 2.1) in der Form *www.barmetler.de* überlegen, unter der seine Webseiten im Internet erreichbar sein sollen. Prinzipiell kann man sich seine Wunschadresse frei aussuchen - auch Phantasienamen sind also möglich -, aber jede Adresse kann weltweit nur einmal vorkommen! Um dies sicherzustellen muss jede Adresse an einer zentralen Stelle registriert werden. Eine Organisation namens DENIC übernimmt diese wichtige Aufgabe für alle Internetadressen die auf „de“ enden.

   Aus technischen Gründen, und um gewisse Formalitäten einhalten zu können, dürfen nur bestimmte Institutionen einen Antrag auf Registrierung bei der DENIC einreichen. Deshalb meldet unser Entwickler die Wunschadresse seinem ISP. Der reicht den Antrag an die DENIC weiter. Dort wird nun überprüft ob diese Domain noch frei ist, oder ob eine andere Person die gleiche Adresse bereits früher registrieren lies. Ist sie noch zu haben wird die Adresse auf den Namen unseres Entwicklers (nicht auf den ISP, obwohl der den Antrag eingereicht hatte!) eingetragen.

2. Während der Antrag in Bearbeitung ist, kann der Entwickler die Webseiten programmieren. Dies kann in einem einfachen Texteditor, aber auch in speziellen Webeditoren (z. B. Microsoft's Frontpage oder Macromedia's Dreamweaver) vorgenommen werden.

3. Die Adresse, welche sich unser Entwickler ausgedacht hat, ist meist ein einprägsames Wort bzw. ein Name. Diesen gibt ein Benutzer üblicherweise in die Adresszeile seines Browsers ein. Leider können technische Systeme wie Computer damit recht wenig anfangen. Sie benötigen zur Bearbeitung die allseits bekannten Nullen und Einsen. Deshalb gibt es neben dem Domainnamen noch die so genannte IP-Adresse. Dies ist eine eindeutige Zahl, welche, ebenso wie die bei der DENIC registrierte Adresse, die Webseiten unseres Entwicklers adressiert. Doch woher weiß man (bzw. der Computer) nun welche Domain zu welcher IP-Adresse gehört?

   Nachdem der Registrierungsvorgang bei der DENIC abgeschlossen wurde, bekommt der ISP eine Mitteilung darüber. Daraufhin weist er der Domain eine IP-Adresse zu. Damit auch alle anderen Computer im Internet diese Zuordnung kennen, gibt es wieder verwaltende Instanzen: bestimmte

URL	IP-Adresse (Dezimalzahl)	IP-Adresse (Dualzahl)
barmetler.de	212.227.109.234	11010100.11100011.01101101.11101010

Tabelle 1: Zuordnung von URL zu IP-Adresse im DNS-Server

Computer im Internet verwalten riesige „Tabellen“ in welchen die Zuordnungen „Domain ↔ IP-Adresse“ gespeichert ist (siehe Tabelle 1). Diese Computer haben die Bezeichnung *Nameserver* oder *DNS-Server*.

4. Nun kann der Entwickler die Seiten auf den Webserver bei seinem ISP kopieren. Man nennt diesen Vorgang: „Einen Upload machen“. Er benutzt dafür ein spezielles Programm (einen FTP Client), das meist aus zwei Bereichen aufgebaut ist. In einem Bereich werden (ähnlich wie beim Windows-Explorer) die Verzeichnisse und Dateien auf dem eigenen, lokalen Computer angezeigt. Der andere Teil stellt die Ordner und Dateien des Webservers dar. Nun können die neu programmierten Webseiten mit wenigen Mauslicks vom lokalen Rechner auf den entfernten Server kopiert werden. Damit sind die Webseiten auf dem Server des ISP unter der gewählten Domain öffentlich erreichbar.

5. Ein User, der die vom Entwickler bereitgestellten Seiten ansehen möchte, wählt sich über einen Internet-Zugangsprovider in das Internet ein. Dann tippt er die URL (welche er natürlich kennen muss) - z. B. *www.barmetler.de* - in die Adressleiste seines Browsers ein.

6. Diese URL wird an den DNS-Server des Zugangsproviders übermittelt.

7. Anhand der Einträge in seiner Tabelle (vgl. Tabelle 1) sucht der Nameserver die zugehörige IP-Adresse heraus und liefert sie an den Browser des Users zurück.

8. Mit Hilfe der nun bekannten IP-Adresse der gewünschten Webseite in Form von „Nullen und Einsen“ kann der Computer des Benutzers endlich den Server des ISP, auf dem die Seiten liegen, ansprechen.

   Dummerweise bietet der ISP (wie oben bereits erläutert) jedoch neben dem Webserver meist noch andere Dienste an. Und um die Sache nicht zu einfach zu machen, sind diese oft unter der gleichen IP-Adresse erreichbar. Damit der Server nun weiß dass er Informationen vom Webserver und nicht vom Mailserver ausliefern soll, übermittelt der Browser automatisch eine Art Kennziffer, die eindeutig den Dienst *Webserver* anspricht. Diese Kennziffer nennt man *Port*.

9. Fordert der Benuter eine Webseite mit ausschließlich statischem Inhalt an, so geht es direkt mit dem nächsten Punkt weiter.

   Hat der Benutzer dagegen eine Webseite mit dynamischen Inhalt - das könnten z. B. Wetterinformationen, Börsenkurse, Informationen wie viele Teile noch vorrätig sind, ... sein - aufgerufen, so muss die Seite erst zusammengesetzt werden.

Dazu werden die benötigten Informationen zu dem Zeitpunkt wo die Webseite angefordert wurde „zusammengetragen“. D. h. man ruft sie ganz aktuell von einem Zulieferer ab, oder liest sie aus einer Datenbank aus. Diese dynamischen Inhalte werden zusammen mit den vorliegenden statischen Teilen (z. B. einer Adresse, der Begrüßung, ...) der Webseite zu einer Webpage verknüpft.

10. Egal ob die Webseite nun statischen Inhalt hat oder erst dynamisch zusammengesetzt werden musste: am Ende wird sie an den Rechner ausgeliefert, von welchem der Benutzer sie angefordert hatte.

Abschließend noch ein Versuch diesen Ablauf anschaulich zu erläutern:
Herr Topfkucker (*„Entwickler“*), ein exzellenter Küchenchef, möchte ein Restaurant eröffnen. Er will zwar kochen, sich aber nicht um die ganze Verwaltungsarbeit, die beim führen eines Restaurants anfällt, kümmern. Deshalb sucht er sich einen Dienstleister (*„Suche eines ISP“*), der für ihn die Räumlichkeiten (*„Server“*) bereitstellt. Der Koch überlegt sich noch einen Namen für sein Restaurant und teilt ihn seinem Dienstleister mit. Dieser meldet den Namen beim Gewerbeamt (*„DENIC“*) an, so dass er geschützt ist.

In der Zwischenzeit kreiert der Maestro bereits die ersten Menüs (*„programmieren“*) und schickt (*„upload“*) dem Dienstleister die ersten Essen. Der stellt sie in die Wärmetheke (*„Veröffentlichung auf dem Webserver“*) wo sie die Gäste abholen können.

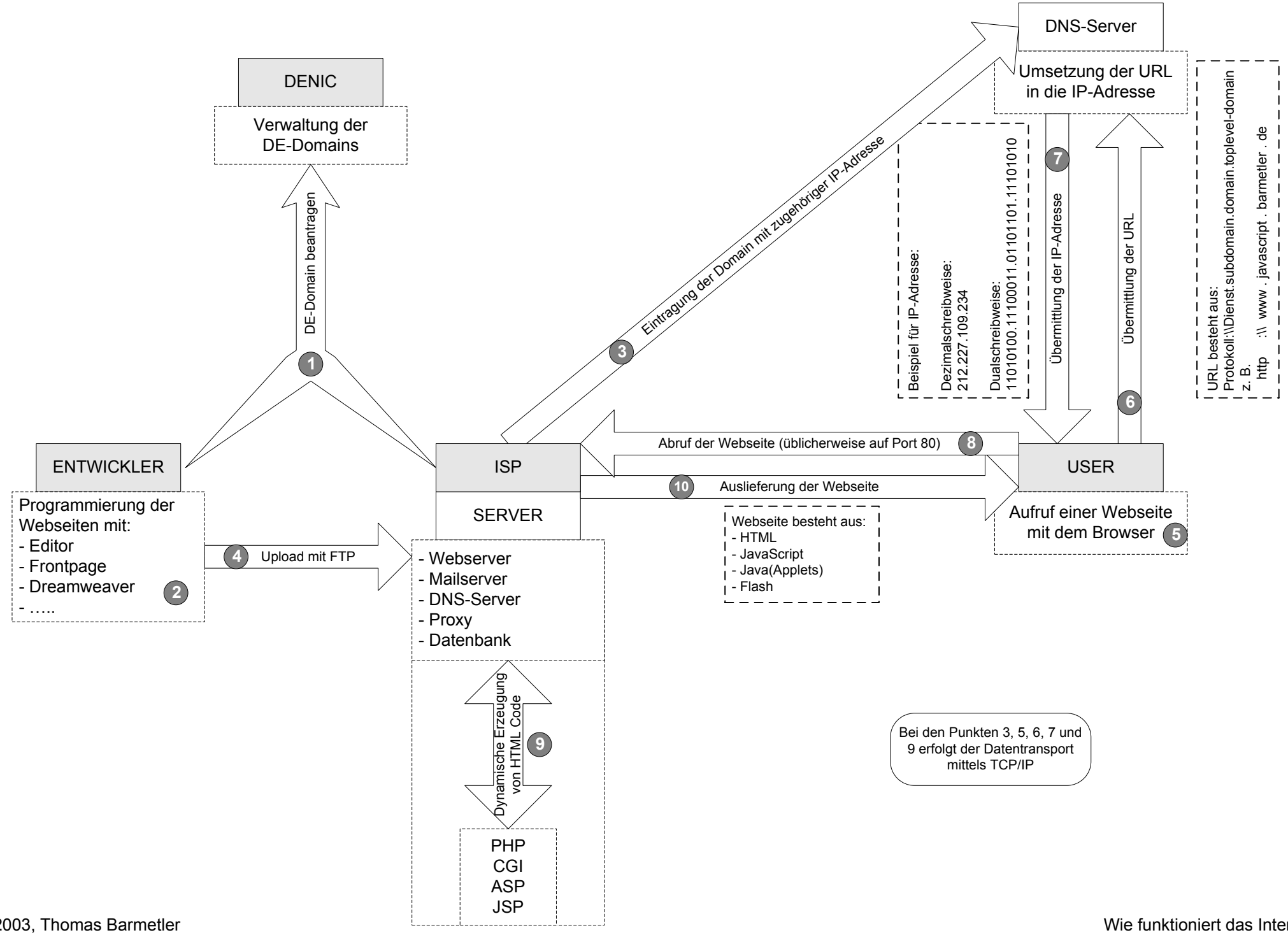
Der Dienstleister trägt das neue Restaurant in der Zwischenzeit bei den Gelben Seiten (*„DNS-Server“*) ein, so dass es auch öffentlich bekannt wird.

Kurz darauf hört das Ehepaar Nimmersatt von Bekannten, dass es einen neuen Gourmettempel geben soll. Sie suchen in den Gelben Seiten nach dem Namen (*„URL, Domain“*) des Restaurants und finden so die Adresse (*„IP-Adresse“*) und Telefonnummer heraus.

Gleich nachdem Frau Nimmersatt dort telefonisch eine Bestellung aufgegeben hat, fährt ihr Mann auch schon los, um die Speisen abzuholen (*„Abruf der Webseite“*). In der Schlemmergasse sieht er, dass es dort mehrere Restaurants (*„verschiedene Dienste des ISP“*) gibt. Also ruft er daheim an und fragt nach, wo er es denn genau abholen soll. Seine Frau teilt ihm also noch die Hausnummer (*„Port“*) mit.

Endlich am Restaurant angekommen teilt der Dienstleister unserem Herrn Nimmersatt mit, dass er zwar das Standardgericht (*„statische Webseite“*), das seine Frau bestellt hatte, sofort aus der Theke könne, aber auf die Pizza mit den besonderen Zutaten (*„dynamische Seiten“*) müsse er noch etwas warten, weil sie erst zusammengestellt werde.

Als auch diese nach kurzer Zeit fertig ist, kann Herr Nimmersatt beide Gerichte einpacken, und sie nach Hause mitnehmen (*„Auslieferung der Webseiten“*).

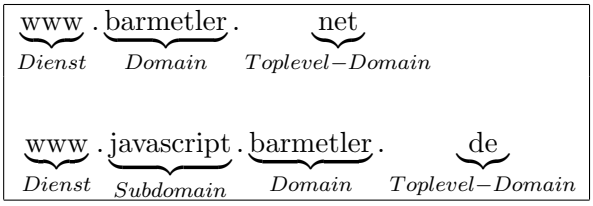


2 Technische Betrachtung des Internet

2.1 Adressierung über URL

Der Uniform Resource Locator (URL) stellt im Internet die Adresse eines Dokumentes dar - vergleichbar mit der Briefanschrift eines Hauses.

Und genau wie diese rückwärts (Land, Stadt, Straße, Name) ausgewertet wird, so geschieht es auch bei der URL. Während die Post die einzelnen Elemente der Adresse durch Zeilenumbrüche auseinander halten kann, erkennen die Server im Internet die einzelnen Teile (hier „Level“ genannt) anhand der trennenden Punkte.



Die höchste, und damit gröbste Ebene steht also ganz rechts, und wird auch als *Toplevel* bezeichnet. Es gibt für jedes Land genau eine Toplevel-Domain. Das Kürzel *de* steht beispielsweise für Deutschland, *uk* für Großbritannien, *it* für Italien, ... Natürlich sind die Länderdomains im Internet nicht an die geographischen Grenzen gebunden, sondern sind weltweit erreichbar und einsetzbar. Oftmals erkennt man daran aber bereits an welches Zielpublikum sich der Entwickler richtet. Bei einer Webpage mit der Toplevel-Domain *de* ist die Sprache der Seite vermutlich deutsch.

Neben diesen Länder-Domains existieren aber auch noch so genannte generische Toplevel-Domains, wie z. B. *com*, *info*, *net*, *gov*, *edu*, ... Diese sind - ebenso wie die Länder-Domains - nicht frei wählbar, sondern standardisiert.

Hierarchisch unterhalb der Toplevel-Domain steht dann die Domain. Dies ist der individuelle Teil der Internetadresse. Er kann frei ausgesucht werden, doch es gilt: First come, first serve. Es bekommt also derjenige einen bestimmten Namen zugesprochen, der ihn zuerst beantragt. Allerdings ist hier durchaus zu beachten, dass beispielsweise ein bekanntes Unternehmen die Freigabe einer Domain, welche ihrem Unternehmensnamen entspricht auf dem juristischen Weg erzwingen kann.

Damit diese Art der Adressierung auch funktioniert, muss natürlich jede Adresse einmalig sein. Deshalb ist es zwingend erforderlich dass eine Domain zunächst bei einer Vergabestelle für die Domainregistrierung beantragt, und (falls sie noch frei ist) registriert wird. Der Antragsteller wird dann als Besitzer der Domain eingetragen. Da für fast jede Toplevel-Domain eine eigene Vergabestelle zuständig ist (für die *de* Domains ist es die DENIC) wäre es sehr mühsam, wenn man eine Domain unter mehreren Toplevel-Domains (z. B. barmetler.net, barmetler.de, barmetler.info,...) registrieren lassen wollte.

Darauf haben sich jedoch einige Dienstleister spezialisiert: die Registrare.

Man übermittelt ihnen - meist in einem Webformular - den gewünschten Domainnamen und die Toplevel-Domains unter welchen die Domain registriert werden soll. Anschließend führt der Registrar eine Online-Abfrage bei den zuständigen Vergabestellen durch, um zu überprüfen, ob dieser Domainname noch frei verfügbar ist. Das Abfrage-Ergebnis wird dem Benutzer meist sofort präsentiert. War die Domain noch frei, so beantragt der Registrar im Auftrag und Namen des Kunden die gewünschte Domain bei den jeweils zuständigen Vergabestellen.

Sobald die Registrierung abgeschlossen wurde erfolgt üblicherweise durch den ISP die erforderliche Eintragung in die Nameserver (DNS-Server), welche eine Voraussetzung für die Erreichbarkeit und die Funktionalität einer Domain ist.

Vereinfacht ausgedrückt verwaltet jeder Nameserver eine große Tabelle, in der jeder Domain eine eindeutige IP-Adresse zugewiesen ist (vgl. Tabelle 2).

Domain	**IP-Adresse**
barmetler.de	212.227.109.234
⋮	⋮

Tabelle 2: Beispieleintrag in einem Nameserver

2.2 Adressierung über IP-Adresse

2.2.1 Die IP-Adresse

Die Notation der IP-Adressen ist spezifiziert. Derzeit gilt noch IPv4 (sprich: IP version 4). Sie legt fest, dass eine IP-Adresse aus 32 Bit besteht, welche in 4 Oktetten, jeweils durch einen Punkt getrennt, angeordnet sind (vgl. Tabelle 1, Seite 4). Damit stehen pro Feld $2^8 = 256$ verschiedene Werte (von 0 bis 255) zur Verfügung. Da dies jedoch sehr schwer zu merken ist wird jede Achtergruppe für sich in eine Dezimalzahl umgewandelt und dann angeschrieben. Der Adressraum des Internet umfasst somit 256^4 (4,3 Milliarden) verschiedene Anschriften. Da ein Teil davon reserviert ist, bleiben etwa 3,7 Milliarden verfügbar - zumindest prinzipiell.

Weil die Adressen nach IPv4 mit der Zeit knapp werden wurde eine neue Spezifikation erarbeitet: die IPv6 (Versionsnummer 5 gab es nie; die 5 kennzeichnete ein experimentelles Protokoll für Echtzeit-Ströme, das ST-2 hieß). Adressen nach IPv6 bestehen aus 128 Bit und werden als Kette von 16-Bit-Zahlen in Hexadezimal-Notation geschrieben, die durch Doppelpunkte getrennt werden: FE80::0211:22FF:FE33:4455. Da die Notation jedoch erheblich schwerer zu durchschauen ist soll hier nicht näher darauf eingegangen werden.

```
 C:> ping barmetler.de

Pinging barmetler.de [212.227.109.234] with 32 bytes of data:

Reply from 212.227.109.234: bytes=32 time=142ms TTL=52
Reply from 212.227.109.234: bytes=32 time=132ms TTL=52
Reply from 212.227.109.234: bytes=32 time=129ms TTL=52
Reply from 212.227.109.234: bytes=32 time=124ms TTL=52

Ping statistics for 212.227.109.234:
       Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
       Minimum = 124ms, Maximum = 142ms, Average = 131ms
```

Tabelle 3: Der Befehl *ping* und seine Antwort

Sind Sie mit ihrem Rechner online, so können Sie die zu einer Domain gehörende IP-Adresse sehr einfach herausfinden. Wechseln sie dazu in ein DOS-Fenster (unter Windows) bzw. eine Shell (unter Unix/Linux). Tippen sie dort den Befehl *ping* gefolgt von dem Domainnamen (mit Toplevel-Domain) ein und bestätigen Sie mit Return. Schon erscheint - neben einigen anderen Informationen - die IP-Adresse (siehe Tabelle 3): im Beispiel 212.227.109.234.

Die anderen Informationen besagen, dass vier Mal ein kleines, 32 Byte großes „Paket“ mit Testdaten an die angegebene Domain geschickt wurde. Für den Weg zu der angegebenen Adresse und wieder zurück (man nennt das den „Round Trip“) benötigten die Pakete eine Zeit zwischen 124 und 142 Millisekunden. Je kürzer diese Zeit ist, desto besser ist die Domain von diesem Rechner aus erreichbar.

2.2.2 Subnetze

Besonders größere Organisationen legen Wert darauf, dass ihre Rechner über möglichst identische IP-Adressen erreichbar sind. D. h. der vordere Teil der IP-Adresse soll für alle Rechner gleich sein, während der hintere Teil den möglichen Zahlenbereich durchläuft (vgl. Tabelle 4). Die 4-Byte-Schreibweise erlaubt die

212.227.109.1
212.227.109.2
212.227.109.3
212.227.109.4
⋮

Tabelle 4: Beispiel eines Subnetzes

Aufspaltung der Routing-Information in zwei Teile, von denen sich der erste auf das Netz, der zweite auf den Hostrechner in dem gegebenen Netz bezieht. Man spricht auch von einer Aufteilung der IP-Adresse in eine Netz- und eine Teilnehmeradresse.

Je nachdem ob nur das erste, oder auch das zweite und dritte Oktett zur Netzwerkadressierung herangezogen werden, ermöglicht dies die Strukturierung des Adressraums in Netzwerkklassen unterschiedlicher Größe. Werden nur die ersten acht Bit verwendet, so spricht man von der Netzklasse A, bei den ersten sechzehn Bit von einem Klasse B Netz, und werden gar die ersten drei Oktette als Netzadresse verwendet, so handelt es sich um ein Klasse C Netz. Die jeweilige Netzklasse ist an den ersten Bits der IP-Adresse zu erkennen. Eine Adresse aus einem Klasse A Netz beginnt mit einer Null, aus einem Klasse B Netz mit den Bits 10 und aus einem Klasse C Netz mit 110.

Oft ist es notwendig aus einer gegebenen IP-Adresse die Netzadresse zu isolieren. Dies geschieht durch eine bitweise UND-Verknüpfung der IP-Adresse mit der Netzmaske. Die Netzmaske wird üblicherweise in dezimaler Form dargestellt und hat, je nach Netzwerkklasse, Werte, wie in Tabelle 5 und Abbildung 2 aufgeführt.

			Länge	
Netzklasse	**führende Bits**	**Netzmaske**	**Netzadresse**	**Teilnehmeradresse**
A	0	255.0.0.0	7 bit	24 bit
B	10	255.255.0.0	14 bit	16 bit
C	110	255.255.255.0	21 bit	8 bit

Tabelle 5: Netzklassen und ihre Kennzeichen

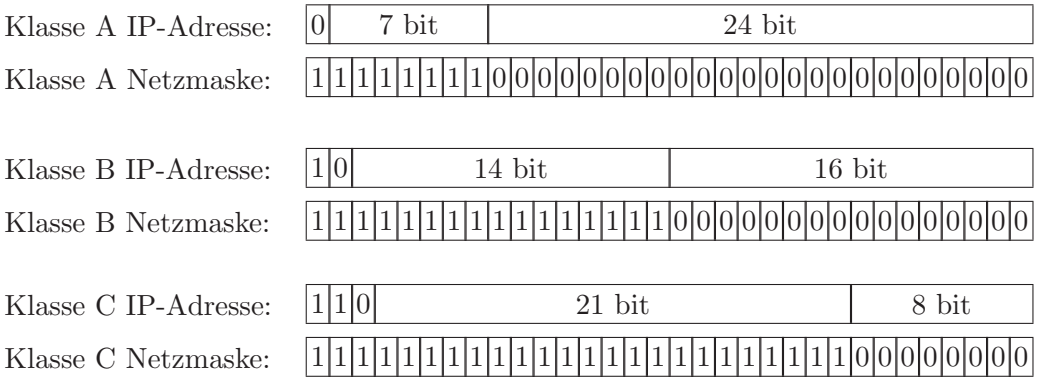


Abbildung 2: Klasse A, B und C IP-Adresse und zugehörige Netzmaske

Besonders innerhalb größerer Institutionen ist es wünschenswert, dass zwar die gesamte Institution nach außen unter einer einheitlichen Netzadresse sichtbar ist, intern jedoch eine weitere logische Unterteilung in Subnetze (z. B. nach Abteilungen) erfolgt. Das kann durch eine flexible Aufteilung der IP-Adresse in Netz- und Teilnehmeradresse erreicht werden. Abbildung 3 veranschaulicht, wie in einem Klasse B Netz (erkennbar an der führenden 10 in der IP-Adresse) mit Hilfe der Netzmaske 255.255.252.0 ein weiteres Subnetz gebildet wird.

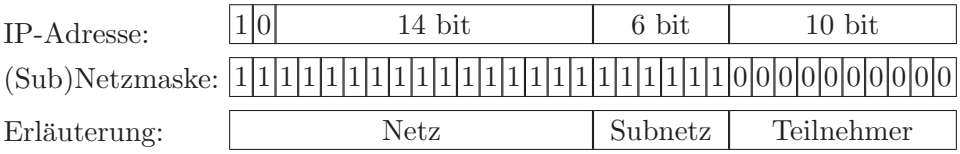


Abbildung 3: Bildung eines Subnetzes in einem Klasse B Netz

Durch die Subnetzmaske wird somit die Netzadrese der IP-Adresse auf Kosten des Teilnehmeradressteiles erweitert. Mit einer Subnetzmaske von 6 Bit kann der 16 Bit Teilnehmeradressteil des Klasse B Netzes so unterteilt werden dass in $2^6 = 64$ Subnetzen jeweils $2^{10} = 1024$ Teilnehmer adressierbar sind.

2.2.3 Umsetzung einer URL in eine IP-Adresse

Bereits bei der Einführung des Internet waren sich die Beteiligten einig, dass eine rein numerische Adresse nicht handhabbar ist. Eine IP-Adresse wie etwa *212.227.109.234* kann sich ein Mensch nun einmal weitaus schlechter merken als die dazugehörigen Domain *barmetler.de*. Aus diesem Grund wurden Domainnamen eingeführt, die es erlauben, Rechnernetze symbolisch anzusprechen. Es spricht aber auch nichts dagegen in die Adresszeile eines Browsers die IP-Adresse einzutippen.

Die eigentliche Kommunikation läuft jedoch immer mittels IP-Adressen ab. Host-Namen sind lediglich eine Hilfestellung für die menschlichen Benutzer. Ursprünglich wurden die für die Namensgebung notwendigen Informationen in einer zentralen Tabelle gespeichert. Das Network Information Center des Defense Data Network hatte die Aufgabe, diese Tabelle zu verwalten. Sie wuchs jedoch schon innerhalb sehr kurzer Zeit so stark an, dass sie nicht mehr wartbar war.

Dieser Umstand führte zur Einführung des *Domain Name System* (DNS) - eines hierarchischen, verteilten Dienstes auf Basis einer Datenbank. Die strenge Hierarchie des DNS ermöglicht erst die Verteilung der Informationen auf verschiedene, unter Umständen weit voneinander entfernte Rechner.

Am einfachsten lässt sich diese Hierarchie als Baum darstellen, wobei jede Verzweigung ihren eigenen Nameserver hat. An der Spitze (oder der Wurzel des

Baumes) steht die Domain *root*. Darunter sind die Toplevel-Domains angeordnet.

Tippt ein Benutzer eine URL in die Adressleiste seines Browsers, so formuliert der so genannte Resolver, im Hintergrund eine Anfrage nach der zu dieser Domain gehörigen IP-Adresse und schickt die Anfrage an einen DNS-Server. Der schlägt den Namen in seiner Datenbank nach und schickt - so seine Suche erfolgreich ist - die Adresse an den anfragenden Computer zurück. Bei Misserfolg übergibt er die Anfrage an den übergeordneten Server zur weiteren Bearbeitung. Dieser Prozess setzt sich solange von Nameserver zu Nameserver von der Sub-Domain über die Domain bis zur Toplevel-Domain fort, bis die IP-Adresse gefunden wurde - oder bis der Nameserver der Toplevel-Domain meldet, dass diese Adresse nicht existiert. Im Erfolgsfall kann der Rechner des Benutzers nun die Seite mit der eingetippten URL bei dem Webserver mit der ermittelten IP-Adresse auf Port 80 (siehe 2.3) anfordern.

2.3 Ports

Ports sind so etwas wie eine „Codenummer“ um einen bestimmten Dienst auf einem Rechner anzusprechen.

Auf einem Computer können beispielsweise gleichzeitig ein Webserver (*Server* ist hier als Software zu verstehen) und ein FTP-Server laufen. Trifft nun an diesem Rechner ein TCP/IP Datenpaket ein, so muss entschieden werden ob es an den Webserver oder den FTP-Server weitergeleitet werden soll. Ein Webserver wird üblicherweise über die Portnummer 80, der FTP-Server über Portnummer 21 angesprochen.

Offizielle Port-Nummern für einzelne Dienste vergibt die Internet Assigned Numbers Authority (IANA). Die Ports 0 bis 1023 sind die 'Well Known Ports' für verbreitete Dienste wie HTTP (Port 80), SSH (22) und Telnet (23). Zwischen 1024 und 49151 liegen die 'Registered Ports', weniger übliche Dienste, bei denen sich jedoch jemand die Mühe gemacht hat, sie bei der IANA zu melden, beispielsweise Kazaa (1214) und Nessus (1241). Der Bereich darüber schließlich steht zur allgemeinen Verfügung.

Wie bereits erwähnt läuft ein Webserver üblicherweise auf Port 80. Manchmal kann es jedoch erforderlich sein ihn auf einem anderen Port zu betreiben. Dann reicht es nicht mehr aus nur die URL in die Adresszeile des Browsers zu tippen. In diesem Fall muss die entsprechende Portnummer bekannt sein, und mit übermittelt werden. Das geschieht durch anhängen eines Doppelpunktes und der neuen Portnummer an die übliche URL:

`http://www.barmetler.de:`*PORTNUMMER*

3 Das ISO/OSI Referenzmodell

Um das Verständnis für die einzelnen Protokolle in Kapitel 4 und deren Zusammenspiel zu erleichtern, wird zunächst das von der International Standardisation Organisation (ISO) eingeführte OSI (Open System Interconnection) Referenzmodell vorgestellt. Es soll damit ein einheitliches Modell der Kommunikation verteilter Systeme geschaffen werden, welches jedoch noch Freiheiten bei der Implementierung und Realisierung lässt. Das OSI-Referenzmodell dient quasi als Rahmen für die Standardprotokolle.

Application Layer	7	Anwendungsschicht
Presentation Layer	6	Darstellungsschicht
Session Layer	5	Kommunikationsschicht
Transport Layer	4	Transportschicht
Network Layer	3	Vermittlungsschicht
Data Link Layer	2	Sicherungsschicht
Physical Layer	1	Bitübertragungsschicht

Abbildung 4: Das ISO/OSI Referenzmodell

Dieses Protokollmodell gliedert sich in sieben Schichten (vgl. Abbildung 4), wobei jede Schicht ihre spezifischen Aufgaben durch Inanspruchnahme von Dienstleistungen der darunter liegenden Schicht erfüllt. Sie bietet ihrerseits wieder der nächsthöheren Schicht Dienste an. Das OSI Referenzmodell sorgt für eine klare, logische Trennung und Klassifizierung von Kommunikationsfunktionen:

Application Layer (Anwendungsschicht): Diese Schicht stellt die Schnittstelle bereit, die von Applikationen genutzt werden, um Dienste im Netz zu erhalten.

Presentation Layer (Darstellungsschicht): Die Darstellungsschicht wandelt Daten in ein generisches Format um, womit sie im Netzwerk übertragen werden können. Eingehende Daten werden so umgewandelt dass sie von der empfangenden Applikationen genutzt werden können.

Session Layer (Kommunikationsschicht): Die Kommuniaktionsschicht ermöglicht, dass 2 Parteien eine Kommunikation (=Sitzung = Session) über ein Netzwerk führen können.

Transport Layer (Transportschicht): Die Transportschicht ermöglicht die Übertragung der Daten über das Netzwerk.

Network Layer (Vermittlungsschicht): Die Vermittlungsschicht wandelt Netzwerkadressen und Namen in ihre physischen Bedeutungen um. Sie ist bei der Versendung von Nachrichten für deren Adressierung zuständig.

Data Link Layer (Sicherungsschicht): Die Sicherungsschicht sendet spezielle Daten von der Vermittlungsschicht zur Bitübertragungsschicht.

Physical Layer (Bitübertragungsschicht): Die Bitübertragungsschicht wandelt bei ausgehenden Nachrichten Bits in Signale und bei eingehenden Nachrichten Signale in Bits um.

Jeder dieser Schichten sind nun ein oder mehrere Protokolle zugeordnet. Abbildung 5 veranschaulicht dies am Beispiel der TCP/IP Protokoll-Familie. Dabei erkennt man gut, dass hier die Freiheiten des OSI-Schichtenmodells dahingehend genutzt wurden, dass die Layer 1 und 2 zusammengefasst, sowie auf die Implementation der Layer 5 und 6 verzichtet wurde.

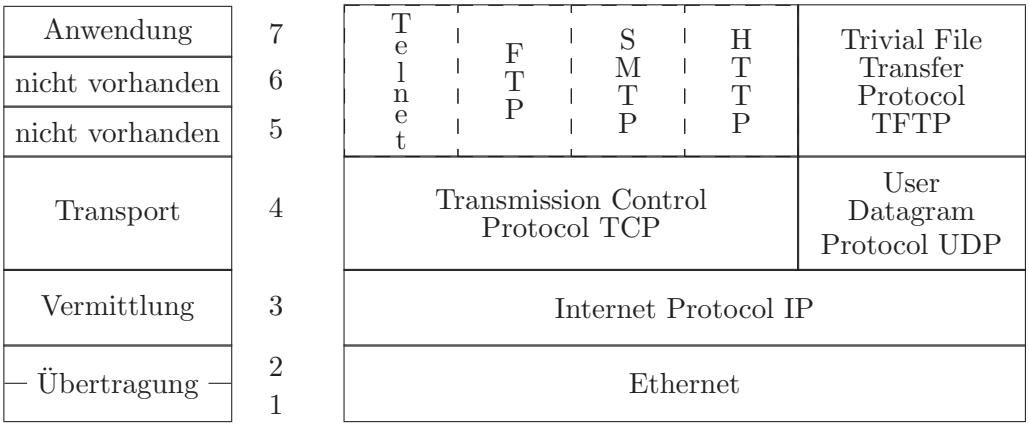


Abbildung 5: TCP/IP Protokoll Schichten

Angenommen eine Anwendung (Schicht 7) möchte Daten über das Internet versenden. Dafür werden die Daten (symbolisiert durch einen Stern: $*$) zunächst in „handliche“ Pakete, so genannte Datagramme, zerlegt. Wie groß diese sein dürfen wird beim Verbindungsaufbau automatisch „ausgehandelt“. Alle Pakete werden einzeln von der darunter liegenden Schicht (Schicht 4, da es Layer 5 und 6 bei der TCP/IP Protokollfamilie nicht gibt) übernommen. Das hier zuständige TCP setzt einige Byte mit Verwaltungsdaten `T` vor jedes Nutzdatenpaket. Dann werden die Datagramme an die nächste Schicht (Layer 3) weitergereicht. Auch das Internet Protocol setzt seinen Header `I` vor jedes Paket und gibt es an die unterste Schicht durch. Ist der sendende Rechner in einem lokalen Netz eingebunden muss auch das Ethernet Protokoll alle Datagramme mit einem Header `E` und einer Checksumme `C` versehen. Diese dient der Korrektur von

Übertragugnsfehlern. Die Datenpakete verlassen nun den Computer des Benutzers und werden durch das lokale Netz bis an ein Internetgateway geleitet. Dort wird der Header des Ethernet Protokolls und die Checksumme wieder entfernt, und die Datagramme über das Internet versandt (siehe Abbildung 6).

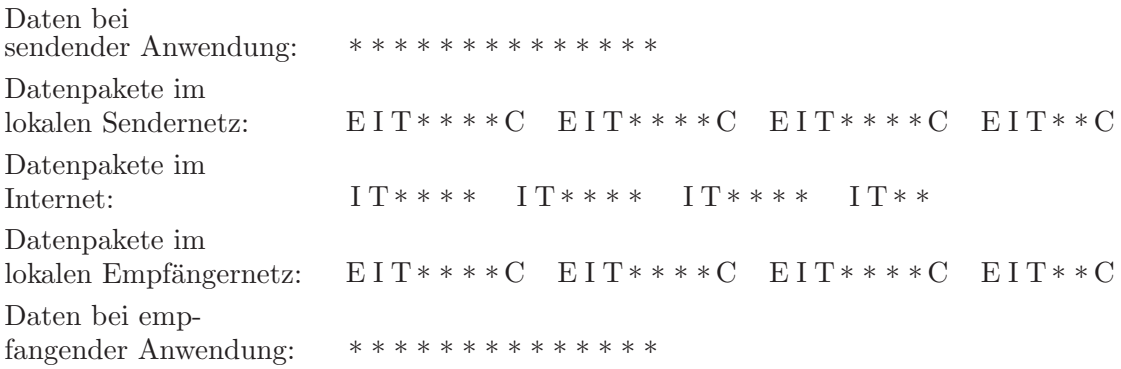


Abbildung 6: Datenpakete mit Protokollheader

Auf der Empfängerseite geschieht der Vorgang in umgekehrter Reihenfolge: Die Pakete kommen an einem Gateway an. Dort erhalten sie einen Ethernet Header und eine Checksumme und werden über das lokale Netz an den Zielrechner geleitet. Der Ethernet-Header und die Checksumme werden wieder entfernt. Durch das Typenfeld (siehe 4.1) wird das auf der nächsthöheren Schicht liegende Protokoll festgestellt. Dieses zeigt zu IP. Das Internet Protocol entfernt nun seinen Header, und weiß aufgrund des Feldes *Protokoll* (siehe 4.2) dass es das Datagramm an TCP weiterreichen muss. TCP setzt nun die einzelnen Datagramme anhand der Sequenznummerierung (siehe 4.3) wieder in der richtigen Reihenfolge zusammen. Anhand der Portnummer (siehe 2.3) kann es erkennen zu welcher Anwendung die Daten gehören, und diese entsprechend nach oben weiterleiten.

4 Protokolle

4.1 Ethernet (Layer 1+2)

Zur Vernetzung zweier Computer wird eine elektrische Verbindung zwischen ihnen benötigt. Ganz schlicht gesprochen muss ein Kabel verlegt und an den Netzwerkkarten der Computer angeschlossen werden. Dafür werden heute in der Regel Ethernet-Steckkarten mit Koaxial- oder Twisted-Pair-Kabel verwendet.

Die Ethernet-Hardware ist in der Lage, in einem Standard festgelegte elektrische Signale zwischen den beiden Ethernetkarten auszutauschen. Die zugehörige Treibersoftware sorgt für korrekten Betrieb. Mehr als einzelne Bits ohne tiefere Bedeutung können die Computer jetzt aber noch nicht austauschen.

Der Ethernet-Header hat folgenden Aufbau:

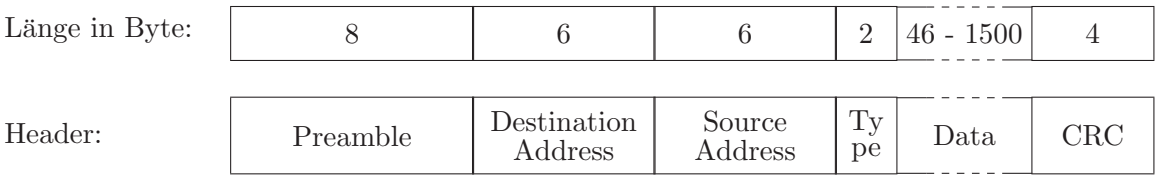


Abbildung 7: Ethernet Header

Erläuterung zu den Feldern des Ethernet Protokolls (siehe Abbildung 7):

Preamble Dient der Bitsynchronisation zwischen Sender und Empfänger. Hat eine festgelegte Bitfolge von alternierenden Nullen und Einsen.

Destination Address Die ersten drei Oktette geben den Hersteller der Ethernetkarte an, die nächsten 3 Oktette sind fortlaufend durchnummeriert. Jede Adresse ist weltweit eindeutig.

Source Address Siehe Destination Address.

Type Verweis auf das in Schicht 3 verwendete Protokoll.

Data Zwischen 46 und 1500 Oktette Nutzinformationen.

CRC Cyclic-Redundancy-Check - Prüfsumme, die nach dem CRC-Verfahren generiert wurde.

4.2 Internet Protocol IP (Layer 3)

IP deckt Schicht 3 des OSI-Referenzmodells ab, und übernimmt Aufgaben der Adressierung und des Routings - also die Übermittlung von Datenpaketen von einem Sender über mehrere Netze hinweg zu einem Empfänger. Die Datenübertragung ist damit schon etwas komfortabler, da jetzt ganze Pakete ausgetauscht werden. Allerdings fehlt noch jeglicher Mechanismus um feststellen zu können, ob alle Pakete angekommen sind (darum nennt man es auch ein verbindungsloses Protokoll) und welche Reihenfolge sie haben müssen.

Um diese Aufgabe zu erfüllen wird jedes Datenpaket mit einem mindestens 20 Oktett langen IP-Header versehen (siehe Abbildung 8).

Erläuterung zu den Feldern des IP-Headers in Abbildung 8:

Vers Version des verwendeten Internet Protokolls. Zur Zeit ist noch Version 4 Standard.

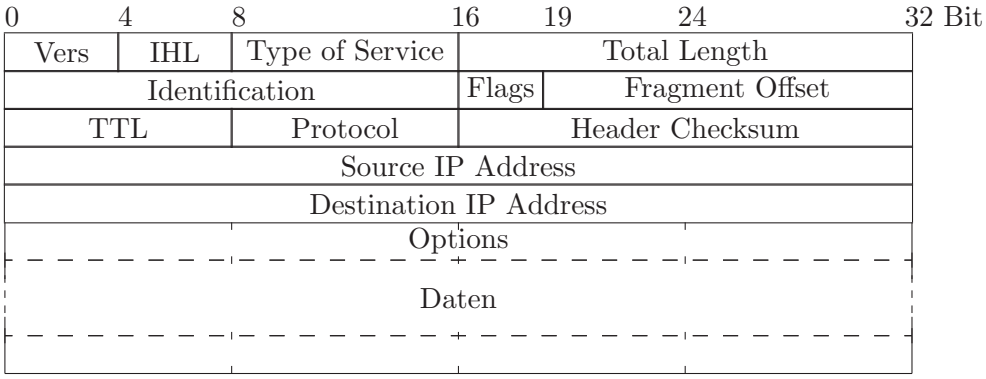


Abbildung 8: IP Header

IHL Internet Header Length - Länge des Headers in 32-Bit Worten (mind. 5).

Type of Service Hostrechner gibt Service an (verschiedene Kombinationen aus Zuverlässigkeit und Schnelligkeit sind möglich). Es wird meistens nur der Wert 0 (Routine) verwendet. Wie mit dieser Angabe zu verfahren ist wird jedoch nicht durch das IP-Protokoll festgelegt.

Total Length Länge des Datagramms (also Header plus Daten).

Identification Eindeutige Nummer um die Fragmente eines Datagramms wieder zusammensetzen zu können.

Flags Anweisungen an das Gateway ob dieses Datagramm fragmentiert werden darf oder nicht, bzw. ob dies ein Fragment eines Datagramms ist oder nicht.

Fragment Offset Dieses Feld wird bei fragmentierten Datagrammen genutzt. Der Wert gibt an, wie viele 64 Bit Blöcke (ohne die Header Oktette) bereits in vorangegangenen Fragmenten enthalten waren.

TTL Time to live - Zähler, der maximal auf 255 steht. Jedes IP-Gateway, schätzt die Laufzeit des Paketes vom letzten Gateway zu sich ab, und subtrahiert diese Zeit (mindestens jedoch 1) vom Zähler. Ist er Null wird das Paket verworfen.

Protocol Sagt der Vermittlungsschicht zu welchem übergeordneten Transportprotokoll das Paket gehört. Z. B.:
06 Transmission Control Protocol TCP
17 User Datagram Protocol UDP

Header Checksum Prüfsumme über die Felder des IP-Headers (nicht über die Daten).

Source IP Address 32-Bit Netz- und Teilnehmeradresse des Senders

Destination IP Address 32-Bit Netz- und Teilnehmeradresse des Empfängers

Options Error-, Debug und sonstige Informationen.

4.3 Transmission Control Protocol TCP (Layer 4)

Über die IP-Schicht wird nun die Transmission Control Protocol-Schicht TCP gelegt. Sie ist verantwortlich für den Aufbau von logischen Verbindungen zwischen zwei Kommunikationspartnern. Es handelt sich dabei um ein verbindungsorientiertes „end-to-end“ Protokoll. Die folgerichtige Übertragung wird mit Hilfe von Sequenznummern garantiert. Da eine Rückmeldung des Empfängers an den Sender über empfangene Pakete zwingend erfolgt, ist auch die Übertragung der Daten gesichert.

Jedes Datenpaket (Datagramm) wird mit einem mindestens 20 Oktett langem TCP-Header versehen (siehe Abbildung 9).

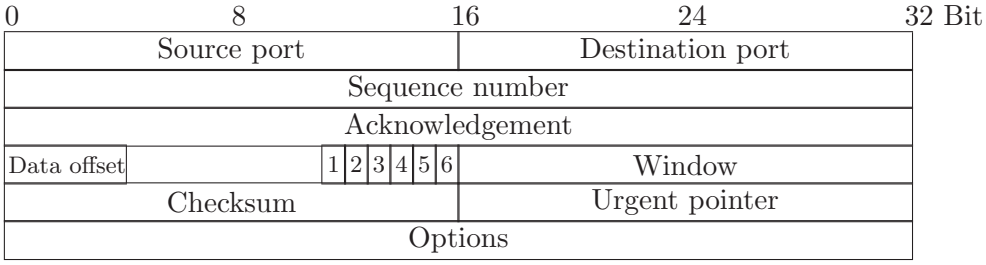


Abbildung 9: TCP Header

Erläuterung zu den Feldern des TCP-Headers in Abbildung 9:

Source port Code des Dienstes, welcher die Daten versendet (vgl. auch 2.3).

Destination port Code des Dienstes, der die Daten empfängt (vgl. auch 2.3).

Sequence number Zähler, der mit jedem gesendeten Datenpaket erhöht wird.

Acknowledgement Rückmeldung des Datenempfängers an den Sender über die bereits fehlerfrei empfangenen Pakete.

Data offset Länge des TCP Headers in 32 Bit Worten.

Flags (1) *URG*: Urgent Pointer ist gesetzt
(2) *ACK*: Acknowledgement-Feld hat gültigen Inahlt
(3) *PSH*: Daten dieses Paketes sofort an Anwendung übergeben
(4) *RST*: Rücksetzen einer Verbindung
(5) *SYN*: Syhnchronisierung der Sequenznummern
(6) *FIN*: Auflösen einer Verbindung (der Sender hat keine Daten mehr)

Window Legt fest, wie viele unquittierte Datenpakete ausstehen dürfen.

Checksum Prüfsumme über den TCP-Header und über sämtliche Nutzdaten.

Urgent pointer Zeigt wichtige Daten im Datenfeld an. Z. B. Interrupts.

Options Error-, Debug- und sonstige Informationen.

4.4 User Datagram Protocol UDP (Layer 4)

UDP ist ein verbindungsloses Protokoll. Vor der Kommunikation muss keine Verbindung aufgebaut werden, und es gibt keine Garantie für den Empfang und die richtige Reihenfolge der Pakete. UDP dient vor allem für Anwendungen, bei denen eine Bestätigung über den erfolgreichen Versand der Pakete und das Einhalten der Reihenfolge nicht wichtig ist, dafür aber auf die Geschwindigkeit Wert gelegt wird. Der UDP-Header (siehe Abbildung 10) liefert nur eine Portnummernzuordnung der Datagramme.

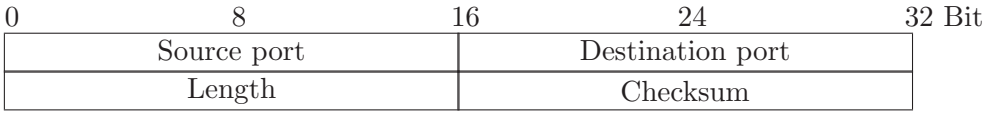


Abbildung 10: UDP Header

Erläuterung zu den Feldern des UDP-Headers in Abbildung 10:

Source port Code des Dienstes, welcher die Daten versendet. Allerdings stimmt die Portnummer eines Dienstes in TCP nicht mit der Portnummer in UDP überein.

Destination port siehe Source port.

Length Länge des Datagramms inkl. UDP-Header.

Checksum Prüfsumme über UDP-Header.

4.5 Simple Mail Transfer Protocol SMTP (Layer 7)

Für den Dienst E-Mail setzt auf die TCP-Schicht ein Protokoll zum Austausch elektronischer Briefe auf: das Simple Mail Transfer Protocol (SMTP).

Der Anwender bedient aber lediglich eine Software, die E-Mails mit Hilfe von SMTP austauscht, z.B. einen der E-Mail-Clients Microsoft Outlook, Netscape Mail, Eudora oder Pegasus.

Wird in einem E-Mail-Client eine Nachricht verfasst, so gibt sie der Client nach unten an die SMTP-Schicht weiter, diese an die TCP-Schicht, diese an die IP-Schicht, diese an die Ethernet-Schicht und erst dort geschieht die wirkliche physikalische Übertragung. Auf der Empfängerseite läuft dieser Vorgang in umgekehrter Reihenfolge ab.

4.6 Hypertext Transfer Protocol HTTP (Layer 7)

Für den Dienst WWW (World Wide Web) wird das Hypertext Transfer Protocol (HTTP) eingesetzt, darauf basieren die WWW-Browser wie z.B. Microsoft

Internet Explorer, Netscape Navigator und Opera.

Auch hier gibt der Browser-Client die Daten an die darunterliegende HTTP-Schicht weiter. Von dort geht es den bereits bekannten Weg über TCP, IP und Ethernet.

4.7 File Transfer Protocol FTP (Layer 7)

Mit dem Dienst FTP (File Transfer Protocol) werden Dateien über das Internet übertragen. Das zugehörige Protokoll heißt ebenfalls File Transfer Protocol (FTP).

Ebenso wie bei den bereits bekannten Diensten E-Mail und WWW gibt der FTP-Client die zu versendenden Daten an die darunterliegende FTP-Schicht ab. Von dort gehen sie den Pfad aller zu übertragender Daten: FTP → TCP → IP → Ethernet.

4.8 Über den Tellerrand geschaut

Für die Client/Server-Kommunikation oder die Steuerung (etwa das Anmelden im Netzwerk) kommt in IBM- und Microsoft-LANs SMB (Server Message Block) zum Einsatz. Dieses Protokoll definiert eine Vielzahl von Befehlen, die dem Informationsaustausch zwischen zwei Stationen im Netzwerk dienen. Der Redirector, der für die Entscheidung zuständig ist, ob der Zugriff lokal oder über das Netzwerk erfolgen muss, fasst NCB-Anforderungen (Network Control Block) in einer SMB-Struktur zusammen, bevor er diese über das Netzwerkprotokoll verschickt.

Für die Kommunikation zwischen zwei Stationen ist es dabei wichtig, dass beide Stationen dieselbe Sprache sprechen - also das gleiche Protokoll verwenden. Dies sind primär IP (Internet Protocol), NetBEUI (NetBIOS Extended User Interface), NWLink (IPX/SPX), TCP (Transmission Control Protocol) und UDP (User Datagram Protocol).

Das Internet Protocol (IP) hat sich zum Standard-Protokoll im LAN- und WAN-Bereich entwickelt hat. Nachteilig gegenüber NetBEUI oder IPX/SPX ist der hohe Planungsaufwand, der für die Vergabe der Adressen und das Einrichten der Subnetze notwendig ist. Die Garantie für den Versand der Daten überlässt IP den Protokollen höherer Ebenen, etwa TCP.

Um innerhalb eines TCP/IP-Netzes eine IP-Adresse in die MAC-Adresse (Hardware-Adresse) der Netzwerkkarte aufzulösen, benutzt man das Address Resolution Protocol (ARP). Ein ARP-Request geht über die Broadcast-Adressierung (FF FF FF FF FF FF) an alle Stationen im Netz. Erkennt eine Station die dabei angegebene IP-Adresse als ihre eigene, so antwortet sie mit einem ARP-Response, in dem ihre MAC-Adresse enthalten ist. Die Größe eines ARP-Pakets beträgt 28 Byte.

Für den Austausch von Fehlermeldungen und Statusinformationen inner-

halb eines TCP/IP-Netzes ist das Internet Control Message Protocol ICMP zuständig. Die bekannteste Anwendung ist wohl das auf ICMP basierende *ping*, mit dem sich die Verbindung zu einer entfernten Station testen lässt.

Im Unterschied zu TCP/IP entwickelte IBM NetBEUI 1985 vor allem für kleinere Netzwerke mit bis zu 200 Stationen. Der Nachteil dieses Protokolls ist die geringe Anzahl der adressierbaren Stationen und seine Unfähigkeit, über Router zu kommunizieren. Es ist aber sehr schnell, besitzt eine gute Fehlerkorrektur und benötigt praktisch keinen Konfigurationsaufwand auf Servern und Clients. Die Identifikation der Stationen im Netzwerk erfolgt über einen eindeutigen Computernamen (NetBIOS-Namen). In der Version 3.0 unterstützt NetBIOS maximal 254 Stationen. Microsoft bezeichnet NetBEUI auch als NBF (NetBIOS Frame), IBM dagegen bleibt oft bei NetBIOS. Das Format und die Header-Größe ist bei NetBEUI je nach Art der Kommunikation unterschiedlich.

Als weitere Protokolle neben der TCP/IP-Familie und NetBEUI findet man vor allem in NetWare-Netzen Internet Packet Exchange (IPX) beziehungsweise Sequence Packet Exchange (SPX). Microsoft nennt den entsprechenden Protokollstack NWLink. IPX und SPX stellen unterschiedliche Kommunikationsprotokolle dar, sie bauen nicht wie TCP und IP aufeinander auf. IPX ist verbindungslos, und es gibt keine Garantie für den Versand und die Reihenfolge der Pakete. Die Garantie für den Versand der Daten wird den Protokollen höherer Ebene überlassen. Der Header des IPX-Paketes hat eine Größe von 30 Byte. SPX dagegen ist verbindungsorientiert. Vor der Übertragung der Pakete baut es erst eine Kommunikationsstrecke auf und garantiert die Reihenfolge und die Übertragung der Pakete. Der SPX-Header enthält alle Felder des IPX-Header und verfügt zusätzlich über 12 Byte (beziehungsweise in der neusten Version 14 Byte) für Sequence- und Acknowledgement-Informationen.

5 Erklärung verwendeter Akronyme

ASP Active Server Pages - Programmiersprache von Microsoft um dynamisch Webseiten zu erstellen.

CGI Common Gateway Interface - Standard zur Ausführung extern laufender Programme auf einem WWW-Server.

CRC Cyclic-Redundancy-Check - Verfahren um eine Prüfsumme zu generieren.

DENIC Deutsches Network Information Center mit Sitz in Karlsruhe. Die Mitglieder der DENIC eG sind Internet Service Provider, kurz ISPs.

DNS Domain Name System - hierarchisch aufgebautes System für die Vergabe von Namen für an das Internet angeschlossene Rechner (Beispiel: der Name „www.javascript.barmetler.de“ enthält die Bezeichnung des Dienstes „www“, die Toplevel-Domain „de“ und die Secondary Domain „javascript.barmetler“).

FTP File Transfer Protocol - Standard zur Datenübertragung via Internet auf der Grundlage von TCP/IP.

HTTP Hypertext Transfer Protocol - Standard für die elektronische Interaktion bei der Übertragung von Web-Dokumenten im Internet.

IANA Internet Assigned Numbers Authority - Sie kontrolliert die Vergabe öffentlicher IP-Adressen um das mehrfache Auftreten derselben Adresse zu verhindern.

IP-Adresse Jeder angeschlossene Rechner ist im Internet über eine numerische Adresse identifizierbar. Eine IP-Adresse (Internet-Protokoll-Adresse) besteht aus vier durch Punkt getrennten Zahlen, die jeweils einen Wert zwischen 0 und 255 annehmen können (z. B. 217.188.092.178). IP-Adressen können fest oder dynamisch (d. h. bei jeder Einwahl neu) vergeben werden.

IPv4 Version 4 des Internet Protocol

IPv6 Version 6 des Internet Protocol

ISO International Standardisation Organisation - Gremium um weltweit einheitliche Standards zu schaffen, so dass technische Geräte kompatibel arbeiten.

ISP Internet Service Provider - Unternehmen, die ihren Kunden Dienstleistungen rund um das Internet bereitstellen, wie: Speicherplatz für Webseiten, Datenbankdienste, Mailserver, ...

JSP Java Server Pages - Technik zum serverseitigen dynamischen Erstellen von Webseiten.

OSI Open System Interconnection - Einheitliches Modell der Kommunikation verteilter Systeme.

PHP PHP Hypertext Preprocessor - Skriptsprache um serverseitig dynamisch Webseiten zu erstellen.

SFD Start of Frame Delimiter - Festgelegte Bitfolge (10101011) im Ethernet Header.

SMTP Simple Mail Transfer Protocol - Standardprotokoll im Internet zur Übertragung von elektronischer Post zwischen Rechnern.

SSH Secure SHell - SSH ist ein Protokoll zum sicheren Einloggen und zum sicheren Betreiben anderer Netzwerkdienste (Dateiaustausch, Kommandoausführung(remote command), ...) über ein unsicheres Netz.

TCP/IP Transmission Control Protocol/Internet Protocol - die Protokollreihe, mit deren Hilfe die Datenübertagung im Internet funktioniert.

UDP User Datagram Protocol - Verbindungsloses Protokoll zur Datenübertragung.

URL Uniform Resource Locator - die Adresse, unter der ein Dokument im World Wide Web zu finden ist. Die allgemeine Syntax lautet: Protokoll://Dienst.Rechnername:Port/Pfad/Datei.

World Wide Web Der multimediale und zweitbeliebteste Dienst (nach E-Mail) des Internet.

www Abkürzung; s. World Wide Web

Literatur

[1] Prof. Dr.-Ing. J. Eberspächer, Praktikum Kommunikationsnetze 2, Lehrstuhl für Kommunikationsnetze der TU München, 1996

[2] Elektronische Ausgaben des Computermagazins c't, Verlag Heinz Heise, 1997 - 2003

[3] Minoli Emma und Minoli Daniel, Delivering Voice over IP Networks, John Wiley & Sons, 1998

[4] Domain-Informationen, Webseite von 1&1 Webhosting, http://hosting.1und1.com

[5] Networkessentials, http://www.networkingessentials.de

[6] Internetmanual, http://www.internet-manual.de

Index