11 Vermittlung von Netzverbindungen

Sind die Nachrichten für Kommunikations-Teilnehmer in entfernten Netzen bestimmt, so reichen Frames mit ihren Hardware-Adressen nicht aus, um den Teilnehmer sicher zu finden. Die Situation ist vergleichbar mit Ihrem Namen und Ihrer Adresse: Ihr Name entspricht der Hardware-Adresse, der in den Frames gespeichert ist. Sie können sich jedoch an verschiedenen Orten (Adressen) auf der Welt aufhalten. Somit würde Sie ein Brief nicht erreichen, da die Post (das Netz) Ihre gegenwertige Adresse nicht unbedingt kennt. Der Brief muss somit Ihre gegenwertige Adresse aufgedruckt haben und die Post muss die Adresse und den Weg zu dieser Adresse kennen, um eine sichere Zustellung zu garantieren.
Das vorliegende Kapitel beschreibt die Lösungsansätze der Telematik, um dieses grundlegende Problem zu lösen.

11.1 Die Aufgabe der Vermittlung

Das beschriebene Problem wird mithilfe der Vermittlung in Netzen gelöst.
Zu Beginn des Telefoniezeitalters erfolgte die Vermittlung von Gesprächen manuell. Man nahm den Hörer ab und wurde mit einer Telefonistin der nächst gelegenen Vermittlungsstelle verbunden. Die Vermittlerin im Amt schaltete die gewünschte Verbindung zur nächsten Zentrale durch. Dort sass wieder eine Telefonistin, welche die Verbindung weiter aufbaute. Dies wurde so lange fortgeführt, bis die Verbindung bei der Zentrale des Empfängers ankam. Das Telefon beim Empfänger läutete und die Vermittlerin vom Amt liess den Empfänger warten (online), bis sie die aufgebaute Leitung rückwärts zur Zentrale des Senders hin bestätigt hatte. Nach dieser Prozedur konnten die beiden Teilnehmer miteinander sprechen.
Diese Zentralen wurden nach und nach durch Relaissteuerungen ersetzt und in der Folge arbeiten heute alle Zentralen mit digitaler Vermittlung.
Generell gilt das folgende Schema in Abbildung 11.1, das den Stellenwert der Vermittlung darstellt.

Abkürzungen in der Abbildung:
IMP Interface Message Processor, Schnittstellenprozessor
PSTN Public Switched Telephone Network

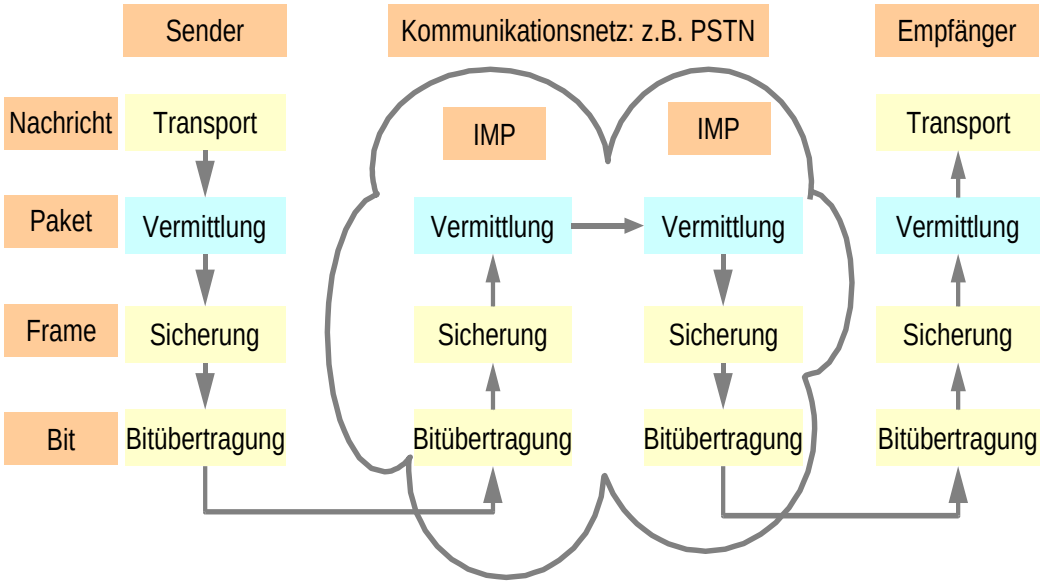

Abbildung 11.1: Das Prinzip der Vermittlung in einem Kommunikationsnetz

11.2 Verbindungsorientiert/Verbindungslos

Praxis-Hinweis:
Die ARPANET (Internet) Entwickler gingen davon aus, dass ein Subnet (Übertragungsnetz) immer etwas Unzuverlässiges sei. Aus diesem Grund verlangten sie, dass die Hostrechner im Netz (Rechner des Senders und des Empfängers) Fehlersuche, Fehlerkorrektur und Flusssteuerung durchzuführen haben.

Auf Grund historischer Zänkereien zwischen der ARPA-Internet-Gemeinde (Advanced Research Projects Agency) und anderen Netzwerkbetreibern (zum Beispiel Telefonnetz-Betreibern) gibt es zwei Auffassungen über die genaue Aufgabe der Vermittlung.
Dies führt dazu, dass sich die Vermittlung auf SEND-PACKET und RECEIVE-PACKET beschränkt und völlig verbindungslos oder verbindungsunabhängig sein sollte. Jedes Paket muss aber eine vollständige Zieladresse haben, da es unabhängig von seinen Vorgängern oder Nachfolgern verschickt wird. Man nennt dies einen verbindungslosen Dienst.
Andere Netzwerkbetreiber, vor allem diejenigen, die das Subnet selber verwalten (beispielsweise PSTN-Betreiber), sehen die Sache etwas anders. Hier herrscht die Ansicht vor, dass das Subnet sehr zuverlässig sei und dass die Vermittlung daher eine Verbindung aufzubauen habe, auf der dann Daten gesendet werden können. Die

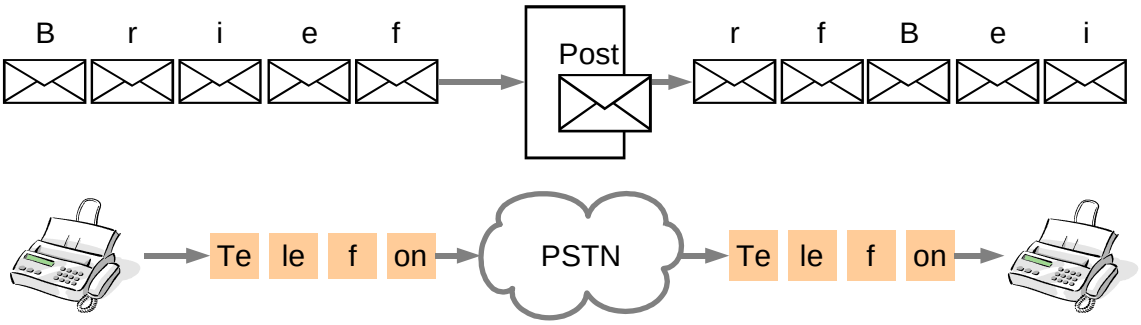

Abbildung 11.2: Verbindungslose und verbindungsorientierte Kommunikation

Verbindung sei mit einer speziellen Kennung zu versehen, welche während der gesamten Datenübertragung verwendet werden müsse. Am Schluss der Übertragung wird die Verbindung ordnungsgemäss abgebaut. Eine spezielle Flusskontrolle soll verhindern, dass der Sender seine Pakete schneller ablegt als der Empfänger. Man nennt dies *verbindungsorientierte* Verbindung.

Das öffentliche Telefonnetz ist verbindungsorientiert, die Briefpost hingegen verbindungslos. Briefe kommen nicht unbedingt in der Reihenfolge des Absenders an. Beim Telefonieren schätzen wir es, wenn die Worte des Partners am anderen Ende in der richtigen Reihenfolge ankommen!

Beim verbindungslosen Dienst hat jedes Paket eine volle Zieladresse und wird unabhängig von allfälligen weiteren Paketen auf individuellen Übermittlungswegen im Netz transportiert. Erst der Empfänger setzt die Pakete wieder in der richtigen Reihenfolge zu einer Nachricht zusammen.

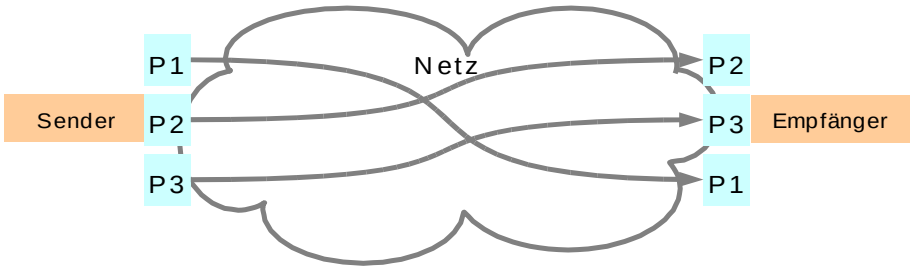

Abbildung 11.3: Verbindungslose Kommunikation

11.2.1 Verbindungsorientierte Verfahren

Eine genauere Betrachtung der verbindungsorientierten Protokolle zeigt, dass hier auf einem realen Netz mit Übertragungsleitungen und Schnittstellen-Computern eine virtuelle Verbindung aufgebaut wird,

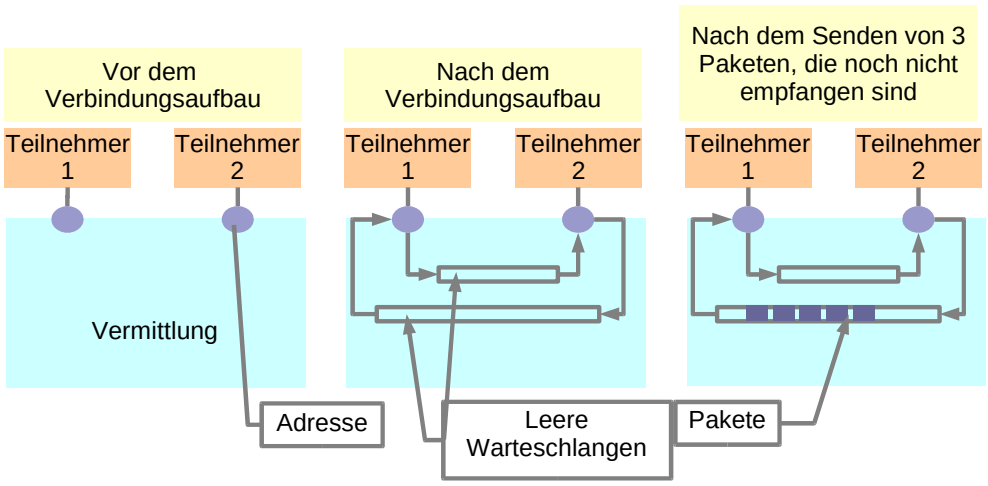

Abbildung 11.4: Verbindungsorientierte Kommunikation

die während der Dauer der Datenübertragung bestehen bleibt und anschliessend wieder abgebaut werden muss, damit andere Benutzer die Einrichtungen wieder benutzen können. Typische Beispiele aus der Praxis sind das öffentliche Telefonnetz, das ISDN-Netz, das X.25-Netz, Frame Relay und ATM-Netze.

11.2.1.1 Leitungsvermittlung im Telefon- oder ISDN-Netz

In öffentlichen Netzen (Telefon, ISDN) wird das Vermittlungsverfahren der Leitungsvermittlung eingesetzt (cirquit switching).

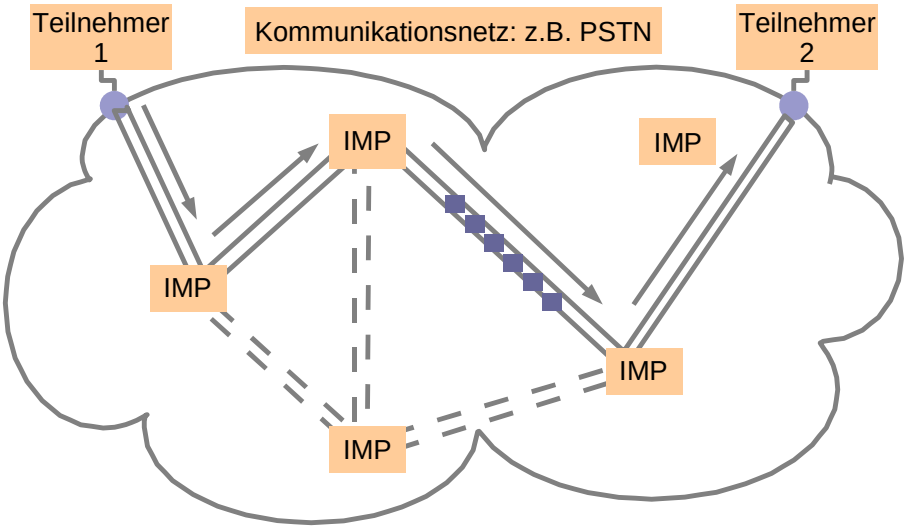

Abbildung 11.5: Leitungsvermittlung am Beispiel PSTN

Die Leitungsvermittlung besteht darin, dass vor der Datenübertragung eine virtuelle Verbindung auf physikalischen Leitungen und in den Knoten bereitgestellt (aufgebaut) wird. Dies geschieht dadurch, dass der Sender die Telefonnummer des Empfängers wählt und damit die Zentralen veranlasst, die Leitung durchzuschalten und Speicher in den Knoten bereitzustellen. Die beiden Teilnehmer (Benutzer) der Leitung senden ihre Daten über diese Leitung und wenn sie damit fertig sind, wird die Leitung wieder abgebaut und die Ressourcen (Speicher, Leitweg-Informationen …) in den Knoten freigegeben.

11.2.1.2 Paketvermittlung mit Virtual Circuits (VC)

Die Paketvermittlung (virtual circuit packet switching) basiert darauf, dass ein Teilnehmer an seiner lokalen Paketvermittlungsstelle (PSE, Packet Switching Exchange) Pakete ins Netz speist. Die Pakete haben eine Adresse und werden auf verschiedenen Wegen über das Netz dem Empfänger zugestellt. (I = Teilinformation, As = Adresse des Senders, Ae = Adresse des Empfängers).

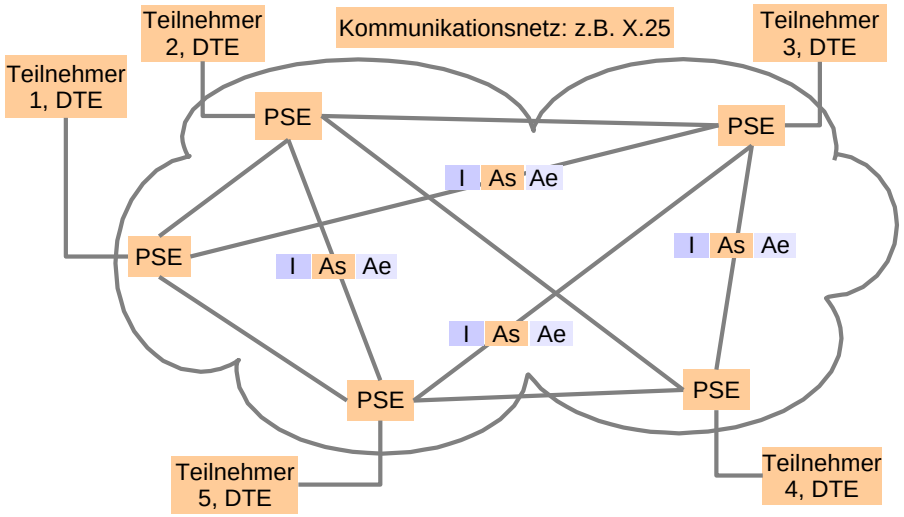

Abbildung 11.6: Paketvermittlung am Beispiel X.25

11.2.1.3 Die Paketgrösse

Die Paketgrösse spielt hier eine Rolle. Kleinere Pakete erlauben vermehrtes Sammeln der Meldungen an den Knoten und haben kleinere Absendeverzögerungen, weil sie schneller verpackt sind. Aber kleinere Pakete haben im Verhältnis zu den Nutzdaten einen grösseren Header (Nachrichtenkopf mit Steuerinformation). Abbildung 11.7 zeigt einen Vergleich zwischen verschiedenen Paketgrössen:

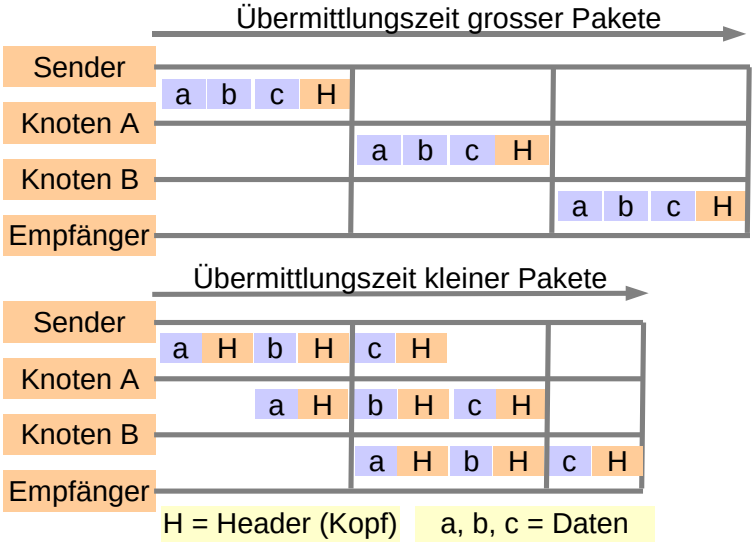

Abbildung 11.7: Der Unterschied in der Übermittlungszeit der Pakete

Kurze Pakete können schneller gesendet werden, da während der Übertragung der Daten b von Sender zum Knoten A bereits die Daten a von Knoten A zum Knoten B übertragen werden können.

11.2.1.4 Der kürzeste Pfad (Routing, Leitwege)

Bei den Virtual Circuits (VC) wird ein Leitweg vor dem Senden der Pakete ausgehandelt und dann auch benutzt. Jeder Knoten muss also wissen, wie ein Paket geroutet (gelenkt) werden soll. Nach der Benutzung wird der Leitweg (route) wieder gelöscht. Dijkstra (1959) hat ein Verfahren zur Bestimmung des kürzesten Pfades entwickelt.

Eine Nachricht soll auf dem folgenden Netz von A nach D übermittelt werden. Die Pfade im Netz haben verschiedene Längen und somit werden verschiedene Leitungskosten anfallen. Ein Netzbetreiber möchte natürlich möglichst geringe Leitungskosten haben, um einen möglichst grossen Ertrag an der Übertragung der Daten zu haben.

Das Problem besteht darin, wie man die einzelnen Knoten programmieren muss, damit sie die Daten auf dem kürzesten Weg übertragen. (Engpässe auf dem Netz werden in unserem Beispiel nicht betrachtet. Diese stellen dann noch eine weitere Schwierigkeit dar.)

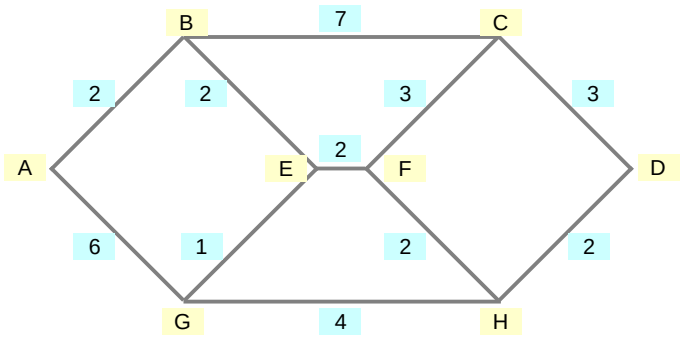

Abbildung 11.8: Ausgangslage

a) Beginnen wir bei A und untersuchen wir für Knoten A, welches die Abstände der nächsten Knoten sind. Beschriften wir B mit (2,A), was bedeutet, dass B von A den Abstand 2 hat. G wird mit (6,A) beschriftet. Wir erkennen, dass B den kleinsten Abstand zu A hat, da B den Abstand 2 hat und G den Abstand 6.

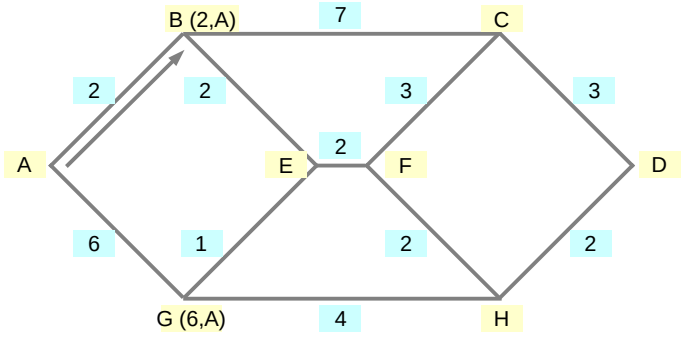

Abbildung 11.9: Situation nach dem 1. Schritt

b) Betrachten wir alle Knotenabstände zu B, also C und E, und markieren diese Knoten wie folgt: Bis zum Punkt C sind es zusätzlich

noch 7, d.h. die totale Entfernung ist somit 9 (9,B). Bis Punkt E ist die totale Entfernung von A 4. Die Beschriftung ist somit E (4,B).

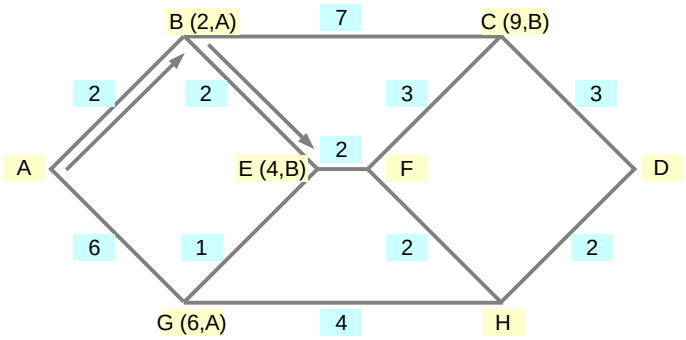

Abbildung 11.10: Situation nach dem 2. Schritt

c) Punkte um E: Wir sehen, dass sich G von G (6,A) auf G (5,E) ändert, weil der Weg über B und E kürzer ist, als von A nach G!

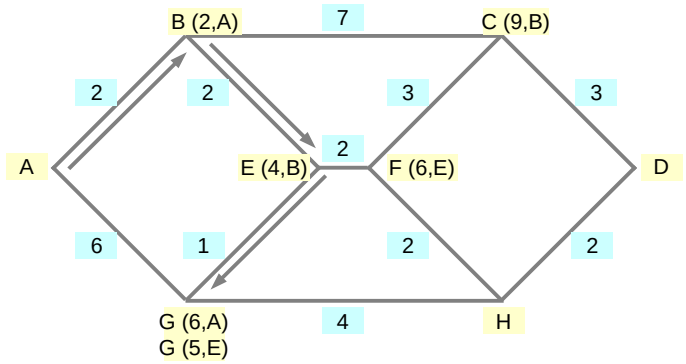

Abbildung 11.11: Situation nach dem 3. Schritt

d) Von G aus ergibt sich folgende Situation. Die zu untersuchenden Knoten sind F und H, weil E bereits untersucht wurde. Dies ergibt für H einen totalen Abstand von 9.

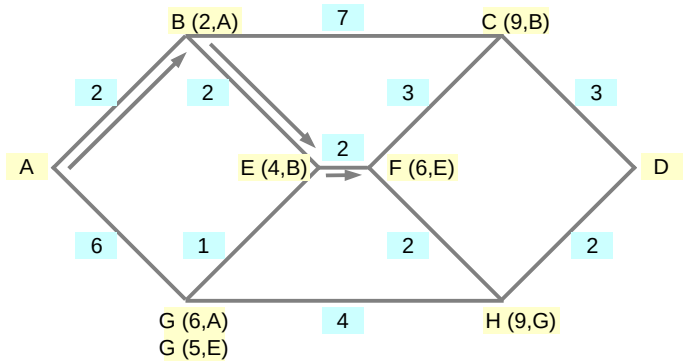

Abbildung 11.12: Korrektur im 4. Schritt

e) Für F kommen C und H in Frage. Für H ergibt sich nun aber ein totaler Abstand von 8, da E die Daten nach der letzten Erkenntnis nicht nach G senden wird, sondern direkt nach F.

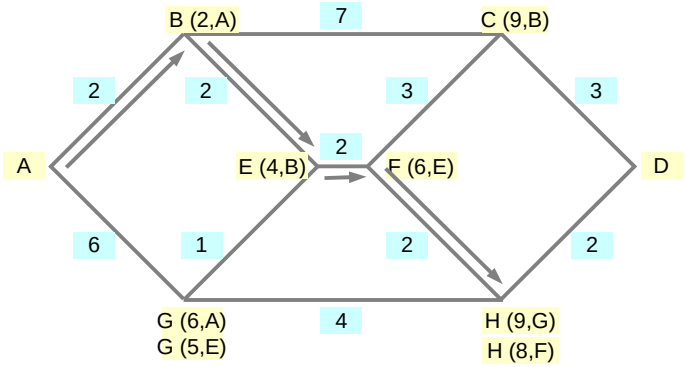

Abbildung 11.13: Situation nach dem 5. Schritt

f) H übermittelt die Daten nach D. Der total zurückgelegte Weg beträgt 10.

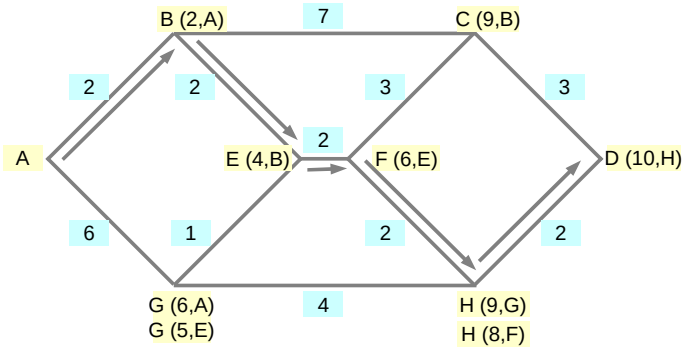

Abbildung 11.14: Der letzte Schritt

11.2.2 Verbindungslose Protokolle

Die Bestimmung von Leitwegen kann in grossen Netzen sehr aufwändig werden. Aus diesem Grund hat man Netze mit verbindungslosen Strategien entwickelt. In diesem Abschnitt werden die Nachrichteneinheiten in verbindungslosen Netzen, die Datagramme und ihre Vermittlung untersucht.

11.2.2.1 Datagramme

Datagramme heissen die unabhängigen Pakete der verbindungslosen Vermittlung. In einem Datagramm-Teilnetz werden keine Routen zum Voraus festgelegt, vielmehr leiten die Knotenrechner in diesen Netzen die Datagramme anhand ihrer Adressen auf dem kürzesten (günstigsten) Weg an ihren Bestimmungsort. So kann es geschehen, dass ein Datagramm von Zürich via Tokio und ein anderes der gleichen Nachricht via Boston nach Paris gelangt.

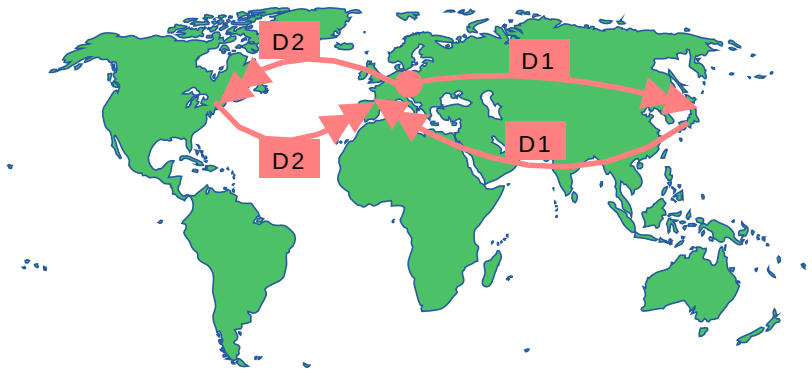

Abbildung 11.15: Vermittlung von Datagrammen im GAN

Eine typische Anwendung eines solchen Verfahrens wird im Internet und anderen TCP/IP-Netzen (UNIX-Welt) auf IP (Internet-Protokoll) basierend eingesetzt.

11.2.2.2 Die Vermittlung im IP (Internet-Protokoll)

Dieses verbindungsunabhängige Protokoll ist seit 1980 im Einsatz. Es beruht auf transparenten, wenn auch nicht immer zuverlässigen Internet-Datagrammen, die von der Quelle zum Host verschiedene Arten von Netzen durchqueren können. Abbildung 11.16 zeigt eine solche Situation:

Das IP-Protokoll arbeitet wie folgt. Die Teilnachricht (TN) wird von der darüberliegenden Transportschicht in 64-KByte-Datagrammen inklusive Transport-Nachrichtenkopf (TK) übernommen. Alle Datagramme werden in der Vermittlungsschicht mit einer IP-Nummer versehen und das Ganze wird als Paket 1 durch das Netz 1 übertragen. Paket 1 wird im Router 1 in Paket 2 umgepackt und über das Netz 2 zum Router 2 übermittelt. Dieser erstellt Paket 3 und sendet das Ganze an den Empfänger. Im Empfänger werden die Datagramme wieder vereinigt, die Teilnachrichten ausgepackt und an die Transportschicht (Layer 4) übergeben. Die Datagramme können unterwegs in kleinere Datagramme zerlegt werden.

Das Datagramm besteht aus einem IP-Kopf mit mindestens 20 Bytes, einem Transport-Nachrichtenkopf (TK) und einem Datenteil (TN). Als Besonderheit wird die Versionsnummer des Protokolls mitgesendet. Falls im Netz ein Rechner mit einem älteren oder neueren IP-Protokoll vorhanden ist, so kann das Datagramm trotzdem transportiert werden. Der Kopf enthält noch andere Angaben, wie z.B. die Angabe über die Länge des Kopfes und eine Identifikationsnummer für allfällige Fragmente eines Datagrammes.

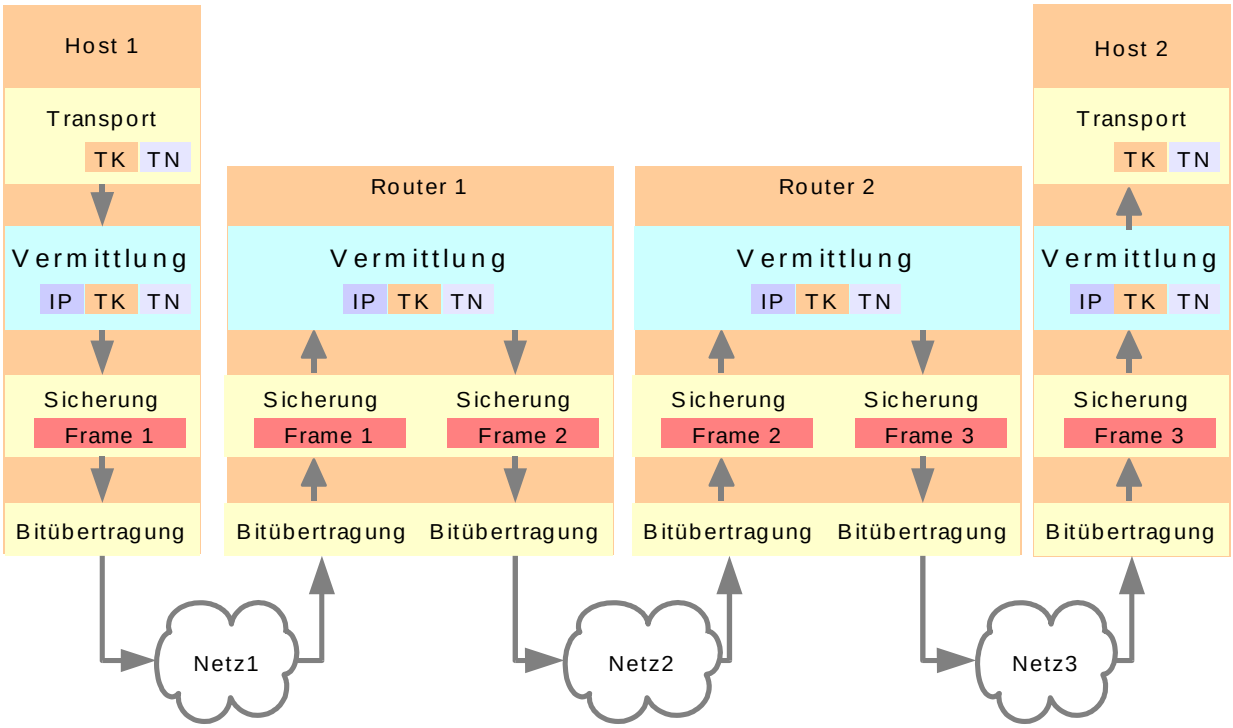

Abbildung 11.16: Vermittlung im IP-Netz

Die Datagramme werden von der Schicht 2 (Verbindungsschicht) mit den Sicherungsnachrichtenköpfen (SK1 für Netz 1, SK2 für Netz 2 und SK3 für Netz 3) und dem Sicherungsnachspann (SN1, SN2 und SN3) in netzabhängige Frames eingepackt. Dies erlaubt das Versenden von Frames über alle existierenden Netzarten (Ethernet, Token Ring, ATM, X.25 …).

11.2.3 Vergleich zwischen Leitungsvermittlung, VC und Datagrammen

Abbildung 11.17 zeigt den Verbindungsaufbau der Leitungsvermittlung, der Paketvermittlung mit VCs und der Paketvermittlung mit Datagrammen im Vergleich. Der Sender (S) sendet in allen drei Fällen die gleichen Daten a, b und c über die Knoten (Kn.A) und (Kn.B) zum Empfänger (E).

Datagramme brauchen keinen Verbindungsaufbau und sind somit für einzelne kurze Meldungen am effizientesten.

Virtual Circuits erlauben eine einfachere Verteilung der Verkehrslast und halten die Paketsequenz ein.

Die Effizienz (Durchsatz, Throughput) hängt wesentlich von der Topologie und der Grösse des Netzes sowie vom Verkehrsverhalten der Nachrichtenquellen ab.

Tabelle 11.1 vergleicht verschiedene Diskussionspunkte in Teilnetzen mit Datagrammen und solchen mit Vcs:

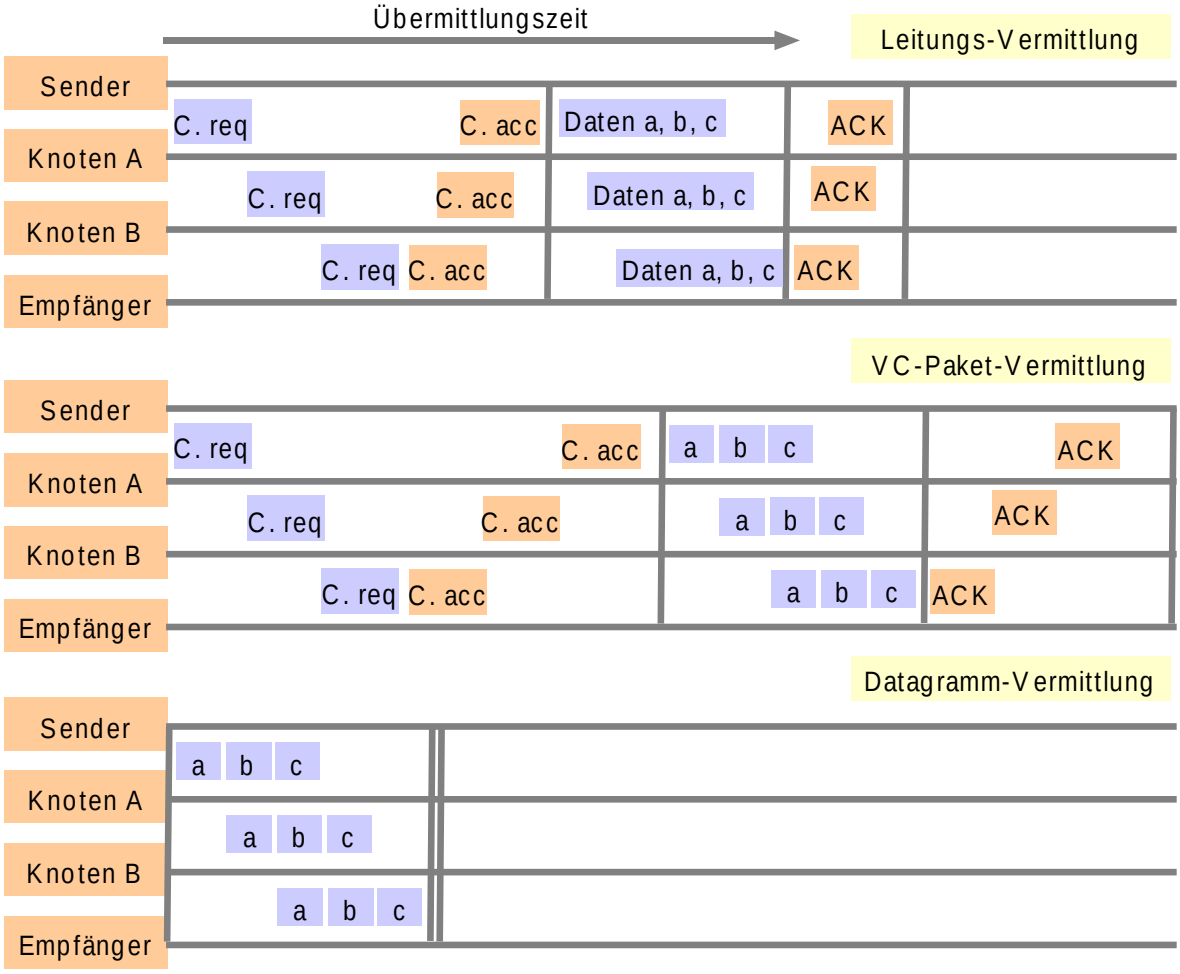

Abbildung 11.17: Vergleich der Vermittlungsarten

Diskussionspunkt	*Datagramme*	*Virtuelle Verbindungen*
Verbindungsaufbau	Nicht erforderlich	Erforderlich
Adressieren	Jedes Paket enthält die volle Quell- und Zieladresse	Jedes Paket enthält eine Nummer der virtuellen Verbindung
Statusinformationen	Das Teilnetz muss keine Status-informationen führen	Für jede Verbindung ist ein Tabelleneintrag erforderlich
Routing	Jedes Paket wird unabhängig be-fördert	Die Route wird beim Aufbau der virtuellen Verbindung gewählt; alle Pakete folgen diesem Leitweg
Wirkung von Router-fehlern	Keine, ausser dass Pakete verloren gehen und noch einmal gesendet werden müssen	Alle virtuellen Verbindungen über den ausgefallenen Router werden beendet
Überlastungsüber-wachung	Schwierig	Einfach, wenn im Voraus für jede virtuelle Verbindung ausreichend Puffer bereitgestellt werden.

Tabelle 11.1: Vergleich zwischen Datagrammen und VC

11.3 Aufgaben

1. Welches sind die Hauptfunktionen des 3. Layers im ISO/OSI-Modell?
2. Nach welcher Art werden bei folgenden Netzwerken die Verbindungen aufgebaut, verbindungslos oder verbindungsorientiert? ISDN, PSTN, ATM, IP, Frame Relay, X.25
3. Was verstehen Sie unter Routing?
4. Welches Routing-Verfahren wurde von Dijkstra entwickelt?
5. Erklären Sie das Routing-Verfahren von Dijkstra kurz.
6. Hat dieses Verfahren auch Nachteile?
7. Wie funktioniert der Verbindungsaufbau in einem X.25-Netz?
8. Wozu braucht ein IP-Paket Quell- und Ziel-Adresse? (Eigentlich würde doch die Ziel-Adresse ausreichen.)
9. Welches sind die unterschiedlichen Wirkungen von Router-Fehlern bei Datagrammen und bei virtuellen Verbindungen?
10. Warum braucht es bei Datagrammen keinen Verbindungsaufbau?

Lösungen unter www.sauerlaender.ch/downloads

12 Sicherung der Nachrichten

Leider gehen immer wieder Nachrichten verloren oder werden nicht empfangen. Um die Nachrichten trotzdem vollständig übertragen zu können, werden in den Netzen Sicherungsmethoden eingesetzt.
Das vorliegende Kapitel beschreibt solche grundsätzlichen Sicherungsmethoden.

12.1 Dienste, Schnittstellen und Protokolle

Es wird Zeit, an dieser Stelle die drei Begriffe Dienst, Schnittstelle und Protokoll im ISO/OSI-Modell genauer zu erklären, bevor die Sicherung der Nachrichten auf der Schicht vier genauer erläutert wird.
Ein *Dienst* wird einer im ISO/OSI-Modell weiter oben liegenden Schicht grundsätzlich von der unteren Schicht über eine *Schnittstelle* zur Verfügung gestellt.
Protokolle hingegen sind Abmachungen (Regelgefüge), auf deren Basis die Einheiten ihre Dienste definieren. Man kann die Protokolle beliebig ändern, so lange man nicht die für die Dienstnutzer sichtbaren Dienste ändert.

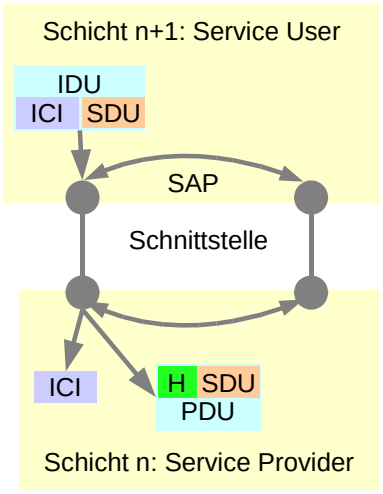

Abbildung 12.1: Zusammenhang zwischen Dienst, Schnittstelle und Protokoll

Eine Protokolldateneinheit (Protocol Data Unit, PDU) in Schicht n beinhaltet neben einem Protokollkopf (Header, H) eine Dienstdateneinheit (Service Data Unit, SDU). Eine Schicht weiter oben benötigt einen bestimmten Dienst einer weiter unten liegenden Schicht. Schicht n+1 besitzt eine Schnittstellendateneinheit (Interface Data Unit, IDU), der bekannt ist, über welche Schnittstellensteuerdaten (Interface Control Information, ICI) sie die benötigten SDU von der unteren Schicht bekommt. Über einen Dienstzugriffspunkt (Service Ac-

cess Point, SAP) und der dazugehörenden Schnittstelle kann die Schicht n+1 den Dienst beantragen. Schicht n erkennt anhand der ICI und SDU, welcher Dienst angefordert wird und kann über den umgekehrten Weg den Dienst bereitstellen.

12.2 Aufbau, Betrieb und Abbau von Verbindungen

Die Transportschicht (Layer 4) ist die letzte der unteren Schichten im ISO/OSI-Modell, die noch direkt mit dem Datentransport auf dem Netz zu tun hat. Die höheren Schichten sind anwendungsorientierte Schichten. Die Schicht vier dient vor allem dem geordneten Aufbau, Betrieb und Abbau der Netzwerk-Verbindung. Dass vor allem der Abbau nicht immer gesichert werden kann, wird weiter hinten dargelegt.

12.2.1 Zweck

Für die Übertragung von Daten haben wir bis jetzt schon einiges bereitgestellt. Es ist uns bereits möglich, Daten in Rahmen zu verpacken und diese so abzusichern, dass Sender und Empfänger eine Chance haben, die Richtigkeit zu überprüfen und allenfalls zu korrigieren. Diese Rahmen können wir bereits in Pakete verpacken und auf verschiedene Arten einer Gegenstelle übermitteln.

Doch was geschieht, wenn Datenpakete verloren gehen oder ein Router defekt ist? Die Vermittlungsschicht würde solche Fehler je nach Protokollvariante herausfinden und die Verbindung unterbrechen. Doch wer soll diese Vorfälle den höheren Schichten mitteilen?

Praxis-Hinweis:
Was geschieht, wenn Router defekt sind, oder Datenpakete aus anderen Gründen nicht weitergeleitet werden können?
Die Schicht 4 stellt die notwendigen Dienste zur Verfügung, um die höheren Schichten zu alarmieren und sichert somit den Aufbau, den Betrieb und den Abbau der Netzwerkverbindung.

Die höheren Schichten des ISO/OSI-Modelles sind anwendungsorientiert und müssen sich auf einen zuverlässigen Transportdienst der Daten im Netz verlassen können (Transport Service User). Die Schichten 1 bis 3 des ISO/OSI-Modelles befassen sich mit der Bereitstellung der Daten für das Übertragungsnetz, der Fehlerkorrektur und dem Vermitteln der Daten auf den verschiedenen Netzen. Diese Schichten sind eng an die Kommunikationsteilnetze gebunden und somit abhängig von den Netzbetreibern (Beispiel WANs mit teilweise proprietären Protokollvarianten). Die Transportprotokolle sind zwischen den höheren, rein anwendungsorientierten Schichten und den tieferen, netzwerkorientierten Schichten angesiedelt und sind somit für den zuverlässigen Transport der Daten vom Quellrechner zum Zielrechner verantwortlich. Würde die Transportschicht nicht zur Verfügung stehen, hätten die Benutzer keine Möglichkeit, verlorene Pakete, auf Grund unzuverlässiger Verbindungen oder ausgefallenen Routern, festzustellen. Normalerweise bauen die Transportprotokolle für jede Transportverbindung eine eigene Netzwerkverbindung auf, mit einer eigenen Transportadresse, die es der Transportschicht des Empfängers erlaubt, verlorene Pakete und verstümmelte Daten für

die höheren Schichten aufzubereiten. Die Schicht 4 ist somit die letzte Schicht, die diese Dienste bereitstellt und gehört somit zusammen mit den Schichten 1 bis 3 zu den Transport Service Providern.

12.2.2 Dienste, Dienstqualität und Dienstoperationen

Die Transportschicht stellt den höheren Schichten verbindungsorientierte oder verbindungslose Dienste für den Verbindungsaufbau, die Datenübertragung und den Verbindungsabbau bereit. Es werden dazu die Elemente „Adressierung“ (Transportadressen, Netzwerkadressen, NA), „Transport Protocol Data Units“ (TPDU), die „Flusssteuerung“, „Zwischenspeicher“ und „Multiplexing“ (siehe drei Pfeile beim Host 2)eingesetzt.

Damit das geschilderte Ziel erreicht werden kann, stützen die Dienste der Transportschicht auf den Diensten der Vermittlungsschicht ab. Die Hardware oder Software, die diese Aufgabe übernimmt, heisst Transportinstanz (TI). Diese Transportinstanz kann sich im Betriebssystem-Kernel (Kernstück des Betriebssystems), in einem Benutzerprozess, in einem Netzwerk-Bibliothekspaket (siehe Theorie der Betriebssysteme) oder auf der Netzwerk-Schnittstellenkarte befinden.

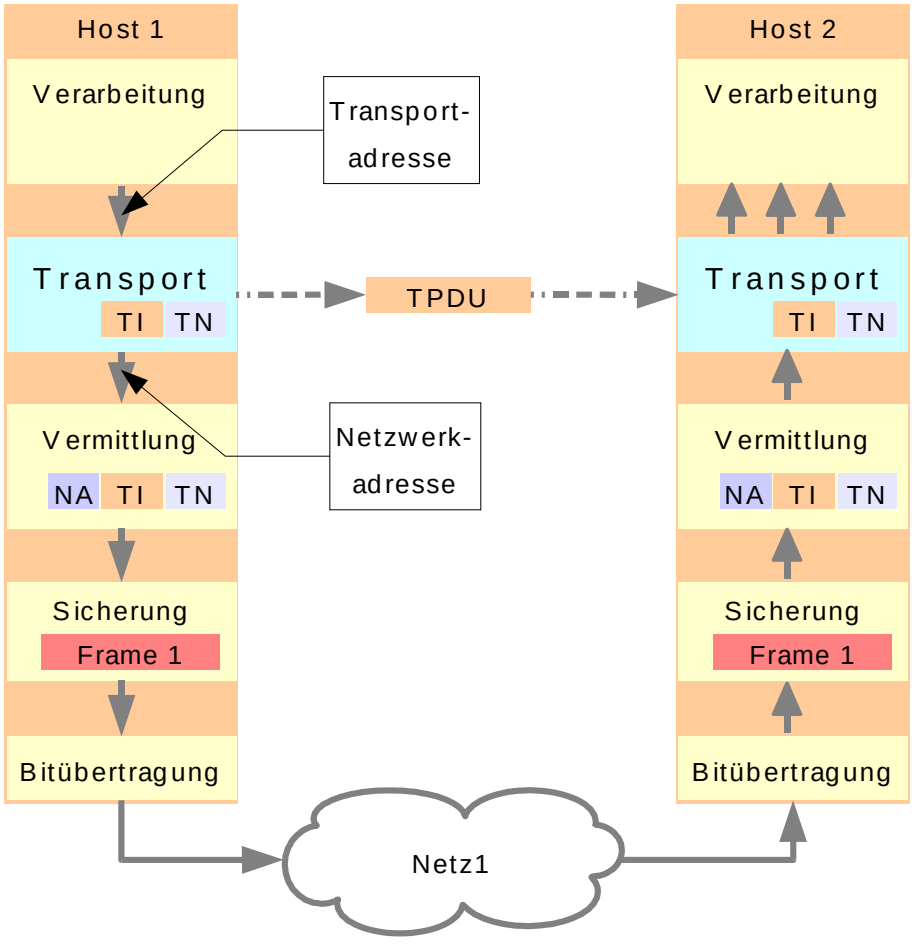

Abbildung 12.2: Die Funktion des Layers 4

12.2.3 Dienstqualitäten

Die Dienstqualitäten der Vermittlungsschicht (QoS, Quality of Service) werden durch die Transportschicht auf das vom Benutzer gewünschte Niveau angehoben. Die Transportschicht kann die Mängel von schlecht arbeitenden Vermittlungsschichten ausgleichen.
Die typischen Parameter für die Dienstqualität auf der Transportschicht sind:

- Dauer des Verbindungsaufbaues (Connection Establishment Delay). Je kürzer die Verzögerung beim Aufbau der Verbindung, desto besser der Dienst.
- Ausfallwahrscheinlichkeit beim Verbindungsaufbau (Connection Establishment Failure Probability). Bei einer Netzüberlastung kann eine Verbindung nicht innerhalb einer festgelegten Zeit aufgebaut werden.
- Durchsatz (Throughput). Es wird in kurzen Abständen in jede Richtung gemessen, wie viele Benutzerbytes pro Sekunde übertragen werden.
- Übertragungsverzögerung (Transit Delay). Es wird in jede Richtung gemessen, wie lange die Übertragung einer Nachricht von der Transportinstanz des Senders bis zur Transportinstanz des Empfängers dauert.
- Restfehlerrate (Residual Error Ratio). Diese misst die Anzahl zerstörter oder verlorener Nachrichten im Verhältnis zur gesamten Anzahl an versendeten Nachrichten.
- Schutz (Protection). Der Benutzer kann angeben, dass er seine Daten vor unerlaubtem Lesen oder Verändern durch Unbefugte (Hacker, Eindringlinge) schützen will.
- Priorität (Priority). Stuft die Verbindung als wichtig ein. Dies ist bei überlasteten Netzen von Bedeutung.
- Störungsausgleichsverhalten (Resilience). Dieser Parameter definiert, wie wahrscheinlich es ist, dass eine Transportschicht die Verbindung im Falle von Überlastung oder internen Problemen spontan beendet.

12.2.4 Dienstoperationen

Die Dienstoperationen ermöglichen den Benutzern dieser Schicht (z.B. Anwendungsprogrammen) den Zugriff auf den Transportdienst. Im Prinzip werden diese Operationen ähnlich genutzt wie die Steueranweisungen in den HDLC-Frames. Die Operatoren werden wiederum in den Nutzdaten eingekapselt und übertragen.
Abbildung 12.3 zeigt die Zusammenhänge des Einkapselns.
Ein Teil der Nutzdaten (TN) wird mit dem TPDU-L4-Header (L4 H) gekapselt und stellt somit die Paket-Nutzdaten in Layer 3 dar. Dort

werden die Frame-Nutzdaten zusammengestellt, indem die L3-Headerinformationen angehängt werden.

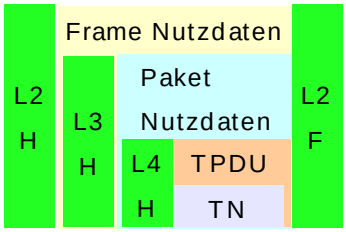

Abbildung 12.3: Das Bereitstellen der TPDU

Eine sehr einfache Transportschnittstelle kann zum Aufbau, Betrieb und Abbau beispielsweise die folgenden fünf Operatoren haben:

- LISTEN, zum Blockieren der Kommunikationseinrichtung, bis ein Prozess versucht, eine Verbindung aufzubauen.
- CONNECT, für den aktiven Versuch, eine Verbindung aufzubauen. Im Nutzdatenfeld der TPDU wird ein CONNECTION REQ. übertragen.
- SEND, um die Informationen zu senden. In der TPDU werden Daten übertragen mit dem Operator DATA.
- RECEIVE blockiert den Prozess, bis TPDUs ankommen mit den Operatoren DATA.
- DISCONNECT wird benutzt, um die Verbindung wieder abzubauen. In der TPDU wird der Operator DISCONNECT REQ. übertragen.

In UNIX heissen diese Operationen „Socket-Operationen“. Zu den oben genannten kommen noch einige zusätzliche hinzu, welche für die IP-basierenden TCP-Verbindungen wichtig sind. An dieser Stelle wollen wir diese Spezialitäten aber nicht abhandeln, da dies den Rahmen des vorliegenden Werkes deutlich sprengen würde.

12.2.5 Elemente der Transportschicht

Die Transportschicht benutzt in ihrem Protokoll ähnliche Elemente wie die Sicherungsschicht (Schicht 2). So werden auch hier Adressierung, Fehlerüberwachung, Folgesteuerung (von Paketen), Flusssteuerung, Zwischenspeicherung und Multiplexing eingesetzt.

Die Adressierung erfolgt über die Transport Service Access Points (TSAP), Transportadressen. Die analogen Adressen auf der Vermittlungsschicht heissen Network Service Access Points (NSAP), Netzadressen. TSAP und NSAP erscheinen in allen Netzen immer in Paaren. So sind im IP-Netz die IP-Adressen den NSAPs zuzuordnen und die TSAPs sind die lokalen „Ports“.

Abbildung 12.2 zeigt beim Host 2 bereits, wie eine Transportschicht der Verarbeitungsschicht mehrere TSAPs zur Verfügung stellen kann (Multiplexen). So können verschiedene Serverprozesse durch ein und dieselbe Transportinstanz aus nur einem Netz beliefert werden. Diese erklärt, weshalb die Transportinstanz eines Webservers die verschiedenen Dienste wie FTP, HTTP einem Benutzer aus dem Netz gleichzeitig zur Verfügung stellen und nebenbei noch virtuelle Verbindungen zu anderen Benutzern (durch Multiplexing) aufrecht erhalten kann.
Die Elemente Flusssteuerung und Zwischenspeicherung dienen einer geschickten Steuerung der Netzauslastung. Unzuverlässig arbeitende Vermittlugsschichten benötigen Pufferspeicher für die TPDUs, damit, ähnlich dem Mechanismus in der Sicherungsschicht, verloren gegangene Pakete aus den Puffern wieder angefordert werden können. Die Flusssteuerung und die Folgesteuerung sorgen dafür, dass die Speicher nie überlaufen und die TPDUs in der richtigen Reihenfolge an die Verarbeitungsschicht übergeben werden können.

12.3 Standards der Transportschicht

Für die in diesem Kapitel besprochenen Transportprotokolle existieren die folgenden Standards:

- ISO Transport Class 4 (ISO TOP Network)
- TCP und UDP (ARPA Network)
- SPX (Novell Netware)

12.3.1 ISO TOP

Dieses Protokoll geht davon aus, dass die Vermittlungsschicht (3) nicht ganz zuverlässig ist und übernimmt die gesamte Fehlerbehandlung und Flusssteuerung selbst. Mit diesem Protokoll können fast alle Netzwerkkarten verbunden werden. Der Aufwand dafür sind komplizierte Transportprotokolle, die einen unzuverlässigen Vermittlungsdienst kompensieren müssen.

12.3.2 TCP und UDP

Das TCP (Transmission Control Protocol) ist zusammen mit dem UDP (User Datagram Protocol) wie IP (Internet Protocol) integraler Bestandteil des ARPA-Netzwerkes. Im Gegensatz zu IP, das mit Host-Adressen arbeitet (d.h. mit System-Adressen), ermöglichen TCP und UDP eine Kommunikation zwischen Prozessen beteiligter Systeme (also zwischen Verarbeitungsinstanzen) über Ports. UDP erlaubt eine ungesicherte Übertragung ohne permanente Verbindung. TCP ist verbindungsorientiert und ermöglicht eine gesicherte Verbindung mit gepufferter Datenübertragung. Es ist duplex-fähig.

12.3.3 SPX

SPX (Sequenced Packet Exchange) ist das Pendant zum TCP von Novell. Dieses Protokoll ist proprietär.

12.3.4 Transmission Control und NetBEUI

Dies sind die Protokolle der IBM-SNA-Netze (Systems Network Architecture), wobei NetBEUI (NetBIOS Extended User Interface) zusammen mit dem Layer 6-Protokoll NetBIOS (Network Binary Input Output System) in kleinen LAN grosse Verbreitung gefunden hat. NetBEUI ist nicht ganz innerhalb Layer 4 anzusiedeln, da es noch Aufgaben aus der Sitzungsschicht übernimmt und eigentlich ein Subprotokoll des NetBIOS ist.

12.4 Abbau der Verbindung – eine heikle Sache

Der Abbau der Verbindung auf Layer 4 ist nicht ganz unproblematisch. Es besteht keine direkte Verbindung zwischen den Teilnehmern wie beim Layer 2 (Punkt-Punkt-Verbindung). Ob eine Verbindung in den dazwischenliegenden Netzen immer restlos abgebaut wurde, kann von den Teilnehmern der Kommunikation nicht immer mit Bestimmtheit garantiert werden. Das kann zu unabgebauten Verbindungsteilen und somit zum Verlust von Kapazitäten führen. Spezielle Mechanismen (z.B. Timeouts) verhindern dies jedoch.

12.5 Aufgaben

1. Problematik des Abbaus von Verbindungen.
   Diskutieren Sie eine Lösung für das folgende Problem (machen Sie eine Skizze):
   Zwei Armeen stehen sich vor einer entscheidenden Schlacht gegenüber. Die blaue Armee hat die weisse Armee in einem Tal eingekesselt und steht mit je 200 Mann auf den gegenüberliegenden Hügeln. Die weisse Armee im Tal hat 300 Krieger. Wenn nun die blaue Armee gleichzeitig angreift, dann ist die weisse Armee in der Unterzahl und verliert die Schlacht. Im anderen Fall ist die Hälfte der blauen Armee in der Unterzahl. Der Feldherr der blauen Armee möchte daher, dass sein Stellvertreter auf der anderen Seite des Tals zum gleichen Zeitpunkt angreift wie er und schickt einen Meldeläufer durch das feindliche Gebiet zu ihm. Kann er sich sicher sein, dass der Stellvertreter die Nachricht erhält und synchronisiert angreift? Zusatzaufgabe: Erklären Sie die Analogie mit dem Abbau von Verbindungen im Netzwerk.
2. In einem Netz gehen Datenpakete verloren. Die Vermittlungsschicht stellt diesen Verlust fest und fordert neue Pakete an. Weshalb wissen die weiter oben liegenden Schichten davon und wie wird verhindert, dass die Pakete doppelt verwendet werden? Bitte antworten Sie in wenigen Sätzen.
3. Erklären Sie bitte in wenigen Sätzen die Funktion der Transport Protocol Data Units (TPDU).
4. Weshalb ist das Multiplexing von Diensten im Layer 4 eine sinnvolle Einrichtung? Was kann damit erreicht werden? Antworten Sie in wenigen Sätzen.

Lösungen unter www.sauerlaender.ch/downloads

13 Anwendungsorientierte Funktionen

Bis jetzt haben Sie alle relevanten Dienste kennen gelernt, die eine gesicherte Nachrichten-Übertragung durch die unterschiedlichen Netze gewährleisten.
Die Nachrichten wurden von den Benutzern ursprünglich mithilfe bestimmter Applikationen erstellt. Die Empfänger sollten somit in der Lage sein, diese Nachrichten mit den entsprechenden Applikationen wieder zu lesen. Die Sender müssen somit der Nachricht Informationen zu den zu benutzenden Applikationen mitliefern.

13.1 Aufgaben der Sitzungsprotokolle (Schicht 5)

Bei der Besprechung der Transportprotokolle in Schicht 4 haben wir gesehen, dass die Schicht 4 bei einem Unterbruch oder einer Störung im Netz versucht, die Verbindung aufrecht zu erhalten. Ein korrekter Abbau der Verbindung zwischen den Endgeräten ist jedoch in einem solchen Fall nicht möglich, da eine unterbrochene Verbindung auch keine TPDUs mehr übertragen kann. Dies kann zu einer unerwünschten Netzüberlastung führen, wenn im Netz Kommunikationsreste von früheren Verbindungen „herumirren“ und die Kapazitäten der Netzknoten verschwenden. Damit dies verhindert werden kann, folgen nach den Transportprotokollen die Sitzungsprotokolle, welche die Aufgabe haben, einen logischen Kommunikationspfad (session connection) zwischen den Applikationen der beiden Endgeräte aufzubauen, zu unterhalten und wieder abzubauen.
In den folgenden Abschnitten werden die 5 Aufgaben der Sitzungsschicht genauer erläutert. Die 5 Aufgaben sind:

- Synchronisation
- Aufbau, Betrieb und geordneter Abbau beim Datentausch
- Dialogverwaltung
- Aktivitätsverwaltung
- Ausnahmeberichterstattung.

Während einer Datenübertragungssitzung sind diese Protokolle dafür verantwortlich, dass die zwischen den beiden Verarbeitungsinstanzen (des Senders und des Empfängers) vereinbarten Betriebsparameter eingehalten werden.

13.1.1 Synchronisationspunkte für Recovery markieren

Die Transportprotokolle stellen zwar einen gesicherten Übertragungsweg zur Verfügung. Damit kann aber ein darüberliegender Prozess nicht sicher sein, dass er oder sein Partnerprozess zu jedem Zeitpunkt die ankommende Information korrekt verarbeiten kann.

Typische Beispiele dafür sind Hardwarefehler beim Schreiben in einen Massenspeicher oder die Blockierung eines Druckers. Das Sitzungsprotokoll bringt am Informationsfluss Synchronisationsmarken an, damit das Anwendungsprogramm des Benutzers im Falle einer Störung eine entsprechende Meldung (Speicher defekt, Papier ausgegangen …) erhält und der Fehler behoben werden kann (manuell oder automatisch). Mithilfe der Sysnchronisationsmarken kann die Wiederaufnahme (Recovery) der Kommunikation erreicht werden.

13.1.2 Aufbau, Betrieb und geordneter Abbau

Ähnlich wie bei den Transportprotokollen muss auch das Sitzungsprotokoll eine Sitzung aufbauen, betreiben und wieder abbauen. Der Unterschied zur Transportverbindung besteht darin, dass hier beim Abbau zuerst ein „REQUEST“ gesendet wird, um einen Datenverlust durch abruptes Trennen der Verbindung zu verhindern.

13.1.3 Die Dialogverwaltung

Viele Verbindungen arbeiten im Duplex-Betrieb. Nicht jede Software kann diesen Betrieb unterstützen.

Beispiel: Bei Anfragen auf einer Datenbank einer Fluggesellschaft macht es wenig Sinn, wenn von einem Terminal aus fünf Fragen gleichzeitig auf dem Host plaziert werden, ohne vorher die einzelnen Antworten vom System zu kennen. Damit ein Dialog mit der Datenbank aber trotz Halb-Duplex-Betrieb geordnet ablaufen kann, wird eine Dialogverwaltung benötigt.

Bei der Dialogverwaltung werden verschiedene Rechte durch Tokens (Marken) verwaltet. Nur der Partner, der das jeweilige Recht (Token) besitzt, kann entsprechende zugehörige Aktionen ausführen. Der Partner, der das Token nicht besitzt, kann es durch S-TOKEN-PLEASE.request anfordern. Tokens werden durch S-TOKEN-GIVE oder S-CONTROL-GIVE übergeben.

13.1.4 Aktivitätsverwaltung

Das Prinzip der Aktivitätsverwaltung besteht darin, dass der Benutzer den Datenstrom vorteilhafterweise in Einheiten einteilt und mithilfe dieser Einteilung den auf die Sitzungsschicht folgenden Protokollen die Möglichkeit gegeben wird, Transaktionen zu formen.

Die Einheiten werden zur Verwaltung der Transaktionen eingesetzt. Zu Beginn einer Einheit (S-ACTIVITY-START.request) setzt das Sitzungsprotokoll automatisch einen Major Synchronisation Request ab. Das Ende einer Activity wird durch ein S-ACTIVITY-END signalisiert.

Eine Activity kann vorzeitig beendet, zeitweise angehalten und wieder aufgenommen werden. Nach dem Anhalten können andere Daten gesendet werden. Aktivitäten erhalten Identifikationen.

13.1.5 Ausnahmeberichterstattung

Eine weitere Funktion der Sitzungsschicht ist die Berichterstattung im Falle eines aufgetretenen Fehlers. Gerät ein Anwender aus irgendwelchen Gründen in Schwierigkeiten (z.B. kein Papier mehr im Drucker), kann er dies der Gegenstelle mit einem S-U-EXCEPTION-REPORT.request mitteilen.

Praxis-Hinweis:
Mithilfe dieser Aktivitätsverwaltung wird der Mangel der Transportinstanz, eine Verbindung auch im Falle unterbrochener Verbindungen sauber zu beenden, kompensiert.

13.1.6 Standards der Sitzungsschicht

Die Hersteller von Netzwerksoftware implementieren diese Kommunikationssteuerungsprotokolle auf verschiedene Weise in ihren Programmen.

- Die ARPANET-Vertreter (TCP/IP) setzen dafür den Remote Procedure Call (RPC) ein.
- Novell rüstet seine Software mit dem Service Advertising Protokoll (SAP) aus.
- IBM verwendet für PC-LANs einen Teil des NetBEUI/NetBIOS und für die Grosssysteme den Data Flow Control.

13.2 Aufgaben der Darstellungsprotokolle (Schicht 6)

Die Darstellungsprotokolle haben die Aufgabe, die folgenden Probleme der Schicht 7 auszugleichen und damit der Schicht 5 weitgehend herstellerunabhängige, bereinigte Daten zur Verfügung zu stellen:

1. Alle in der Anwendungsschicht (Layer 7) implementierten Standards sind mit unterschiedlichen Programmiersprachen umgesetzt (Notation).
2. Die Daten aus Layer 7 haben unterschiedliche Dateistrukturen und Parameter (lokale Syntax, Sprache).
3. Schliesslich gibt es einige Regeln für die bitweise Darstellung von Datenstrukturen und Parametern (lokale Codierung). Das bedeutet, dass unterschiedliche Rechner auch unterschiedliche Datenformate für Buchstaben und Zahlen verwenden.
4. Die Chiffrierung und Dechiffrierung von Daten ist eine weitere Aufgabe dieser Schicht.
5. Die Kompression und Dekompression von Daten findet ebenfalls in Layer 6 statt.

13.2.1 Beispiel zur Aufgabe der Codierung

Stellvertretend für die Aufgaben der Darstellungsschicht betrachten wir eine Auswahl aus der Vielfalt der Zeichencodes. Es ist auch ganz nützlich, wenn wir einige der Zeichencodes kennen.

Internationales Alphabet Nr. 2	*Internationales Alphabet Nr. 5*	*EBCDIC Alphabet*	*PC Alphabet*
Auch bekannt unter dem Namen IA2	Auch bekannt unter den Namen IA5 und vor allem ASCII.	Firmenstandard von IBM.	Standard in der PC-Welt.
Der Code besteht aus fünf Bit. Dementsprechend werden nur Grossbuchstaben, Ziffern und einige Spezialzeichen codiert. Verwendung vor allem im Bereich Fernschreiber/ Telex. Wird immer weniger verwendet (Lochstreifen sind out).	Der Code besteht aus sieben Bit, wobei ein Bereich mit nationalen Sonderzeichen belegt werden kann. Es bestehen Versuche, den Code auf acht Bit zu erweitern. Verwendung im Bereich von Textterminals, zur Textübertragung etc. Sehr verbreitet.	Der Code besteht aus acht Bit. Er enthält ebenfalls nationale Sonderzeichen. Üblich in der IBM-Welt.	Der Code besteht aus acht Bit, wobei im 7-Bit-Bereich praktisch der ASCII-Zeichensatz übernommen wird. Enthält im oberen Bereich (über 128) unter anderem nationale Sonderzeichen, Grafikzeichen und einige griechische Buchstaben. Durch die Verwendung im PC-Bereich sehr verbreitet.

Tabelle 13.1: Zeichensätze[62]

13.2.2 Standards der Darstellungsschicht

Praxis-Hinweis:
Bei Novell sind diese Dienste im Netware Core Protokoll (NCP) untergebracht. Bei IBM sind diese Protokolle für PC-Netze im Net BIOS und für grosse Netze in den Presentation Services untergebracht.

Auch hier haben die verschiedenen Netzwerksoftware-Anbieter wieder unterschiedliche Lösungen vorgeschlagen:
ARPANET (TCP/IP) nennt diese Protokolle Lightweight-Presentation-Protokoll (LPP) und External Data Representation (XDR).

13.3 Aufgaben der Anwendungsprotokolle (Schicht 7)

Die Anwendungsprotokolle stellen die Verbindung zu den Anwenderprogrammen sicher. Mithilfe dieser Protokolle kann vermieden werden, dass sich ein Programmierer einer Applikation mit den Netzwerkfunktionen abgeben muss. Er muss lediglich die Schnittstelle zu den standardisierten Protokollen der Anwendungsschicht kennen und kann seine Programme darauf aufbauen.
Unterschiedliche Dateisysteme haben oft verschiedenartige Konventionen für Dateinamen oder zur Darstellung von Text; diese Inkompatibilitäten werden mittels der Anwendungsprotokolle behoben.
Beispielsweise gehört die Behandlung der elektronischen Post (E-Mail) in diese Protokollgruppe, aber auch die Funktion des Server Message Blocks (SMB) von IBM und andere vergleichbare Aufgaben.

62 EBCDIC: Extended Binary Coded Decimal Information Code
ASCII: American Standard Code for Information Interchange

Einige Protokolle sollen im Folgenden genannt werden. Die Liste kann nicht vollständig sein und wird auch laufend durch neue Protokoll-Varianten erweitert.

13.3.1 Beispiele von Protokollen (TCP/IP, ARPANET)

Die Protokolle des Application Layers sind von den Herstellern von Applikationssoftware oft in ganz spezieller Art und Weise implementiert worden. Dies hat zur Folge, dass eine ganze Vielzahl solcher Protokolle existiert. Einige der Protokolle wollen wir an dieser Stelle genauer betrachten.
Für TCP/IP-Netze gibt es folgende Protokolle:

Praxis-Hinweis:
Einige Firmen haben eigene Protokolle für ihre Netzwerkimplementationen geschrieben und es existieren daher eine ganze Vielzahl von Applikations-Protokollen. Neben den TCP/IP-ARPANET-Protokollen haben vor allem IBM, ISO, DEC, 3Com, Xerox, Apple und Banyan VINES solche Protokolle entwickelt.

13.3.1.1 FTP (File Transfer Protocol)

FTP ist dasjenige Protokoll, mit dessen Hilfe es möglich ist, Daten zwischen Rechnern im TCP/IP-Netz zu übertragen. Gleichnamige Programme von diversen Herstellern benutzen dieses Protokoll in ihren Applikationen.
Für das Herunterladen von Daten von einem Server über das TCP/IP-Netz müssen wir zuerst eine Verbindung vom Client zum Server herstellen. Die Applikation verlangt daher die IP-Adresse (oder den IP-Namen) des Servers. Normalerweise benötigt man ein Passwort, um auf die Dateien eines fremden Rechners (Server) zugreifen zu können. Einige Daten sind jedoch auf dem Internet frei erhältlich und es ist möglich, mit dem so genannten „anonymous FTP“ darauf zuzugreifen und herunterzuladen. Beim anonymen FTP müssen wir auf dem Client bei der Benutzeridentifikation das Wort „anonymous“ eingeben und bei der Passwortaufforderung geben wir unsere E-Mail-Adresse ein.

Praxis-Hinweis:
Die Protokolle setzen zum grossen Teil direkt auf dem TCP (Transmission control Protocol) oder dem UDP (User Datagram Protocol) auf.

Praxis-Hinweis:
Der Aufruf von FTP in einem kommandozeilenorientierten Betriebssystem lautet:
ftp <Internetadresse>

13.3.1.2 Telnet (Virtual Terminal Emulation)

Das Telnet-Protokoll wurde entwickelt, um von Remote-Rechnern aus auf Grossrechnern oder Unix-Servern Programme ausführen zu können. Das gleichnamige Programm benutzt dieses Protokoll. Auch hier müssen Sie eine Benutzeridentifikation und ein Passwort eingeben, um die Dienste des Hosts zu nutzen. Der Host muss über ein leistungsfähiges Multi-User-Betriebssystem verfügen.

Praxis-Hinweis:
Der Aufruf von Telnet in einem kommandozeilenorientierten Betriebssystem lautet:
telnet <Internetadresse>
oder
telnet hostname.ort.land

13.3.1.3 SMTP (Simple Mail Transfer Protocol)

Dies ist das Protokoll, das von den E-Mail-Programmen benutzt wird, um unsere elektronische Post auf dem TCP/IP-Netz zu versenden. Dem Protokoll liegen die Mailprotokolle des UNIX zu Grunde. Weil die Mail-Protokolle des UNIX mit eigenen acht-Bit-Codes arbeiten, müssen alle anderen Codes umgewandelt werden. Wird diese Umwandlung nicht durchgeführt, ist es beispielsweise nicht möglich, deutsche Umlaute per E-Mail zu übermitteln.

13.3.1.4 HTTP (HyperText Transfer Protocol)

HTTP ist das Kommunikationsprotokoll zwischen World Wide Web (WWW)-Servern und WWW-Clients.

Das HTTP übermittelt Hyper Text im HTML-Format (Hyper Text Markup Language).

Die Verbindung des Clients mit einem Server geschieht über eine TCP/IP-Verbindung.

Praxis-Hinweis:
Der Client kann auf dem Server Seiten im HTML-Format aufrufen, wenn er den URL (Uniform Resource Locator)
(z.B. www.acm.org) oder die Adresse
(z.B. 194.235.16.2)
der Seite kennt.

13.3.1.5 Protokolle für UNIX-Systeme

Für UNIX-Systeme existieren die folgenden Protokolle:

X-Windows-Systems (Betriebssystem Oberflächen-Schnittstelle)

lpr	Remote print
rcp	remote copy
rex ec	remote execution
login	remote login
rsh	remote shell
NFS	Network File System
UUCP	Unix to Unix Copy Program

Für die Steuerung oder die Verwaltung von TCP/IP-Netzen sind die folgenden Protokolle im Einsatz:

13.3.1.6 DNS (Domain Name System)

Für die Benutzer von TCP/IP-Netzen ist die IP-Adresse im Zahlenformat (xxx.xxx.xxx.xxx) nicht unbedingt brauchbar. Aus diesem Grund wurde das Domain Name System erfunden. Dieses System stellt einen Bezug her zwischen einer Adresse im Zahlenformat und dem besser verständlichen Namen im Textformat. So wird beispielsweise aus der IP-Adresse 194.235.xxx.xxx der Name im Textformat www.dus.ch.

13.3.1.7 SNMP (Simple Network Management Protocol)

Praxis-Hinweis:
Mit diesem Protokoll lassen sich alle Arten von Netzkomponenten überwachen. Hosts, Router, Bridges, Drucker und andere Geräte haben heute diese Protokolle in ihrer Firmware implementiert und können somit Statusinformationen an den Netzmanager senden.

Mit zunehmender Grösse der Netze stieg auch die Anfälligkeit auf Fehler. In TCP/IP-Netzen besteht eine Möglichkeit, nicht funktionierende Rechner von jedem Punkt aus im Netz ausfindig zu machen. Man sendet kleine Pakete an den fehlerhaften Rechner und wartet auf die Antwort. Gibt der Rechner Antwort, ist die Verbindung und der Rechner in Ordnung. Das Verfahren benötigt das Programm „Ping“.

Bald einmal war diese Lösung nicht mehr angemessen und man entwickelte ein eigenes Protokoll für die Netzwerk-Überwachung und -Verwaltung, das SNMP. Dieses Protokoll wurde in vielen kommerziellen Netzwerk Management Systemen eingebaut und hat heute eine grosse Verbreitung.

13.3.1.8 DHCP (Dynamic Host Configuration Protocol)

Das DHCP ermöglicht die Verwaltung der Netzwerkadressen in einem Netzwerk. Ein DHCP-Server verteilt eine bestimmte Anzahl

von IP-Adressen innerhalb eines Netzwerksegmentes dynamisch an die Stationen. Die Adressen sind also nicht mehr fest einer Station zugeordnet, sondern werden nur im Bedarfsfall einer Station ausgeliehen. Benutzt die Station das TCP/IP-Netz nicht mehr, gibt sie die Adresse dem DHCP-Server zurück.

13.3.1.9 NTP (Network Time Protocol)

Das NTP wird zur Zeitsteuerung in einem globalen Netzwerk benutzt.

13.3.1.10 RPC (Remote Procedure Calls)

Das RPC ermöglicht die Fernsteuerung eines externen Rechners. Die Remote Procedure Operationen basieren auf dem Client-Server-Modell. Das heisst, dass ein Client eine Anfrage an den Server sendet, eine Aktion abwartet und kontrolliert, ob die Anfrage fehlerfrei erkannt wurde.

13.3.1.11 Weitere Protokolle im Überblick

TFTP Trivial File Transfer Protocol
FTAM File Transfer Access und Management
VTS Virtual Terminal Service etc.
Die Liste ist nicht abschliessend, weil hier, wie bereits gesagt, sehr viele Protokolle existieren.

13.3.1.12 Beispiele von Protokollen (Novell)

Novell hat eigene Protokolle entwickelt: die NDS (Netware Directory Services) und das Netware Lite.

13.3.1.13 Beispiele von Protokollen (IBM-PC und IBM-SNA)

Der PC-LAN-Manager und diverse Remote Network Program Loader von IBM sind in vielen modernen PC-Betriebssystemen implementiert. Diese Protokolle sind direkt mit dem NetBIOS in Layer 6 verbunden. Für Grossrechner existieren unter dem Begriff SNA Transaction Services einige Netzwerkprotokolle, die direkt auf dem Data Flow Control in Layer 5 aufsetzen.

13.4 Die vollständige Kommunikation

Abbildung 13.1 zeigt eine Kommunikation über mehrere Stationen.
In der Bitübertragungsschicht muss jedes Gerät mit dem Nachbargerät physisch mit der richtigen Schnittstelle und Übertragungstechnologie (z.B. Ethernet, Twisted Pair-Kabel, RJ45-Stecker) verbunden sein. Im Layer 2 sind die Geräte über eine virtuelle Verbindung gekoppelt. Geräte, wie Repeater oder Modems, die keinen Layer 2 haben, merken somit nichts von den virtuellen Verbindungen anderer Geräte. Router unter sich kommunizieren über die virtuelle Verbindung in Layer 3. Hier gilt ebenfalls, dass alle Geräte ohne Layer 3

oder Geräte aus „fremden“ Netzen (hier ATM) nichts „merken“ von dieser Verbindung. Weiter oben liegende Schichten (sofern vorhanden!) kommunizieren somit direkt mit den Endpunkten zwischen Client und Server. Die Geräte dazwischen, die lediglich einen Aspekt der Kommunikation bereitstellen, merken davon nichts.

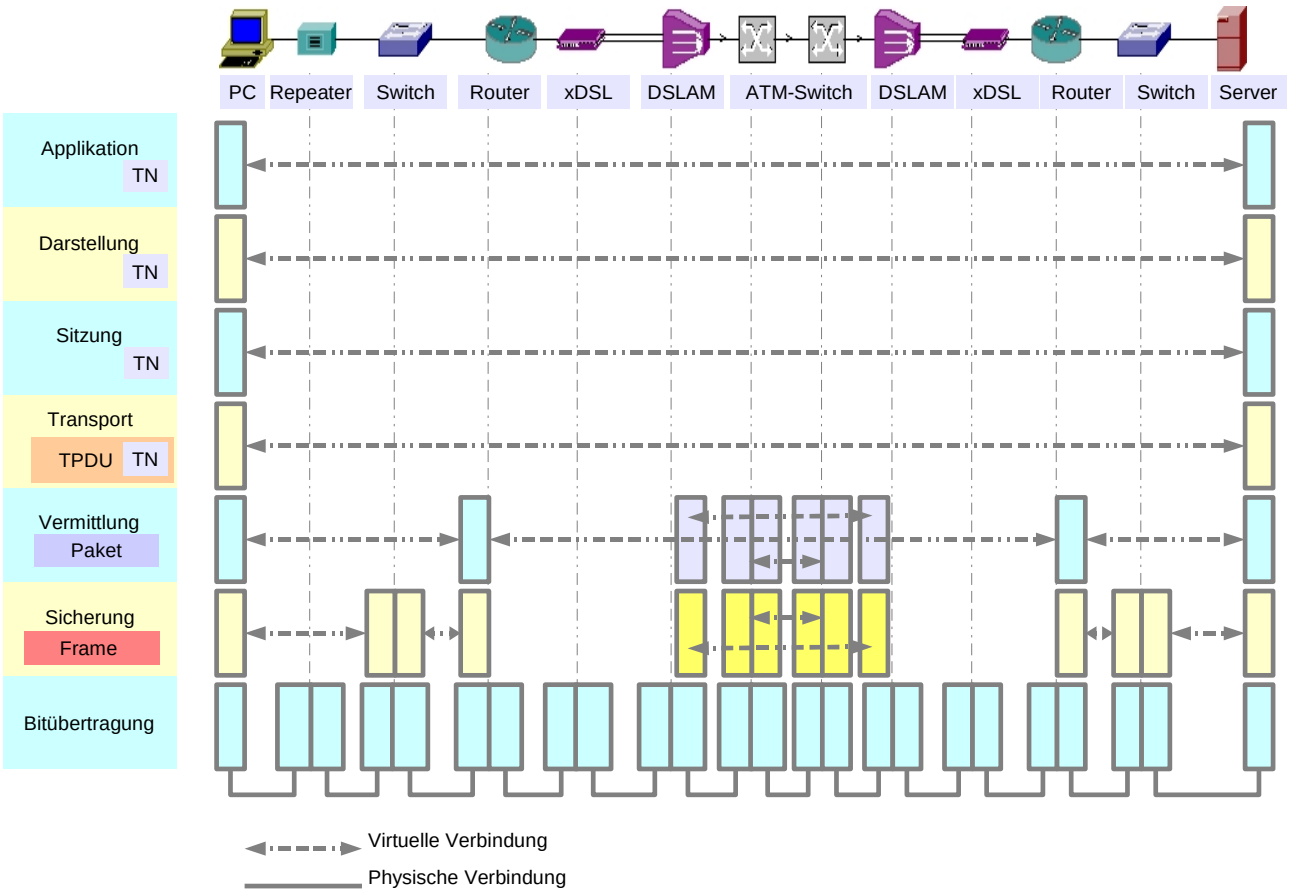

Abbildung 13.1: Eine Übertragungsstrecke (vereinfacht)

13.5 Aufgaben

1. Installieren Sie in einem Labornetzwerk einen Netzwerkscanner (z.B. Ethereal) und erfassen Sie einige Frames. Suchen Sie nach den bisher besprochenen Protokollnamen und Diensten. Erstellen Sie eine Liste mit den gefundenen Protokollnamen und geben Sie an, auf welcher Seite im Buch diese besprochen wurden. (Hinweis: Es ist verboten, ohne Erlaubnis ein produktives Netz zu scannen. Falls Ihr Labornetzwerk keine sinnvollen Frames aufweist, verwenden Sie die Unterlagen aus der Lösung.)
2. Wie stellen die Netzwerkscanner die Protokolle der verschiedenen Schichten dar? Zeigen Sie das anhand eines Beispielausdrucks und mit einem Satz als Erklärung.

Lösungen unter www.sauerlaender.ch/downloads