TCP/IP - Lehrgang

Dr. Rudolf Strub

[1]

[1] Es ist eine etwas erweiterte Übersetzung des RFC 1180 von Dr. R. Strub, e-mail: strub@ee.ethz.ch. Das RFC ist als Lehrgang über die Gesamtheit der TCP/IP Protokolle gedacht, mit besonderem Augenmerk auf die Stationen die ein IP-Datenpaket auf seinem Weg vom Quell- zum Ziel-Host über einen Router durchläuft. Es ist keine Spezifikation des Internet-Standards.

Inhaltsverzeichnis

1 Einführung

Dieser Lehrgang enthält lediglich eine Übersicht über die Schwerpunkte von TCP/IP, und soll als Leitfaden durch die TCP/IP-Technologie dienen. Er enthält nichts über historische Entwicklung und Konsolidierung von TCP/IP, die kaufmännischen Aspekte und ihre Zukunft im Vergleich zu ISO - OSI. In der Tat wird auch ein grosser Teil der technischen Informationen weggelassen, insbesondere was die feineren Details betrifft. Was verbleibt ist ist ein Minimum an Informationen, deren Verständnis für das professionelle Arbeiten in einer TCP/IP-Umgebung unumgänglich ist. Dies schliesst insbesondere die Aufgaben des Systemadministrators, des Systemprogrammierers und von Netzwerk-Managern ein.

Dieser Lehrgang benutzt Beispiele aus der UNIX - TCP/IP - Umgebung, die Hauptpunkte treffen jedoch auf alle TCP/IP-Implementierungen zu.

Es sei nochmal darauf hingewiesen: diese Schrift dient der Erläuterung der Zusammenhänge und nicht der Definition des TCP/IP-Standards. Sollten Fragen über die exakten Spezifikationen auftreten, sind die aktuellen Standards in dem entsprechenden RFC nachzusehen.

Der nächste Abschnitt gibt eine Übersicht über TCP/IP, gefolgt von detaillierteren Beschreibungen der einzelnen Komponenten in den anschliessenden Abschnitten.

2 TCP/IP - Übersicht

Unter der allgemeinen Bezeichnung 'TCP/IP' versteht man im allgemeinen alles und jedes was mit den spezifischen Protokollen TCP und IP zusammenhängt. TCP/IP kann demnach Anwendungen, weitere Protokolle und sogar des Netzwerk- oder Übertragungsmedium einschlieen. Ein Beispiel für weitere Protokolle sind UDP, ARP und ICMP. Beispiele für Anwendungen sind TELNET, FTP und rcp. Treffender wäre der Ausdruck Internet-Technologie. Demnach ist ein Netzwerk, welches auf der Basis der Internet-Technologie arbeitet ein Internet.

2.1 Die grundlegende Struktur

Um diese Technologie zu verstehen, müssen wir zuerst die folgende logische Struktur verstehen (Fig.1):

Dies ist die logische Struktur des mehrschichtigen Protokolls in einem Computer im Internet. Jeder Computer der über das Internet kommunizieren kann, verfügt über diese logische Struktur die sein Verhalten auf dem Netz bestimmt. Die Rechtecke stellen Prozesse dar, denen Daten auf dem Weg durch den Computer

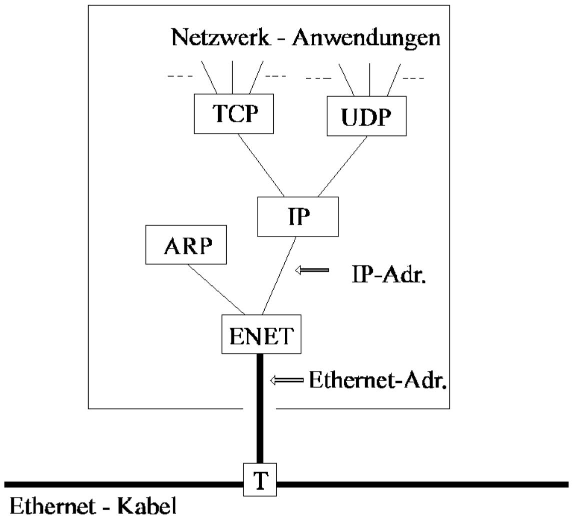

Fig.1: Basis-Knoten eines TCP/IP-Netzes

unterworfen werden, und die Linien welche die Rechtecke verbinden symbolisieren die Wege welche die Daten nehmen. Die waagerechte Linie am unteren Rand stellt das Ethernetkabel dar, 'T' ist der Transceiver, er dient der elektronischen Anpassung. Das Verständnis dieser logischen Struktur ist von grundlegender Bedeutung für das Verständnis der Internet Technologie. Wir werden uns daher in diesem Lehrgang immer wieder darauf beziehen.

2.2 Terminologie

Der Bezeichnung einer Dateneinheit, welche sich durch das Internet bewegt, hängt davon ab, an welcher Stelle des Protokollstapels sie sich gerade befindet. Kurz zusammengefasst: befinden sich die Daten auf dem Ethernet heissen sie ETHERNET FRAME, befinden sie sich zwischen dem Ethernet-Treiber und dem IP-Modul heissen sie IP-PAKET, befinden sie sich zwischen dem IP-Modul und dem UDP-Modul heissen sie UDP-DATAGRAM, sind sie zwischen dem IP-Modul und dem TCP-Modul heissen sie TCP- SEGMENT (oder allgemeiner transport message) und wenn sie sich in einer Anwendung befinden heissen sie application message.

Diese Definitionen sind leider nicht perfekt. Sie werden von einer Publikation zur nächsten anders gehandhabt. Spezifischere Definitionen finden sich in RFC 1122 section 1.3.3.

Ein Treiber ist ein Programm, welches direkt mit der Interface-Hardware kommuniziert.. Ein Modul ist ein Programm welches mit einem Treiber oder mit einem oder mehreren Modulen kommunizieren kann. Die Begriffe Treiber, Modul, Ethernet-Frame, IP-Paket, UDP-Datagramm, TCP-Message und Application-Message werden in dieser Schrift im diesem Sinne verwendet.

2.3 Der Datenfluss

Lassen Sie uns nun den Datenfluss durch den Protokollstapel in Fig.1. verfolgen. Von Anwendungen, welche TCP (Transmission Control Protocol) verwenden, gelangen die Daten in den TCP-Modul. Bei Anwendungen, die UDP (User Datagram Protocol) verwenden, gelangen die Daten von der Anwendung zum UDP-Modul. Eine typische Anwendung, die TCP benutzt, ist FTP (File Transfer Protocol). Sein Protokollstapel lautet beispielsweise FTP/TCP/IP/ENET. Eine UDP-Anwendung ist SNMP (Simple Network Management Protocol). Ihr Protokollstapel lautet daher SNMP/UDP/IP/ENET.

Das TCP-Modul, UDP-Modul und der Ethernet-Treiber sind n-zu-1 Multiplexer. Sie sind in der Lage von mehreren Eingängen einen auf den (einzigen) Ausgang zu schalten. Sie sind auch 1-zu-n Demultiplexer. Als Demultiplexer schalten Sie einen Eingang auf einen von mehrere Ausgängen entsprechend dem Typfeld im Protokollkopf der ankommenden Daten.

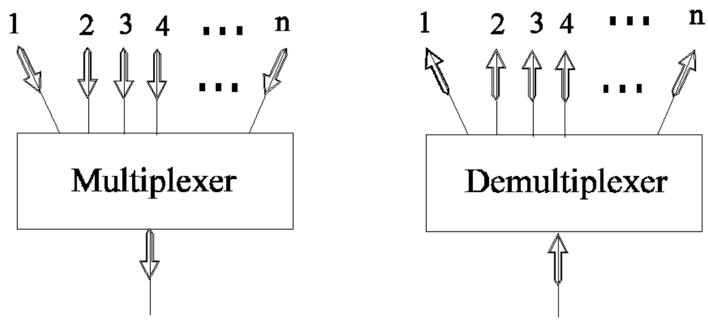

Fig.2: n-zu-1 Multiplexer und 1-zu-n Demultiplexer

Erreicht nun ein Ethernet-Frame aus dem Netz kommend den Ethernet-Treiber, kann das Datenpaket entweder zum ARP- (Address Resolution Protokoll) Modul oder zum IP- (Internet Protocol) Modul weitergeleitet werden. Der Wert der im Typfeld des Ethernet-Frame steht entscheidet, ob die Weiterleitung zum ARP- oder IP-Modul erfolgt. Erreicht ein IP-Paket das IP-Modul, kann es zum TCP- oder zum UDP-Modul weitergeleitet werden. Dies wird bestimmt durch den Inhalt des Protokoll-Feldes im IP-Header. Wenn ein UDP-Datagramm das UDP-Modul erreicht, entscheidet dieses je nach Inhalt des Port-Feldes in der Application-Message zu welcher Anwendung das Datenpaket weiterzuleiten ist. Wenn eine TCP- Message den TCP-Modul erreicht, entscheidet dieser ebenfalls anhand des Inhalts des Portfeldes der TCP-Message zu welcher Anwendung das Datenpaket weiterzuleiten ist.

Der 'abwärts' führende Multiplexerpfad ist wesentlich einfacher in der Durchführung, da von jedem Ausgangspunkt jeweils nur ein Weg zum Ausgang führt. Jeder Modul fügt daher lediglich an der richtigen Stelle seine Header-Information hinzu, damit das Paket im Zielcomputer korrekt demultiplext werden kann.

Unabhängig ob es sich um UDP- oder TCP-Anwendungen handelt, die Datenströme laufen stets über das IP-Modul und von dort zum darunterliegenden Netzwerk-Interfacetreiber.

Obwohl die Internet-Technologie eine Vielzahl von Netzmedien unterstützt, wird in diesem Lehrgang stets von ei-nem Ethernet ausgegangen, da es das am häufigtsen verwendete physikalische Netz ist, welches unter IP benutzt wird. Der Computer in Fig.1 besitzt nur einen Ethernet-Anschlu. Die 6-Byte Ethernet-Adresse ist für jedes Interface (Ethernetkarte) einmalig, da sie vom Hersteller auf der Karte eingeschrieben wurde.

Der Computer hat unabhängig davon eine 4-Byte IP-Adresse. Diese Adresse ist im IP-Modul gespeichert. Diese Adresse mu ebenfalls einmalig sein auf dem Internet. Sie wird daher von der Netzadministration vergeben. Ein funktionierender Computer mu immer seine eigene IP- und Ethernet-Adresse kennen.

2.4 Zwei Netzwerk-Anschlüsse

Ist ein Computer mit zwei verschiedenen Ethernets verbunden, ergibt sich die eine logische Struktur nach Fig.3.

Dieser Computer hat sowohl 2 Ethernet-Adressen als auch zwei IP-Adressen. Es ist zu sehen, da er zwei physikal-ische Netzwerk-Interfaces hat und das IP-Modul ist nun ein n-zu-m Multiplexer und m-zu-n Demultiplexer.

Das IP-Modul multiplext nun sowohl die ankommenden wie die abgehenden Datenströme, d.h. unser IP- Modul ist wesentlich komplexer geworden als das in unserem ersten Beispiel (Fig.1). Daten können von einem Netzwerk empfangen und in ein anderes gesendet werden und umgekehrt. Der Prozess des Weiterbeförderns von Datenpaketen von einem Netz zu einem Anderen heisst 'forwarding'. Ein Computer der die Aufgabe des forwarding von IP-Paketen hat, heit IP-Router.

Wie man aus der Fig.5 ersehen kann, berührt das Weiterbefördern von IP-Paketen niemals die TCP- und UDP-Module des Routers. Einige Implementationen von IP-Routern weisen diese Module daher gar nicht auf.

2.5 IP bildet ein zusammenhängendes logisches Netzwerk

Der Erfolg des Internet ist im wesentlichen auf die Wirkungsweise des IP-Moduls zurückzuführen. Jedes Modul und jeder Treiber hängt seinen Kopf an das Datenpaket, wenn dieses auf seinem Weg abwärts durch den Protokollstapel vorbeikommt. Steigt ein Datenpaket durch den Protokollstapel aufwärts in Richtung

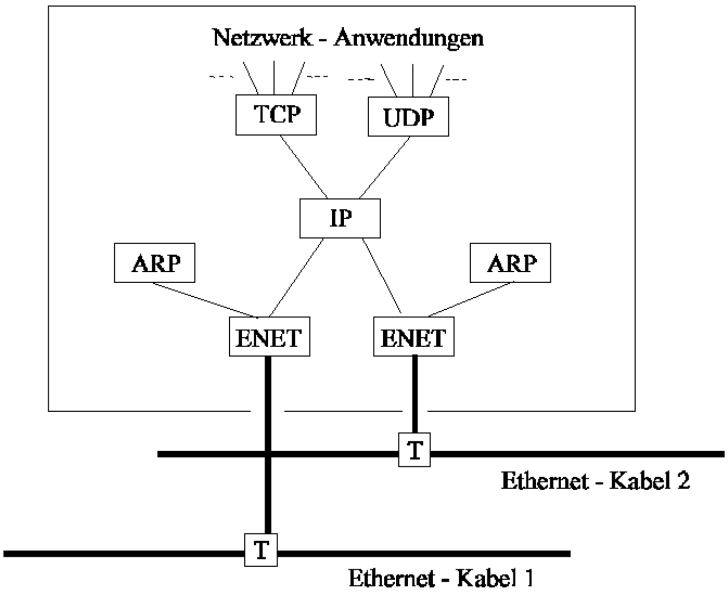

Fig.3: TCP-IP Netzwerk-Knoten auf 2 Ethernetzen

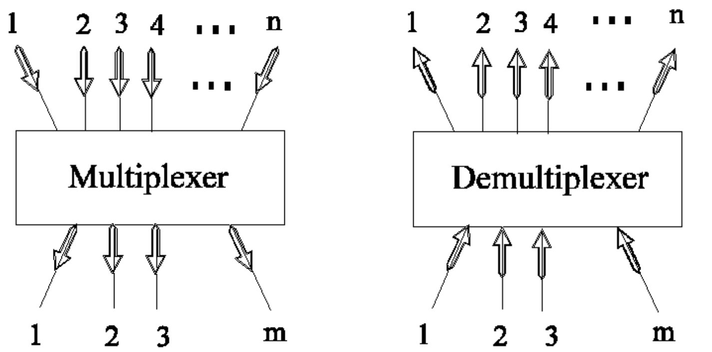

Fig.4: n-zu-m Multiplexer und m-zu-n Demultiplexer

einer Anwendung, entnimmt jedes Modul bzw. Treiber an dem es vorbeikommt die zuständigen Kopfinformationen. Der IP-Header enthält die IP-Adresse, welche aus mehreren physikalischen Netzwerken ein einziges logisches Netz bildet. Die Verbindung von mehreren physikalischen Netzen zu einem logischen Netz, ist die Ursache für die Bezeichnung 'Internet'. Eine Zusammenfassung mehrerer physikalischer Netze zu einem logischen Netz, das einen Bereich in dem sich IP-Pakete bewegen können bildet, heisst ein INTERNET.

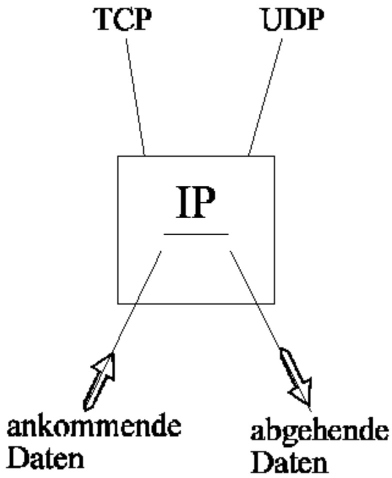

Fig.5: Beispiel für IP-Weiterbeförderung (forwarding)

2.6 Unabhängigkeit vom physikalischen Netzwerk

IP verbirgt die darunterliegende Netzwerk-Hardware gegenüber der Netzanwendung. Wenn sie ein neues physikalisches Netzwerk erfinden, können sie dieses an das Internet anschliessen, indem sie einen neuen Treiber schreiben, der das Internet mit der darunter liegenden Hardware verbindet. So bleiben die Anwendungen intakt und unberührt von Änderungen in der Hardwaretechnologie.

2.7 Interoperabilität

Wenn zwei Computer über das Internet miteinander kommunizieren können, besitzen beide die Eigenschaft der Interoperabilität (interoperability). Die meisten der heute auf dem Markt befindlichen Computer haben diese Eigenschaft. Sollten sie beim Kauf ein Exemplar erwischen, das dazu nicht in der Lage ist, und sich auch nicht nachrüsten lässt, haben sie ein äusserst seltenes Exemplar entdeckt.

2.8 Was kommt nach der Übersicht?

Vor diesem Hintergrund wollen wir im weiteren die folgenden Fragen beantworten:

- Wie findet man die für das Absenden eines IP-Paketes notwendige Ethernet-Adresse?

- Wie findet IP heraus, welches von mehreren darunterliegenden Netzen für das Absenden eines IP- Paketes gewählt werden muss?
- Wie erreicht ein Client auf dem einen Computer, den Server auf einem anderen?
- Warum gibt es sowohl TCP als auch UDP, warum reicht nicht eins von beiden?
- Welche Netzwerk-Anwendungen gibt es?

Diese Punkte werden nach einem Abstecher ins Ethernet erklärt .

3 Ethernet

Dieser Abschnitt gibt einen kurzen Überblick über die Ethernet-Technologie. Ein Ethernet Frame (d.h. eine Dateneinheit auf dem Ethernet) enthält eine Zieladresse, eine Absenderadresse ein Typfeld und die zu transportierenden Daten. Eine Ethernetadresse besteht aus 6-Byte. Jedes Gerät hat seine eigene Ethernetadresse und liest nur Ethernet-Frames mit dieser Adresse. Alle Geräte "hören" aber auch auf Ethernet Farmes mit der 'Wildcard'-Adresse FF-FF-FF-FF-FF-FF" (in hexadezimal), die 'Broadcast'- Adresse heisst und da sie von allen Geräten im Netz gelesen wird.

Das Ethernet benutzt CSMA/CD (Carrier Sense and Muliple Access with Collision Detection). CSMA/CD bedeutet, da alle Geräte über ein eiziges Medium miteinander kommunizieren, auf dem natürlich immer nur ein Gerät aufs mal senden darf, aber alle ständig "hören". Wenn zwei Geräte (nachdem sie geprüft haben, da die Leitung frei ist) gleichzeitig versuchen zu senden, wird die Sende-Kollosion von beiden Geräten entdeckt. Der Sendevorgang wird von beiden abgebrochen und nach einer kurzen und zufälligen Zeitspanne einer neuer Versuch unternommen.

3.1 Eine menschliche Analogie

Eine schöne Analogie zum Ethernet ist eine Gruppe von Leuten, die sich in einem vollständig dunklen Raum unterhalten. In dieser Analogie stellen die Schallwellen in der Luft das physikalische Netzwerk dar, statt der elektrischen Signale auf Koaxkabeln.

Jeder kann hören, wenn ein anderer spricht (Carrier Sense). Jedermann in dem Raum hat dieselben Fähigkeiten der Unterhaltung (Multiple Access), aber keiner hält lange Reden, weil alle höflich sind. Ist eine Person unhöflich wird sie aufgefordert den Raum zu verlassen (d.h. sie wird vom Netz abgeschaltet). Niemand spricht solange ein anderer am Sprechen ist. Wenn jedoch zwei Leute gleichzeitig mit sprechen beginnen, bemerken es beide, weil jeder etwas hört, was er selbst nicht gesagt hat (Collision Detection). Beide werden in diesem Fall einen Moment warten, bevor einer erneut zu sprechen beginnt. Der Andere hört dies und wartet bis der Erste zu Ende gesprochen hat, bevor er einen neuen Versuch unternimmt. Um Verwechslungen zu vermeiden, hat jede Person ihren eigenen Namen (eigene Ethernetadresse). Jedesmal wenn jemand spricht, beginnt er mit dem Namen der Person für die seine Rede bestimmt ist, und sagt seinen eigenen Namen (Ethernet Ziel- und Absenderadresse). Z. B. 'Hallo Jane, hier ist Tarzan, ..blah blah blah '. Wenn der Sprecher zu allen sprechen möchte, kann er sagen 'an alle' (broadcast address). Z.B. 'Hallo an alle, hier ist Tarzan, ...blah,blah, blah...'.

4 ARP

Wie wird nun die Ethernetadresse bestimmt beim Absenden eines IP-Paketes? Für die Übersetzung der IP-Adresse in die Ethernetadresse wird das ARP (Address Resolution Protokoll) benutzt. Diese Übersetzung ist nur für abgehende IP-Pakete notwendig, wenn der IP-Header und der Ethernet-Header gebildet werden.

4.1 ARP-Tabelle für die Adressübersetzung

Die Übersetzung wird mit Hilfe einer einfachen Tabelle ausgeführt. Die Tabelle heisst ARP-Tabelle, ist im Speicher abgelegt, und enthält für beide Adressen jeweils eine Spalte. Bei der Übersetzung einer IP- Adresse in eine Ethernetadresse wird in der Tabelle nach der IP-Adresse gesucht, die zugehörige Ethernetadresse steht dann in derselben Zeile dahinter.

Die folgende Tabelle zeigt eine vereinfachte ARP-Tabelle

IP-Adresse	Ethernetadresse
141.20.30.1	08-00-39-00-2F-C3
141.20.30.3	08-00-5A-21-A7-22
141.20.30.4	08-00-10-99-AC-54

Tab.1: Beispiel einer ARP-Tabelle

Einer allgemein akzeptierten Konvention folgend, werden die 4-Byte langen IP-Adressen in dezimaler Schreibweise getrennt durch einen Punkt angegeben. Die Ethernetadressen werden als 6-Byte-Adresse in hexadezimaler Schreibweise angegeben. Die Zahlen werden durch einen Gedankenstrich oder durch Doppelpunkt getrennt.

Die ARP-Tabelle ist notwendig, da die Ethernetadressen und die IP-Adressen unabhängig voneinander festgelegt werden, und es daher keinen Algorithmus gibt mit dem man aus der IP-Adresse die Ethernetadresse ermitteln könnte. Die

IP-Adresse wird vom Netzwerk-Manager festgelegt, und hängt von der Stelle im Internet ab, an der sich der Computer befindet. Wenn der Computer an einer anderen Stelle ans Internet angeschlossen wird, muss die IP-Adresse geändert werden. Die Ethernetadresse wird vom Hersteller der Hardware festgelegt, entsprechend dem Ethernet-Adressraum für den der Hersteller eine Lizenz hat. Wird die Ethernet Hardware (Ethernet-Karte) ausgewechselt, ändert sich damit auch die Ethernetadresse.

4.2 Ein typisches Übersetzungs-Szenarium

Eine Netzwerkanwendung, wie z.B. Telnet, sendet eine 'application message' an das TCP-Modul, worauf das TCP-Modul eine entsprechende TCP-Message zum IP-Modul schickt. Die IP-Adresse des Zielcomputers ist der Anwen-dung dem TCP- und dem IP-Modul bekannt. Im IP-Modul wird das IP- Paket gebildet und zum Ethernettreiber ge-schickt, doch davor mu die Ethernetadresse des Zielcomputers bestimmt werden. Dies geschieht mittels der ARP-Tabelle.

4.3 Das ARP Request/Response Paar

Hier stellt sich nun die Frage woher der Inhalt der ARP-Tabelle kommt. Die Antwort lautet: die Tabelle wird automatisch nach dem Bedarfsprinzip gefüllt. Zwei Dinge passieren, wenn zu einer IP-Adresse keine Ethernetadresse in der ARP-Tabelle gefunden wird:

1. Ein ARP-Anforderungspaket (ARP-request-paket) mit der Broadcast-Ethernetadresse ('an alle') wird an alle im Netz befindlichen Computer gesendet.

2. Das abgehende IP-Paket wird zurückgehalten und geht in eine Warteschlange.

Die Ethernet-Interfaces aller im Netz befindlichen Computer empfangen das Broadcast-Ethernet-Frame. Jeder Ethernettreiber prüft das Typfeld im Ethernet-Frame und leitet das ARP-Paket zum ARP-Modul weiter. In dem ARP-Paket ist nun die Frage enthalten *Wenn du die im Zielfeld genannte IP-Adresse die deine ist, schicke deine Ether-netadresse zum Absender zurück.*. Ein ARP-Request-Paket enthält also folgendes:

Sender IP-Adresse	141.20.30.3
Sender Enetadresse	08-00-5A-21-A7-22
Ziel IP-Adresse	141.20.30.2
Ziel Enetadresse	<leer>

Tab. 2.: Beispiel eines ARP-Request

(errinnern wir uns, da die Broadcast-Adresse FF-FF-FF-FF-FF-FF im Ethernettreiber 'hängen' blieb, und sie daher im ARP-Paket leer ist.) Jeder ARP-Modul prüft nun die vorliegende Ziel-IP-Adresse und wenn es die eigene IP-Adresse ist, sendet er eine Antwort an die Absenderadresse. Das ARP-Response-Paket enthält die Information *Ja, dies ist meine IP-Adresse, meine Ethernetadresse befindet sich im Absenderfeld.* D.h die gesuchte Ethernetadresse befindet sich im Adressfeld des Antwortpaketes. Es könnte so aussehen:

Sender IP-Adresse	141.20.30.2
Sender Enetadresse	08-00-28-00-38-A9
Ziel IP-Adresse	141.20.30.3
Ziel Enetadresse	08-00-5A-21-A7-22

Tab.3: Beispiel einer ARP Response

Die obige Antwort erreicht den ursprünglichen Absender. Der Ethernettreiber schaut auf das Typfeld und schickt das ARP-Paket zum ARP-Modul. Das ARP-Modul entnimmt dem Paket nun die IP-Nummer und die zugehörige Ethernetadresse und ergänzt damit seine ARP-Tabelle.

Die ARP-Tabelle sieht nun folgendermassen aus:

IP-Adresse	Ethernetadresse
141.20.30.1	08-00-39-00-2F-C3
141.20.30.2	08-00-28-00-38-A9
141.20.30.3	08-00-5A-21-A7-22
141.20.30.4	08-00-10-99-AC-54

Tab.4.: ARP-Tabelle nach der Antwort

4.4 Das Szenarium geht weiter

Der neue Tabelleneintrag ist nun installiert, automatisch und nur Millisekunden nachdem er gebraucht wurde. Wir erinern uns nun an Schritt 2 in dem das IP-Paket welches abgeschickt werden sollte, in eine Warteschlange geschickt wurde. Nun wo die erweiterte ARP-Tabelle dazu in der Lage ist die gesuchte Ethernetadresse zu liefern, kann der Ethernettreiber aus dem IP-Paket ein Ethernet-Frame bilden und dies über das Ethernet verschicken. Mit diesen neuen Schritten 3, 4 und 5 sieht das gesamte Szenarium wie folgt aus:

1. Ein ARP-Anforderungspaket (ARP-request-paket) mit der Broadcast-Ethernetadresse ('an alle') wird an alle im Netz befindlichen Computer gesendet.

2. Das abgehende IP-Paket wird zurückgehalten und geht in eine Warteschlange.

3. Die ARP-Antwort enthält die notwendige Ergänzung der ARP-Tabelle (IP zu Ethernet-Adresse).

4. Das in der Warteschlnge befindliche IP-Paket wird mittels der ARP-Tabelle um die Ethernet- Adresse des Zielcomputers ergänzt.

5. Das in 4. gebildete Ethernet Frame wird über das Ethernet verschickt.

Zusammengefasst: wenn in der ARP-Tabelle die zur IP-Adresse gehörige Ethernetadresse nicht gefunden wird, wird das IP-Paket zurückgehalten. Die Übersetzungstabelle wird über ein ARP-Request/Response ergänzt und das zurückgehaltene IP-Paket nun verschickt.

Jeder Computer hat für jedes vorhandene Ethernet-Interface eine eigene ARP-Tabelle. Wenn ein Zielcomputer nicht existiert trifft keine ARP-Response ein, und die ARP-Tabelle kann nicht ergänzt werden. Das IP-Modul muss darauf verzichten das IP-Paket an diese Adresse zu verschicken. Die darüber liegende Protokollschicht kann zwischen einem unterbrochenen Ethernet und der Abwesenheit eines Computers mit der gewünschten IP-Adresse nicht unterscheiden.

Einige Implementierungen von IP und ARP schicken das IP-Paket nicht in eine Warteschlange, während der Wartezeit auf die ARP-Response. Stattdessen wird das Datenpaket an das TCP-oder UDP-Modul zurückgeschickt. Nach einer gewissen Zeit wird von diesen Modulen ein erneuter Versuch gestartet, der durch die inzwischen stattgefundene Ergänzung der ARP-Tabelle erfolgreich ist.

5 Das Internet Protokoll

Das IP-Modul bildet das Kernstück der Internet Technologie, und sein Schwerpunkt ist seine Route- Tabelle. IP be-nutzt diese im Speicher liegende Tabelle um alle Entscheidungen bei der Verschickung von IP-Paketen zu treffen. Der Inhalt der Route-Tabelle wird vom Netzwerkadministrator bestimmt. Um zu Verstehen wie die Route-Tabelle benutzt wird, müssen wir verstehen wie das Internet arbeitet. Dies Wissen ist notwendig für eine erfolgreiche Administration und Wartung eines IP-Netzes. Die Route-Tabelle wird am besten zu verstehen sein, wenn wir uns zuerst einen Überblick über den Datentransport (routing) verschaffen, dann die IP Netzwerkadressen beschreiben und uns dann mit den Details des Routings beschäftigen.

5.1 Direktes Routing

Die Fig.6 zeigt ein kleines Internet, bestehend aus den 3 Computers A, B und C. Jeder Computer verfügt über den Protokoll-Stapel, wie er in Fig.1 (Seite 5) dargestellt ist. Jeder Computer besitzt ein eigenes Ethernet-Interface mit

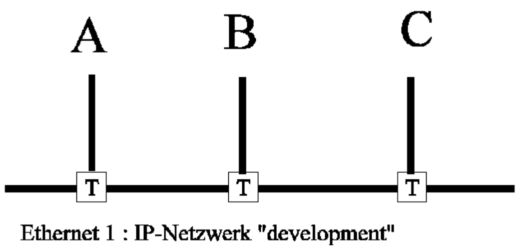

Fig.6: Ein einzelnes IP-Netzwerk

einer eigenen Ethernet-Adresse. Jedem Computer wurde durch den Netzwerkmanager eine eigene IP-Adresse zugewiesen, auch das IP-Netzwerk hat seine Netzwerknummer zugewiesen bekommen.

Wenn A ein IP-Paket an B sendet, enthält dessen IP-Header die IP-Adresse von A, als die IP-Adresse des Absenders und der Ethernet-Header enthält als Absender-Ethernetadresse die von A . Ebenso enthält der IP-Header die IP-Adresse von B als IP-Ziel und der Ethernet-Header enthält als Ziel die Ethernet-Adresse von B.

Adress	Absender	Ziel
IP-Header	A	B
Ethernet-Header	A	B

Tab.5: Adressen in einem Ethernet Frame für ein IP-Paket von A nach B

In diesem einfachen Fall ist der IP-Header doppelt gemoppelt, und die IP-Informationen liefern nichts was im Ethernet-Header nicht schon enthalten wäre, das einzige was die beiden Adresspaare verursachen sind zusätzliche Kosten, durch die zusätzliche CPU-Zeit und die Netzbelastung. Wenn das IP-Paket von A im IP-Modul von B eintrifft, vergleicht dieser die Ziel-IP-Adresse mit seiner eigenen, stellt Übereinstimmung fest, und transportiert das Datagramm anschliessend in die darüber liegende Protokollschicht. Diese Art der Kommunikation zwischen A und B heisst ‘direct routing’.

5.2 Indirektes Routing

Ein wesentlich realistischeres Bild eines Internets zeigt Fig.7. Es setzt sich zusammen aus 3 Ethernets und 3 IP-Netzwerken die durch einen IP-Router verbunden sind. Dieser Router ist der Computer D. Jedes Netzwerk besteht aus 4 Computern, jeder mit seiner eigenen IP- und Ethernet-Adresse.

Mit Ausnahme von Computer D haben alle Computer den bekannten Protokollstapel aus Fig.1. Computer D ist ein IP-Router. Er ist mit allen drei Netzen

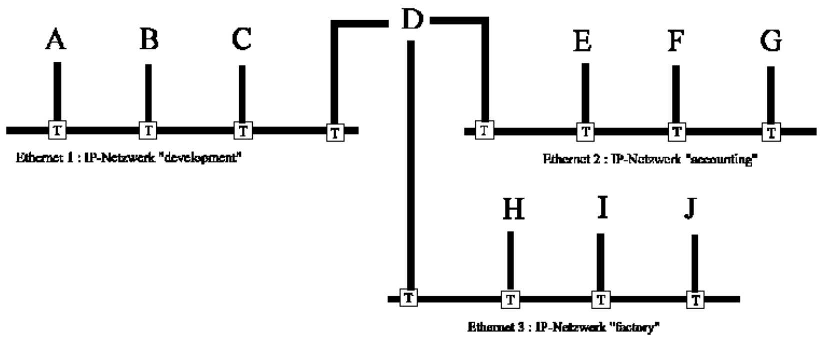

Fig.7: Drei IP-Netzwerke in einem Internet

verbunden und hat deshalb 3 IP-Adressen und 3 Ethernetadressen. Der Computer D hat einen TCP/IP-Protokollstapel ähnlich dem, der in Fig.3 dargestellt wurde, wobei er jedoch 3 ARP-Module und 3 Ethernet-Treiber anstelle der jeweils zwei in Fig.3. Der Computer D hat jedoch nur ein IP-Modul.

Der Netzwerkmanager hat jedem Ethernet eine eigene Nummer, IP-Netzwerknummer genannt, zugeteilt. In Fig.7 sind die Netzwerknummern nicht aufgeführt, sondern nur die Netzwerknamen.

Wenn der Computer A an den Computer B ein IP-Paket verschickt, ist der Vorgang derselbe wie in dem o.g. einfachen Beispiel in einem einfachen Netzwerk. Jede Kommunikation in einem einfachen Netzwerk verläuft nach dem Schema des 'direct routing'.

Die Computer A und D kommunizieren direkt. Dies gilt auch für D und E, und D und H, da in all diesen Fällen die beteiligten Computer demselben IP-Netzwerk angehören. Wenn jedoch der Computer A mit einem Computer jenseits des Routers D kommuniziert ist dies keine direkte Kommunikation mehr.

Wenn A mit einem Computer ausserhalb des eigenen Netzwerkes kommunizeren will, muss er die Dienste von D in Anspruch nehmen. Diese Kommunikation nennt man indirekt. Für dieses s.g. Routen von IP-Paketen sind die IP-Module zuständig. Es geschieht für TCP, UDP und die Netzwerkanwendungen vollständig transparent.

Wenn A ein IP-Paket an E sendet, sind die IP- und Ethernet-Absenderadressen in diesem Paket die von A. Die IP-Zieladresse ist die IP-Adresse von E. Da jedoch der Computer A das IP-Paket an D schicken mu, damit dieser es an E weiterbefördert (forwarding) lautet die Ethernet-Ziel-Adresse D.

Adress	Absender	Ziel
IP-Header	A	E
Ethernet Header	A	D

Tab. 6: Adressen eines Ethernet Frame: IP-Paket von A nach E (vor D)

Der Ethernet-Treiber von D empfängt das Paket und leitet es an das IP-Modul weiter, dieser stellt fest, da es sich nicht um die eigene sondern um die IP-Adresse von E handelt und schickt das Paket dorthin.

Adresse	Absender	Ziel
IP-Header	A	E
Ethernet Header	D	E

Tab. 7: Adressen eines Ethernet Frames: IP-Paket von A nach E (nach D)

Fassen wir zusammen: für die direkte Kommunikation zwischen zwei Computern gilt, da beide Absenderadressen (IP-Nummer und Ethernetadresse) diejenigen des absendenden Computers sind und beide Zieladressen diejenigen des Computers zu dem das Datenpaket soll. Bei einer indirekten Kommunikation stimmen die Adressen nicht mehr auf diese Art paarweise überein.

Das hier angenommene Internet ist immer noch ein sehr einfaches Beispiel. Reale Netzwerke werden durch viele Faktoren komplizierter. Sie bestehen aus einer Vielzahl von Routern und verschiedenen Typen von physikalischen Netzwerken. Das Beispiel zeigt jedoch, wie ein grosses Ethernet in kleinere aufgespalten werden kann, um den Ethernet-Broadcastverkehr zu lokalisieren.

5.3 Die Routing-Regeln des IP-Moduls

Die Übersicht im vorigen Abschnitt zeigte zwar, was passiert, aber nicht wie es funktioniert. Wir sehen uns daher jetzt die Regeln oder Algorithmen an, mit deren Hilfe das IP-Modul diese Aufgaben löst.

- Für ein abgehendes IP-Paket (das Paket kommt von einer höheren Protokollschicht) muss IP entscheiden, ob es direkt oder indirekt zu verschicken ist. Ausserdem muss es entscheiden, an welches der unter ihm liegenden Netzwerkinterfaces das Paket weiterzureichen ist. Diese Entscheidungen werden mit Hilfe der Route-Tabelle getroffen.
- Für ein ankommendes IP-Paket (das Paket kommt von der darunter liegenden Schicht der Netzwerkinterfaces) muss IP entscheiden, ob das Paket im Netz weiter zu befördern ist (forwarding) oder ob es an die darüber liegenden Protokollschicht zu leiten ist. Ist das Paket weiterzuleiten, ist es wie ein abgehendes Paket zu behandeln.
- Ein ankommendes Paket, welches weiterzureichen ist, ist niemals über daselbe Interface weiterzuleiten, über das es eingegangen ist.

Diese Entscheidungen sind zu treffen, bevor das IP-Paket weitergereicht wird, und bevor die ARP- Tabelle konsultiert wird.

5.4 IP-Adressen

Ein Netzwerkmanager weist jedem Computer eine IP-Adresse zu, entsprechend dem IP-Netzwerk, an dem der Computer angeschlossen ist. Ein Teil der 4 Byte langen Adresse stellt die Nummer des Netzwerks dar, der andere Teil der IP-Nummer stellt die Nummer des Computers innerhalb des Netzwerkes dar (Host-Nummer). Für den Computer in der Tab.1. mit der IP-Nummer 141.20.30.3 lauetet die Netzwerknummer 141.20.30 und die Hostnummer ist 3.

Der Anteil der IP-Nummer der die Netzwerknummer darstellt (der Rest ist immer die Hostnummer) wird bestimmt durch die ersten 4 Bit der IP-Nummer. Alle Beispielnummern in diesem Lehrgang sind vom Type C. Das bedeutet, da die ersten 3 Bits beinhalten, da die nachfolgenden 21 Bit die Netzwerkadresse und die letzten 8 Bit die Host-nummer darstellen. In der Klasse C sind also 2 097 152 Netzwerke möglich, in jedem Netzwerk bis zu 254 Hosts (die Nummern 0 und 255 sind reserviert).

Die Vergabe der IP-Adressen wird vom NIC (Network Information Center) wahrgenommen. Alle Internets die zu einem einzigen weltweiten Internet zusammengeschlossen sind, müssen die vom NIC zugewiesenen Nummern verwenden. Auch wenn jemand der ein eigenes Netz aufbaut, ohne die Absicht es an das Internet anzuschliesssen, sollte sich die Nummern vom NIC zuweisen lassen, da es unweigerlich zu Verwechslungen und Chaos führt, wenn er sich später dazu entschliesst eine Verbindung zu einem anderen Internet herzustellen.

5.5 Namen

Die meisten Leute sprechen einen Computer lieber mit einem Namen als mit einer Nummer an. In der IP- Technologie ist deshalb die Möglichkeit enthalten jeder IP-Nummer einen Namen zuzuweisen. Nehmen wir an der Computer mit der Nummer 223.1.2.1 habe den Namen alpha. In kleineren Netzwerken wird häufig die Übersetzung von Name-zu-IP-Nummer in jedem Computer einzeln durchgeführt, und zwar mit Hilfe der Datei 'hosts' (unter UNIX meist im Verzeichnis /etc). Für grössere Netzwerke ist diese Übersetzungsdatei in einem Server gespeichert, und jeder Host aus dem Netzwerk greift nach Bedarf auf diesen Computer zu. Einige Zeilen aus dieser Datei für das in Fig.7 dargestellte Netzwerk könnten so aussehen:

IP-Nummer	Name	Host
223.1.2.1	alpha	A
223.1.2.2	beta	B
223.1.2.3	gamma	C
223.1.2.4	delta	D
223.1.3.2	epsilon	E
223.1.4.2	iota	I

Die IP-Adressen stehen in der ersten, die Namen in der zweiten Spalte, in der dritten Spalte stehen die Hostbezeichnungen aus Fig.7. In den meisten Fällen kann man identische 'hosts'-Dateien auf allen Computern einrichten. Sicher werden sie bemerkt haben, da für delta nur ein Eintrag in der Tabelle erfolgte, obwohl delta 3 IP- Adressen hat. Delta kann nämlich mit jeder seiner IP-Adressen erreicht werden, und es ist gleichgültig welche der Adressen benutzt wird. Wenn ein IP-Paket das IP-Modul von delta erreicht, vergleicht dieser die Zieladresse im IP-Paket mit allen eigenen Adressen.

Aber auch die einzelnen Netzwerke werden mit Namen versehen. Wenn wir 3 Netzwerke vorzuliegen haben, könnte die 'networks'-Datei, welche die Namemszuteilung dokumentiert, wie folgt aussehen:

Netzwerknummer	Netzwerkname
223.1.2	development
223.1.3	accounting
223.1.4	factory

(Wir errinern uns, da für Netzwerke vom Typ C die ersten 3 Byte der IP-Nummer die Netzwerkadresse sind).

In diesem Beispiel ist also alpha der Computer Nr.1 im 'development'-Netzwerk, beta Nr. 2 im selben Netzwerk usw.. Man kann auch sagen alpha ist development.1, beta ist development.2 usw.

Die obige 'hosts'-Datei ist für den Nutzer ausreichend, der Netzwerkmanager wird aber wahrscheinlich die Zeile für den Computer delta ersetzen durch den folgenden Eintrag:

223.1.2.4	devnetrouter	delta
223.1.3.1	facnetrouter	
223.1.4.1	accnetrouter	

Diese 3 neuen Zeilen in der 'hosts'-Datei enthalten nun alle 3 IP-Nummern von delta und seinen Namen. Tatsächlich enthält die erste Zeile zwei Namen für den Host, sie können beide synonym verwendet werden. In der Praxis ist delta der allgemein gebräuchliche Name, die anderen 3 Namen werden lediglich für die Verwaltung der Routing-Tabelle verwendet. Diese Dateien werden von Kommandos benutzt, die der Netzwerkverwaltung dienen und von den Netzwerkanwendungen. Für die Funktion von Internet sind sie nicht notwendig, aber sie machen die Arbeit für uns leichter.

5.6 IP Route-Tabelle

Woher weiss IP welches der unter ihm liegenden Netzwerkinterfaces (Ethernett-reiber) zu verwenden ist, wenn ein IP-Paket abgeschickt werden soll? Zu diesem Zweck wurde im IP die Routing-(oder Verbindungs-)Tabelle eingerichtet. Der Schlüssel mit dem in der Tabelle gesucht wird, wird aus der IP- Adresse des Zielcomputers gewonnen, deren er-sten 3 Byte ja die Adresse des Netzwerkes steckt.

Die Routing-Tabelle enthält eine Zeile für jede Verbindung. In den einzelnen Spalten stehen dann die IP- Netzwerknummern, das direkt/indirekt-Flag, die IP-Adresse des Routers und die Nummer des zu verwendenden Interface. Vor jedem Abschicken eines IP-Paketes muss diese Tabelle konsultiert werden. In den meisten Computern kann die Routetabelle mit dem Befehl ROUTE" modifiziert werden. Der Systemverwalter (Superuser) trägt so die vom Netzwerkmanager zu vergebenden IP-Adressen ein.

5.7 Details des Direct Routing

Um dies zu veranschaulichen, wollen wir das Beispiel, das wir weiter oben hatten, nochmals genauer betrachten.

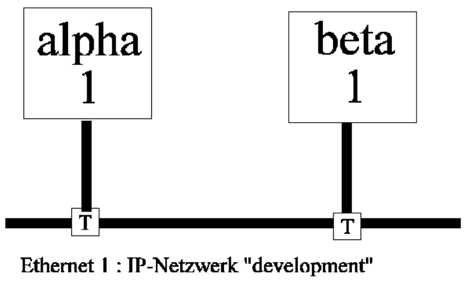

Fig.8: Detail eines IP-Netzwerkes

Die Route-Tabelle in alpha könnte dann so aussehen:

Netzwerk	Direct/indirect Flag	Router	Interfacenummer
development	direct	<leer>	1

Tab.8: Beispiel einer einfachen Route-Tabelle

Auf einigen UNIX-Systemen kann man sich diese Tabelle in dieser Form mit dem Befehl netstat -r" ansehen. In unserem einfachen Beispiel haben alle Computer dieselbe Route-Tabelle.

Zur Verdeutlichung hier dieselbe Tabelle allerdings mit den entsprechenden Netzwerknummern statt der Namen.

Netzwerk	Direct/indirect Flag	Router	Interfacenummer
223.1.2	direct	<leer>	1

Tab.9: Beispiel einer einfachen Route-Tabelle mit Nummern

5.8 Das Szenarium einer direkten Kommunikation

Alpha sendet ein IP-Paket an beta. Das IP-Paket befinde sich im IP-Modul von alpha, und die IP-Adresse von beta (Ziel) sei beta oder 223.1.2.2. Das IP-Modul extrahiert daraus die Netzadresse von beta und fragt damit die Route-Tabelle ab. Es wird in der ersten Zeile ein passender Eintrag gefunden (Netznummer stimmt überein).

Die nächsten Eintragungen in der Zeile der Route-Tabelle besagen, da der gewünschte Computer direkt über das Interface 1 erreicht werden kann. Mittels der ARP-Tabelle erfolgt nun die Übersetzung der IP- Adresse in die Ethernet-Adresse von beta und das Ethernet-Frame wird über das Interface 1 direkt an beta verschickt.

Wenn eine Anwendung versucht Daten an eine IP-Adresse zu verschicken, welche sich nicht im Netzwerk ‘development’ befindet, findet IP in der Route-Tabelle keinen passenden Eintrag und verwirft das Datenpaket. Einige Computer machen eine Fehlermitteilung Network not reachable”.

5.9 Details des Indirect Routing

Wir wollen uns nun den komplizierteren Fall des indirekten Routing genauer ansehen.

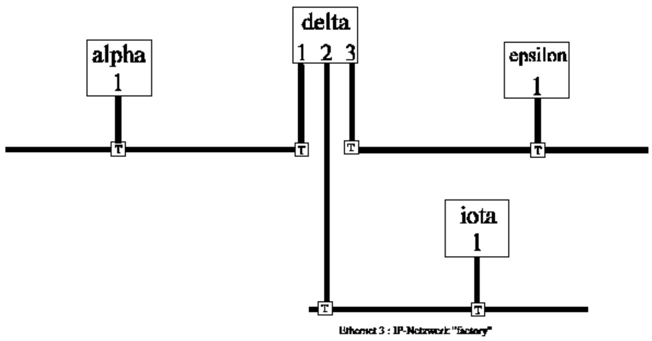

Fig.9: Details von drei verbundenen Netzwerken

Die Route-Tabelle in alpha könnte jetzt so aussehen:

Netzwerk	Direct/indirect Flag	Router	Interfacenummer
development	direct	<leer>	1
accounting	indirect	devnetrouter	1
factory	indirect	devnetrouter	1

Tab.10: Route-Tabelle in alpha

Zur Verdeutlichung dieselbe Tabelle nochmals allerdings mit den entsprechenden Netzwerknummern statt der Namen (Tab.11).

Netzwerk	Direct/indirect Flag	Router	Interfacenummer
223.1.2	direct	<leer>	1
223.1.3	indirect	223.1.2.4	1
223.1.4	indirect	223.1.2.4	1

Tab.11: Route-Tabelle in alpha mit Nummern

Der Router in der Routetabelle von alpha ist die IP-Adresse die delta im Netzwerk development hat.

5.10 Das Szenarium einer indirekten Kommunikation

alpha sendet ein IP-Paket an epsilon. Das IP-Paket befinde sich im IP-Modul von alpha, und die IP- Adresse des Ziels sei die IP-Nummer von epsilon (223.1.3.2). Das IP-Modul extrahiert die Netzadresse des Netzes, in dem sich epsilon befindet (223.1.3), untersucht die Route-Tabelle und findet in der zweiten Zeile Übereinstimmung. In diesem Eintrag ist vermerkt, da die Computer im Netz 223.1.3 über den Router devnetrouter erreicht werden können. Das IP-Modul von alpha konsultiert nun die ARP-Tabelle um die Ethernetadresse von devnetrouter zu ermitteln und schickt das IP-Paket über sein Interface 1 zum devnetrouter. Das IP-Paket enthält jedoch immer noch als IP-Zieladresse die von epsilon.

Das IP-Paket erreicht delta über das Interface 1 mit dem delta mit dem Netzwerk development verbunden ist, und wird an das IP-Modul von delta weitergeleitet. Dort wird die IP-Adresse des Ziels untersucht. Da es sich nicht um eine eigene (eine von den 3 die delta gehören) handelt, entscheidet delta, da das IP-Paket weiterzuleiten ist.

Das IP-Modul von delta extrahiert nun ebenfalls die Netzwerknummer aus der IP-Adresse des Zielcomputers (223.1.3) und konsultiert seine Route-Tabelle. Dies sieht folgendermaen aus:

Netzwerk	Direct/indirect Flag	Router	Interfacenummer
development	direct	<leer>	1
factory	direct	<leer>	3
accounting	direct	<leer>	2

Tab.12: Route-Tabelle in delta

Dieselbe Tabelle nochmal jedoch mit den Nummern statt der Namen (Tab.13):

Netzwerk	Direct/indirect Flag	Router	Interfacenummer
223.1.2	direct	<leer>	1
223.1.3	direct	<leer>	3
223.1.4	direct	<leer>	2

Tab.13: Route-Tabelle in delta mit Nummern

In der zweiten Zeile wird Übereinstimmung gefunden. Nun sendet das IP-Modul das Paket an epsilon über das Interface 3. Das Ethernet-Frame enthält als Zieladresssen sowohl im IP-Header als auch im Ethernet-Header die von Epsilon. Das Paket erreicht epsilon und dort das IP-Modul. Dieses erkennt die eigen IP-Nummer als Zieladresse und reicht das Paket zu der darüber liegenden Protokollschicht weiter.

5.11 Zusammenfassung des Routing

Wenn ein IP-Paket durch ein grosses Internet wandert, kann es durch viele Router kommen, bevor es sein Ziel erreicht. Der Weg den es nimmt wird nicht durch eine zentrale Leitstelle festgelegt, sondern ist das Ergebnis des Inhalts aller Route-Tabellen auf welche das Paket auf seiner Reise trifft. Jeder Computer legt lediglich den nächsten Schritt fest und zu welchem Computer das IP-Paket in diesem Schritt übertragen wird. In einigen UNIX-Systemen kann der Weg eines Pakets verfolgt werden, mit dem Befehl `traceroute <IP-Adresse des Ziels>`.

5.12 Verwaltung der Route-Tabellen

Das Unterhalten von korrekten Route-Tabellen in allen Computern eines grossen Netzwerkes ist eine schwierige Aufgabe. Die Konfiguration der Netze wird durch die Netzwerkmanager ständig verändert, um den Anforderungen gerecht zu werden. Fehler in den Route-Tabellen können die Kommunikation jedoch in einer Art und Weise blockieren, deren Ursachen nur durch entsetzlich weitschweifige Analysen zu entdecken sind. Es ist ein weiter Weg von der Konfiguration eines einfachen Netzwerkes zu einem zuverlässigen Internet. Die einfachste Methode einem Ethernet ein IP-Netzwerk zuzuweisen, ist es jedem Ethernet eine einzelne IP-Netzwerknummer zuzuweisen.

Hilfe kommt auch von einigen Protokoll- und Netzwerk-Anwendungen. ICMP (Internet Control Message Protocol) kann verschiedene Routing-Probleme erkennen und melden. Für kleine Netzwerke werden die Route-Tabellen für jeden Computer vom Netzadministrator von Hand erstellt. Für grössere Netze wird dieser Vorgang automatisiert, mittels eines Routing-Protokolls, welches die Wege durch das Netz bestimmt.

Wenn ein Computer in ein anderes Netz geschaltet wird, muss seine IP-Nummer geändert werden. Wenn er aus einem Netz entfernt wird, wird seine IP-Nummer ungültig. Solche Veränderungen verursachen regelmässige Änderungen in den 'hosts'-Dateien. Bereits bei mittelgroen Netzwerken kann es schwierig werden diese kleine Datei in allen Computern auf dem aktuellen Stand zu halten. Das 'Domain Name System' hilft dieses Problem zu lösen.

6 User Datagram Protocol

UDP ist eines der beiden Hauptprotokolle die über dem IP angeordnet sind. Es bietet Dienste für nutzerbezogene Netzwerkanwendungen. Beispiele für Netzwerkanwendungen die UDP nutzen sind

- Network File System NFS
- Simple Network Management Protocol SNMP

Die Dienste die UDP bietet liegen nur wenig über den Möglichkeiten der IP-Schnittstelle.

UDP ist eine verbindungslose Datenübermittlung, welche keine Garantie für die Zustellung enthält. UDP unterhält keine Verbindung zu dem entfernten UDP-Modul, sondern schiebt die Daten einfach auf das Netz und akzeptiert ankommende Datagramme vom Netz ohne in direkte Verbindung mit der Gegenstelle zu treten. UDP fügt den Möglichkeiten von IP, nur zwei Dinge hinzu:

- die eine ist die Möglichkeit des Auffächerns (Multiplexen) der Informationen auf verschiedene Anwendungen auf der Basis von Portnummern
- die andere ist die Bildung einer Kontrollsume um die Integrität der übermittelten Daten zu sichern.

6.1 Ports

Wie kann ein Client auf dem einen Computer einen Server auf einem anderen erreichen?

Der Kommunikationsweg zwischen einer Anwendung und UDP geht über eines der UDP-Ports. Diese Ports sind durchnummeriert, beginnend mit Null. Eine Anwendung welche Dienste anbietet (Server) wartet auf Messages welche über das für die Anwendung bestimmte Port einkommen. Der Server wartet geduldig auf einen Client welcher seine Dienste in Anspruch nehmen möchte.

So wartet beispielweise der SNMP-Server , genannt ein SNMP Agent, immer auf Port 161. Es kann in einem Computer nur ein SNMP-Agent installiert sein, da es nur ein UDP-Port 161 gibt. Diese Nummer ist bekannt, da es eine feste durch das Internet zugewiesene Nummer ist. Wenn ein SNMP-Client die Dienste beansprucht, sendet er seine Anforderung an das UDP-Port 161 des Zielcomputers.

Wenn eine Anwendung eine Nachricht über UDP verschickt, kommt am anderen Ende auch genaue eine Nachricht an. Wenn z.B. eine Anwendung fünf mal auf ein UDP-Port schreibt, wird die Anwendung am anderen Ende auch genau fünf Mal von dem entsprechenden Port lesen. Auch die Grösse der empfangenen Nachricht entspricht genau der die abgesandt wurde. UDP bewahrt die Grösse einer Nachricht wie sie von der Anwendung festgelegt wurde. UDP wird nie zwei Nachrichten miteinander verbinden zu einer, oder eine Nachricht teilen.

6.2 Kontrollsumme

Ein ankommendes IP-Paket in dessen Typfeld des IP-Headers 'UDP' vermerkt ist, wird vom IP zum UDP-Modul weitergeleitet. Wenn das UDP-Modul das Datagramm erhält, prüft es die UDP- Kontrollsumme. Wenn die Prüfsumme im Datagramm Null ist, bedeutet dies, dass der Sender dem Datenpaket keine Prüfsumme mit auf den Weg gegeben hat, in diesem Fall kann der Prüfsummenmechanismus übergangen werden. Ist Ethernet das einzige Netzwerk zwischen den UDP- Modulen, braucht es keine Prüfsumme. Es ist aber trotzdem empfehlenswert den Prüfsummentest einzuschalten, da beispielsweise einmal die Änderung einer Route-Tabelle über ein unzuverlässigeres Medium erfolgt, und wie wir oben gesehen haben können Fehler in einer Route-Tabelle verheerende Folgen haben.

Wenn die Prüfsumme gültig ist (oder Null), wird geprüft, ob an der verlangten Portnummer eine Anwendung liegt,, und wenn dies zutrifft die Nachricht in eine Warteschlange eingereiht, damit die Anwendung sie von dort lesen kann. Durch die Warteschlange können Daten zwischengespeichert werden, wenn die Nachrichten schneller eintreffen als die Applikation sie lesen kann. Ist an dem gewünschten Port keine Anwendung angebunden oder der Speicher für die Warteschlange ist voll, wird die Nachricht verworfen, und ist damit verloren. Sollten weitere Nachrichten eintreffen, werden sie solange verworfen bis wieder Platz in der Warteschlange ist.

7 Transmission Control Protocol (TCP)

Die Dienste von TCP unterscheiden sich wesentlich von UDP. TCP bietet eine verbindungsorientierten Datenstrom, anstelle der verbindungslosen Datagrammübergabe bei UDP. TCP garantiert die Datenzustellung, während dies UDP nicht tut. TCP wird von Anwendungen benutzt, die eine absolut sichere Zustellung der Daten erfordern, und nicht gestört werden können durch Time-Outs oder Rücksendungen. Die typischsten Netzanwendungen die TCP nutzen sind FTP (File Transfer Protocol) und Telnet. Andere verbreitete Netzanwendungen die auf TCP basieren sind das X-Windows-System und rcp (remote copy) und die Kommandos der r-Serie. Die gösseren Möglichkeiten von TCP haben allerdings auch ihren Preis in Form von grösserer CPU- Belastung und grösserer Bandbreite auf dem Netz. Der interne Aufbau der TCP-Module ist wesentlich komplexer als derjenige der UDP-Module. Ähnlich wie bei UDP sind die Netzwerkanwendungen mit TCP-Ports verbunden. Für die spezifischen Anwendungen sind genau festgelegte Portnummern bestimmt. So benutzt z.B. Telnet das Port Nr. 23. Der Telnet-Client kann den Server einfach dadurch finden, da er sich mit dem Port 23 auf dem gewünschten Computer verbindet. Wenn eine Anwendung die TCP benutzt startet, kommunizieren zuerst die TCP-Module des Client- und des Server-Computers miteinander. Die TCP-Module an beiden Enden der Verbindung tauschen Statusinformationen aus und bilden eine virtuelle Schleife. Diese virtuelle Schleife beansprucht Hilfsmittel in beiden TCP-Modulen. Die virtuelle Schleife ist voll duplex, d.h Daten können in beiden Richtungen simultan transportiert werden. Die Applikation schreibt Daten auf das TCP-Port, sie gehen über das Netz und werden von der Applikation am anderen Ende gelesen.

TCP macht aus dem Byte-Strom Pakete nach eigenem Belieben, eventuelle Grenzen zwischen enzelnen Abschnitten einer Nachricht werden nicht berücksichtigt. Wenn eine Anwendung beispielsweise 5 mal auf das TCP-Port schreibt, kann es vorkommen, da die Anwendung auf der anderen Seite 10 mal lesen muss, um alle Daten zu bekommen, oder sie mu auch nur einmal lesen. Es besteht keine Korrelation zwischen der Anzahl und dem Umfang der einzelnen Schreib- und Leseoperationen an den beiden Enden. TCP ist vergleichbar mit einem gleitenden Fenster, da sich über die Zeichen einer Mitteilung schiebt. In diesen Mechanismus sind ausserdem Zeitüberschreitungs- und Rücktransportmechanismen eingebaut. Alle abgehenden Daten müssen vom TCP der Gegenstelle bestätigt werden. Die Bestätigungen können an die Daten angehangen werden. Beide empfangenden Seiten können den Datenfluss am anderen Ende steuern, damit wird einem Überlauf des Zwischenspeichers vorgebeugt. Wie bei allen Protokollen die nach dem Prinzip des gleitenden Fensters arbeiten, hat das Fenster eine vorgeschriebene Grösse. Erst nach der Übermittlung eines ganzen Fensters ist eine Bestätigung erforderlich. Für TCP ist die Fenstergrösse in Bytes vorgeschrieben.

8 Netzwerk Anwendungen

Warum gibt es sowohl TCP als auch UDP, und nicht nur das eine oder das andere?

Sie befriedigen verschiedene Anforderungen. Die meisten Anwendungen sind nur für eine der beiden Protokolle ausgelegt. Als Programmierer wählen sie das Protokoll welches ihren Anforderungen besser entspricht. Wenn Sie einen zuverlässigen Datentransport brauchen, kann TCP das Bessere für sie sein. Wenn sie einen

Dategramm-Dienst benötigen, könnte UDP das Beste für sie sein. Wenn sie Effizienz in langgezogenen Schleifen benötigen, sollten sie TCP wählen. Wenn sie Effizienz in in schnellen Netzwerken mit kurzen Verzögerungszeiten brauchen, sollten Sie UDP wählen. Wenn ihre Anforderungen nicht eindeutig in eine diese Kategorien fallen, ist die "beste" Wahl nicht so einfach. Dann gibt es aber noch den Weg, da sie in ihrer Anwendung die Unzulänglichkeiten des gewählten Protokolls ausgleichen. Wenn sie beispielsweise Zuverlässigkeit brauchen und haben UDP gewählt, müssen sie die Zuverlässigkeit in ihrer Anwendung verbessern. Wenn sie TCP gewählt haben, brauchen aber einen Record orientierten Dienst, mu die Anwendung Marken in den Datenstrom einfügen, um die einzelnen Records zu kennzeichnen.

Welche Netzanwendungen sind verfügbar?

Es sind viel zuviel um sie alle aufzulisten. Ihre Zahl wächst beständig. Einige der Anwendungen gibt seit dem Begin der Internet-Technologie: TELNET und FTP. Andere sind relativ neu: X-Windows und SNMP. Im folgenden werden die in diesem Lehrgang vorgestellten Anwendungen kurz beschrieben:

8.1 TELNET

TELNET gibt uns die Möglichkeit über TCP uns in einem fernen Computer einzulogen. Die Arbeitsweise und das äussere Bild sind der Arebitsweise eines Telefons ähnlich. Man wählt über die Tastatur einen Computer an indem man in der Befehlszeile eingibt 'telnet delta' und erhält ein Login-Prompt des gewünschten Computers delta. Telnet arbeitet sehr zuverlässig; es ist eine alte Anwendung und besitzt eine ausgedehnte . Z.B. kann ein Client auf einer VAX/VMS sitzen und der Server auf einer UNIX System V.

8.2 FTP

File Transfer Protocol (FTP) ist so alt wie TELNET, benutzt ebenfalls TCP und hat dieselbe ausgedehnte Interoperabilität. Die Arbeitsweise und das Erscheinungsbild ist fast das gleiche wie bei TELNET. Aber anstatt der üblichen Kommandos hat man einen kleinen Satz von Befehlen um Inhaltsverzeichnisse anzusehen und Dateien von einem Computer zum anderen zu kopieren.

8.3 rsh

Remote shell (rsh oder remsh) ist ein Befehl von einer ganzen Familie von Befehlen um mit UNIX- ähnlichen Kommandos auf einem entfernten Computer arbeiten zu können. Der UNIX Copy-Befehl cp wird zu rcp. Das UNIX Kommando 'wer ist eingelogt' who wird zu rwho. Die Liste lässt sich fortsetzen, wobei stets vor den UNIX-Befehl ein r zu setzen ist. Sie werden daher auch r* Kommandos genannt. Die r* Kommandos werden hauptsächlich zwischen

UNIX-Systemen eingesetzt. Sie wurden hauptsächlich für die Arbeit zwischen 'vertrauten' Hosts entworfen. Hierbei wird weniger Wert auf die Sicherheit gelegt, dafür umso mehr auf die bequeme Bedienbarkeit. Um den Befehl 'cc file.c' auf einem fernen Computer delta auszuführen, ist es lediglich nötig die folgende Zeile einzugeben: 'rsh delta cc file.c'. Um die Datei 'file.c' nach delta zu kopieren, ist 'rcp file.c delta:' einzugeben. Um sich auf delta einzuloggen, mu man 'rlogin delta' eingeben, und wenn der Computer entsprechend konfiguriert wurde, wird man nicht einmal mit einem Passwort belästigt.

8.4 NFS

Das Network File System, zuerst von Sun Microsystems Inc. entwickelt, benutzt UDP und ist hervorragend geeignet um UNIX-File-System auf mehrere Computer zu mounten. Eine disklose Workstation kann damit auf die Harddisk ihres Servers zugreifen, als ob es eine lokale Festplatte wäre. Die Daten die sich auf der Festplatte des Computers alpha befinden, können vom Computer beta benutzt werden als ob sie auf der eigenen Festplatte wären, wenn das Filesystem in dem sich die Dateien befinden, per NFS auf beta gemounted sind. NFS bringt eine erhebliche Netzbelastung mit sich, und ist wegen der lansamen Links oft hinderlich, seine Vorteile sind jedoch unübersehbar. Der NFS-Client ist in den Kernel des Betriebssystems implementiert, um sicherzustellen, da alle Kommandos von den gemounteten Platten Gebrauch machen können als ob es lokale wären.

8.5 SNMP

Simple Network Managment Protocol (SNMP) benutzt UDP und ist für die Verwendung in zentralen Netzwerk-Verwaltungsstationen gedacht. Es ist bekannt, dass wenn genügend Daten vorliegen, der Netzwerkmanager daraus Netzwerkprobleme erkennen und analysieren kann. Die zentrale Station, welche SNMP benutzt sammelt diese Daten von den anderen Computern im Netz ein. SNMP schreibt das Format dieser Daten vor, die zur zentralen Station oder dem Netzwerkmanager geschickt werden zur Interpretation.

8.6 X-Windows

Das X-Windows System benutzt das X-Window-Protocol auf TCP um Fenster auf dem Grafikbildschirm einer Workstation zu zeichnen. X-Window ist jedoch viel mehr als ein Hilfsprogramm zum Zeichnen von Fenstern, es ist eine vollständige Philosophie für den Entwurf einer Benutzeroberfläche.

9 Weitere Informationen

Detaillierte Information über die Internet Technologie wurde in diesem Lehrgang nicht zusammengetragen. Im folgenden ist daher eine Liste von Begriffen angegeben, die dazu anregen soll sich mit ihnen zu beschäftigen. Sie sind als nächste Stufe bei der Erarbeitung weiterer Details zu betrachten, für Leser die mehr lernen wollen.

- Administrationskommandos: arp, route und netstat
- ARP : permanent entry, publish entry, time-out entry, spoofing
- IP route table: host entry, default gateway, subnets
- IP: time-to-live counter , fragmentation, ICMP
- RIP : routing loops
- Domain Name System

10 Literatur

[1] Comer, D. : Ïnterworking with TCP/IP Principles, Protocols and Architecture", Prentice Hall, Englewood Cliffs, New Jersey, USA, 1988

[2] Feinler, E., et al, DDN Protocol Handbook, Volume 2 and 3, DDN Network Information Center, SRI International, 333 Ravenswood Avenue, Room EJ291, Menlow Park, California, USA, 1985

[3] Spider Systems, Ltd., Pakets and Protocols", Spider Systems Ltd., Stanwell Street, Edinburgh, UK EH6 5NG 1990