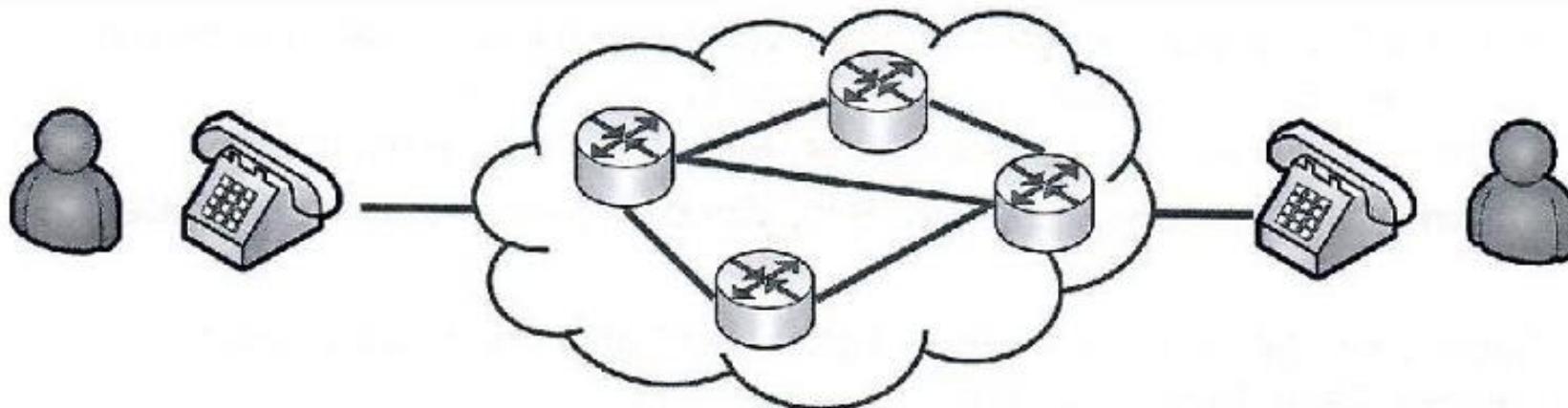




InternetKommunikation

Wie das vom Bachmeier Labermeier Yapmeier Quatschmeier bloß
besser ☺

2) Arten von Kommunikationsnetzen und –Diensten



- Digitale Telefonnetze (ISDN)
- Kabelfernsehen (CATV)
- Terrestrial Digital Video Broadcast network (DVB-T)
- Satellite communication network (SAT, DVB-S)
- Zellulare Netze (GSM, GPRS, UMTS, EDGE, HSDPA, LTE)
- Wireless LAN (WIFI, 802.11)
- Mobile Fernsehnetze (DVB-H, DMB)
- Power-Line Communication
- Fahrzeugkommunikation (LIN Bus, ...)
- Datennetze (Internet)
- Smart Grid
- Sensornetze

Quelle: [LKN]

3) Aufgaben von Kommunikationsnetzen

- Kommunikationsnetze erlauben einen vorübergehenden oder dauernden **Informationsaustausch** zwischen räumlich getrennten Kommunikationspartnern (**Teilnehmer**: Menschen oder Maschinen)
- **Information**: Sprache, Ton, Text, Bild, Video, Multimedia, (Sensor-)Daten,...
- Dabei erbringen Kommunikationsnetze den Teilnehmern einen oder mehrere **Teilnehmer-Dienste**
- Beispiel 1 (einfache Dienste):
 - A ruft B an
 - A lädt Datei M von Endsystem S
- Beispiel 2 (komplexe Dienste):
 - A unterhält eine Audio- und Videokonferenz mit B und C
 - Wenn B mich anruft, wenn ich gerade telefoniere (belegt bin), dann soll dieser Anruf zu C weitergeleitet werden
 - Finde D, welcher mir Information M zusenden kann

Quelle: [LKN]



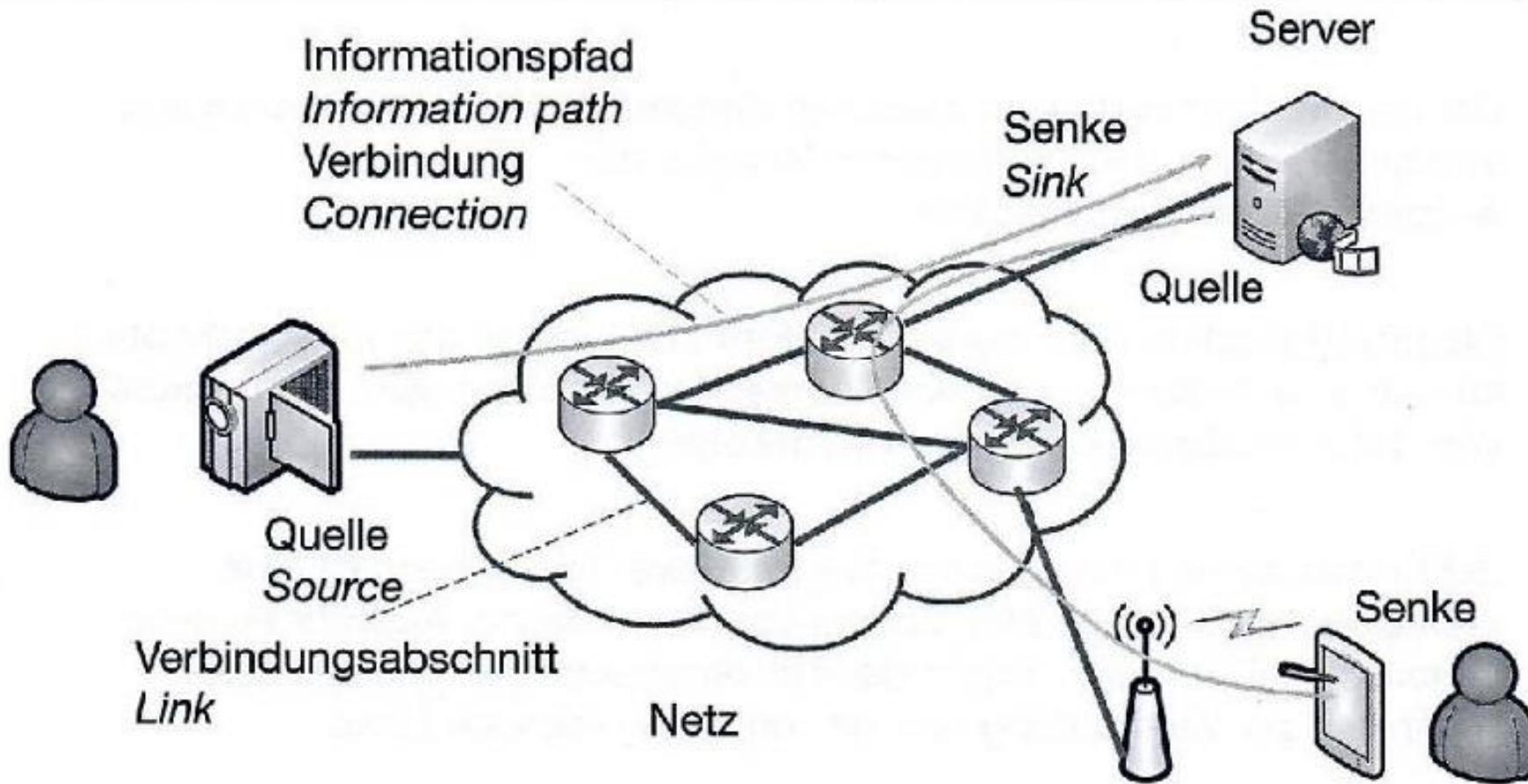
3) Aufgaben von Kommunikationsnetzen

- Der Informationsaustausch zwischen Endgeräten und/oder Netzknoten erfolgt anhand von wohldefinierten **Regeln**, den **Kommunikationsprotokollen**
- Die physikalischen oder logischen Informationspfade und die Netzknoten können in unterschiedlicher Konfiguration ausgebildet sein. Man spricht von **Netzstrukturen** oder von **Netzgraphen**
- Bei Kommunikationsnetzen mit mehr als zwei Teilnehmern ist eine **Zielauswahl** zu treffen. Man spricht von **Vermittlung**. Ausnahme: reine Verteilnetze (Rundfunk, TV). In den Netzknoten gibt es verschiedene Techniken zur Verknüpfung von ein- und ausgehenden Links.
- Das Finden und Einstellen von Wegen in einem Netz nennt man **Routing**

Quelle: [LKN]



5) Der Informationspfad im Netz



Informationen werden längs eines festen oder variablen Weges (**Informationspfad**) von einer **Quelle** zu einer oder mehreren **Senken** geführt. Dieser Pfad kann aus mehreren **Abschnitten (Links)** bestehen, in denen die Informationen mit verschiedenen Verfahren und Darstellungsformen transportiert werden.

Quelle: [LKN]



10) Beispiele von Kommunikationsnetzen

Mobile(Tele-)Kommunikation (GSM, GPRS, EDGE, UMTS, HSDPA/HSUPA, LTE , 5G)

- **2G: GSM**

Standard der sogenannten zweiten Generation („2G“) als Nachfolger der analogen Systeme der ersten Generation (in Deutschland: A-Netz, B-Netz und C-Netz) und ist der weltweit am meisten verbreitete Mobilfunk-Standard.

GSM wurde mit dem Ziel geschaffen, ein mobiles Telefonsystem anzubieten, das Teilnehmern eine europaweite Mobilität erlaubte und mit ISDN oder herkömmlichen analogen Telefonnetzen kompatible Sprachdienste anbot.

In Deutschland ist GSM die technische Grundlage der D- und E-Netze.

Datenübertragung: Datenrate von 9,6 kbit/s.

https://de.wikipedia.org/wiki/Global_System_for_Mobile_Communications

- **GPRS:** General Packet Radio Service

Bezeichnung für den paketorientierten Dienst zur Datenübertragung in GSM-Netzen

https://de.wikipedia.org/wiki/General_Packet_Radio_Service

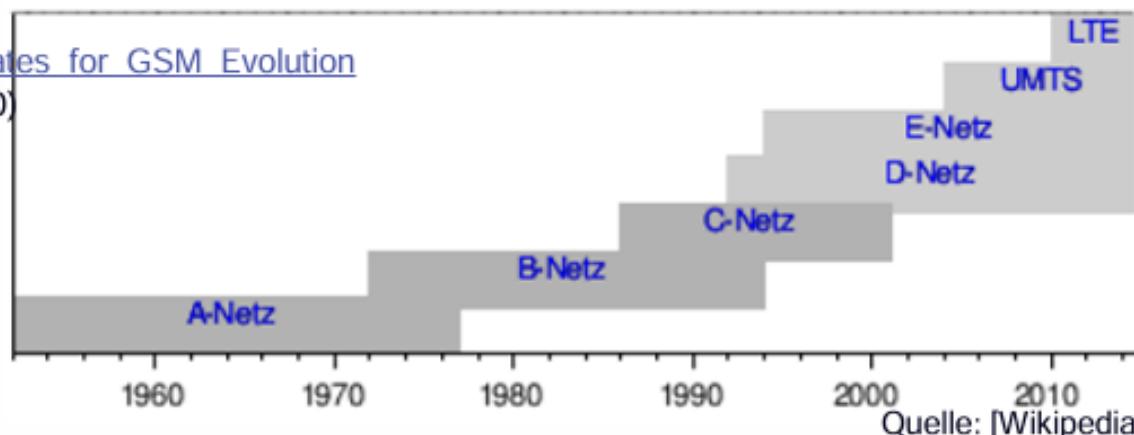
Max. Mögliche Übertragungsrate: 53 kBit/s

- **EDGE:** Enhanced Data Rates for GSM Evolution

Technik zur Erhöhung der Datenübertragungsrate in GSM-Mobilfunknetzen durch Einführung eines zusätzlichen Modulationsverfahrens. Erweiterung von GPRS

https://de.wikipedia.org/wiki/Enhanced_Data_Rates_for_GSM_Evolution

Max. Mögliche Downloadrate: 220 kBit/s (up 110)



10) Beispiele von Kommunikationsnetzen

Mobile(Tele-)Kommunikation (GSM, GPRS, EDGE, UMTS, HSDPA/HSUPA, LTE, 5G)

- **3G: UMTS: Universal Mobile Telecommunications System**

Mobilfunkstandard der dritten Generation (3G), mit dem deutlich höhere Datenübertragungsraten als mit dem Mobilfunkstandard der zweiten Generation (2G), dem GSM-Standard möglich sind: 384 kBit/s

https://de.wikipedia.org/wiki/Universal_Mobile_Telecommunications_System

- **HSPA: High Speed Packet Access**

Erweiterung des Mobilfunkstandards UMTS, die höhere Datenübertragungsraten ermöglicht. Sie gliedert sich in HSDPA zur Erhöhung der Datenübertragungsrate des Downlinks und in HSUPA für den Uplink.

https://de.wikipedia.org/wiki/High_Speed_Packet_Access

- **HSDPA: High Speed Downlink Packet Access**

Datenübertragungsverfahren des Mobilfunkstandards UMTS. Das Verfahren ermöglicht DSL-ähnliche Datenübertragungsraten im Mobilfunknetz.

https://de.wikipedia.org/wiki/High_Speed_Downlink_Packet_Access

- **HSUPA: High Speed Uplink Packet Access**

Datenübertragungsverfahren des Mobilfunkstandards UMTS, das höhere Datenübertragungsraten im Uplink ermöglicht und die Roundtrip-Zeiten (oft als Ping bezeichnet) verkürzt.

https://de.wikipedia.org/wiki/High_Speed_Uplink_Packet_Access

- **HSPA+**

Erweiterung von HSPA. (auch HSPA Evolution oder HSPA evolved).

Eingeführt wird HSPA+ mit Geschwindigkeiten ab 14,4 MBit/s im Downlink und 5,76 MBit/s im Uplink.

Ausblick: Längerfristig sollen sogar 168 MBit/s im Downlink und 23 MBit/s im Uplink möglich sein.

Derzeitiger Stand: 42 Mbit/s sind verfügbar



Quelle: [Wikipedia]



10) Beispiele von Kommunikationsnetzen

Mobile(Tele-)Kommunikation (GSM, GPRS, EDGE, UMTS, HSDPA/HSUPA, LTE, 5G)

- **4G: LTE: Long Term Evolution**



Mobilfunkstandard der vierten Generation. Mit bis zu 300 Mbit/s sind je nach Empfangssituation deutlich höhere Downloadraten als bei älteren Standards möglich.

https://de.wikipedia.org/wiki/Long_Term_Evolution

- **LTE-Advanced (Long-Term-Evolution-Advanced)**

Erweiterung des Mobilfunkstandards LTE, die höhere Datenübertragungsraten ermöglicht.

Zu den Verbesserungen gehören z.B. höhere Bandbreiten mit bis zu 1000 Mbit/s

- Betreiber Beispiele:

- Telekom:

<https://telekom.tarife-angebote.de/verfuegbarkeit/netzabdeckung-mobilfunk>

- Vodafone:

<http://www.vodafone-lte.de/verfuegbarkeit/lte-check>

<http://www.vodafone-lte.de/verfuegbarkeit/hspa-check>



Quelle: [Wikipedia]



10) Beispiele von Kommunikationsnetzen

Mobile(Tele-)Kommunikation (GSM, GPRS, EDGE, UMTS, HSDPA/HSUPA, LTE, 5G)

- 5G:
 - bis zu 10 Gbit/s
 - Echtzeitübertragung
 - Latenz: eine Millisekunde

<https://de.wikipedia.org/wiki/Echtzeit>



10) Beispiele von Kommunikationsnetzen

Satellitenkommunikation

<https://de.wikipedia.org/wiki/Satellitenkommunikation>

<https://de.wikipedia.org/wiki/Satellitentelefon>

Bei mobiler Satellitenkommunikation wird über ein Satellitentelefon eine Verbindung zu einem meist geostationären Nachrichtensatellit aufgebaut.

Der Vorteil der Satellitenkommunikation gegenüber terrestrischen Netzen ist, unter einer einzigen Satelliten-Ausleuchtzone gleichzeitig die Verbindung von zum Beispiel den Kanarischen Inseln bis zur chinesischen Grenze nutzen zu können und damit geographisch weit verteilte Netzketten und Nutzer von Sprache, Daten und Video erreichen zu können.

Andererseits sind die Verbindungspreise höher als bei terrestrischen Mobilfunksystemen oder dem Festnetz. Auch müssen die Antennen zum Satelliten ausgerichtet werden, was überall dort möglich ist, wo auch eine theoretische Sichtverbindung zum Satelliten besteht.

Ein entsprechendes mobiles Satellitentelefon ist in einem Gehäuse in der Größe zwischen einem etwas größeren normalen Mobiltelefon oder eines Laptops untergebracht. Es ermöglicht nicht nur Telefonieren, sondern alle anderen Arten der Datenübertragung wie Fax, E-Mail oder Internet.

Stationäre Satellitenanlagen sind heute nahezu mobil und werden **VSAT** (Very Small Aperture Terminals) genannt. Der Vorteil der stationären beispielsweise 75 cm VSATs liegt bei der sehr hohen Übertragungsbandbreite von mehreren Mbit/s, was sehr schnellen Internet-Zugriff von nahezu jedem Punkt der Erde ermöglicht und das zu moderaten Preisen.



Satellitentelefone (IsatPhone Pro/Inmarsat, Iridium 9555, Thuraya XT)



Satellitenkommunikationsgerät für Daten- und Sprachkommunikation

Quelle: [Wikipedia]



11) Typisierung von Kommunikationsnetzen

- ***Wide Area Networks (WAN)***
 - Weitverkehrsnetz, z.B. nationales Netz, weltweites Netz, Kernnetz bei der Mobilkommunikation, Internet Backbone Netze
 - Übergänge zu anderen WAN und zu LAN, MAN
- ***Metropolitan Area Networks (MAN)***
 - Regionale Netze, z.B. in einer Stadt
 - Ausdehnung bis zu 100 km
- ***Local Area Networks (LAN)***
 - Lokale Netze, z.B. Büro, Heimbereich, Cafe
 - Drahtlose Netze (Wireless LAN)
 - Verbindung von Servern in einem Rechenzentrum
 - Fahrzeugnetze
 - Ausdehnung bis zu 500 m
- ***Personal Area Networks (PAN)***
 - Vernetzung von persönlichen Geräten
 - Techniken: USB, IrDA, Bluetooth
 - Ausdehnung: wenige Meter

Quelle: [LKN]



Signale

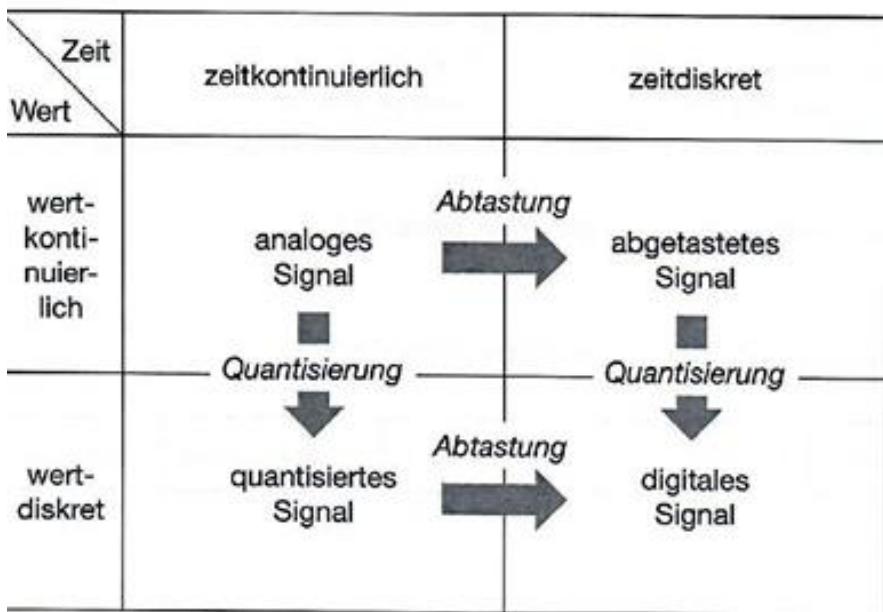
- Zeichenübertragung und damit **Informationsübertragung** geschieht unter Verwendung von **Signalen**.
- Ein Signal ist die **Darstellung einer Nachricht durch physikalische Größen**,
- z.B. Spannungssignale, Stromsignale, Lichtsignale, akustische Signale,...
- Vier grundsätzliche Signalklassifizierungen
 - Analoges Signal: wert- und zeitkontinuierlich, d.h. jeder Wert kann zu jeder Zeit vorkommen
 - Abgetastetes Signal: die Werte des Signals liegen nur zu bestimmten Zeiten (zeitdiskret) vor
 - Quantisiertes Signal: das Signal kann nur bestimmte Werte einnehmen
 - Digitales Signal: wert- und zeitdiskretes Signal

Quelle: [LKN]

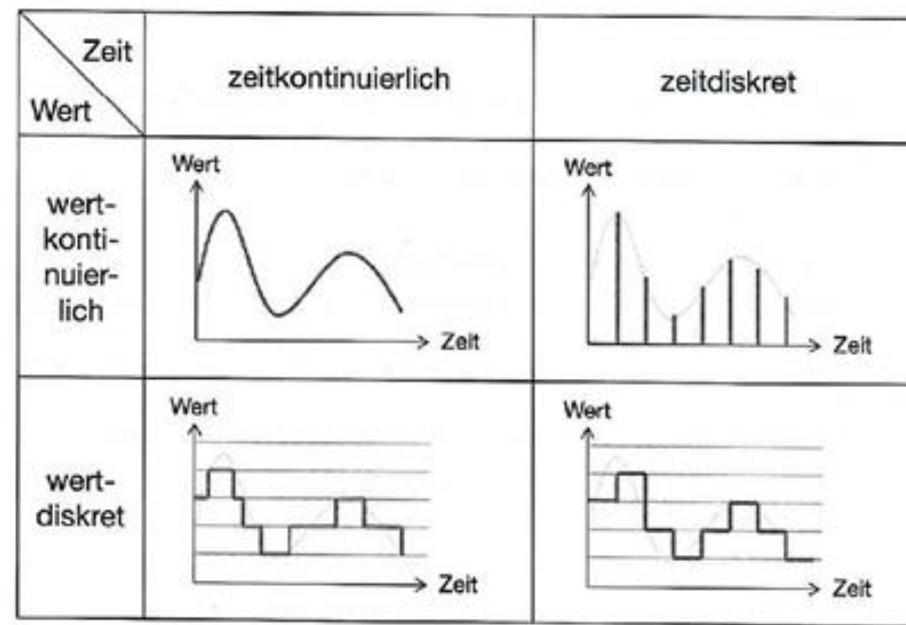


Klassifizierung von Zeitsignalen (Wert/ Zeit kont./diskret)

Klassifizierung



Beispiele

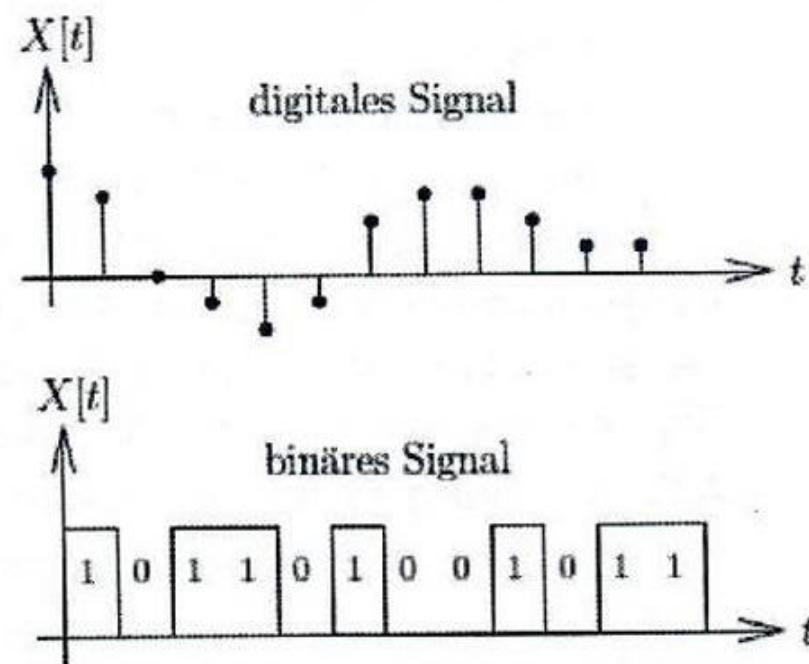


Quelle: [LKN]



Binäres Signal

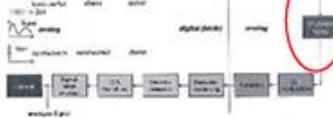
- **Analoges Signal:**
wert- und zeitkontinuierlich
- **Digitales Signal:**
wert- und zeitdiskret
- **Binärsignal:** Signal
mit 2 Quantisierungsstufen
 - Spezialfall des digitalen Signals
 - nur zwei Werte nämlich „0“ und „1“
können eingenommen werden



Quelle: [LKN]



Informationsübertragungstrecke: Überblick



Übertragungsmedien im Kanal (Mobilfunk, Richtfunk, Sat)

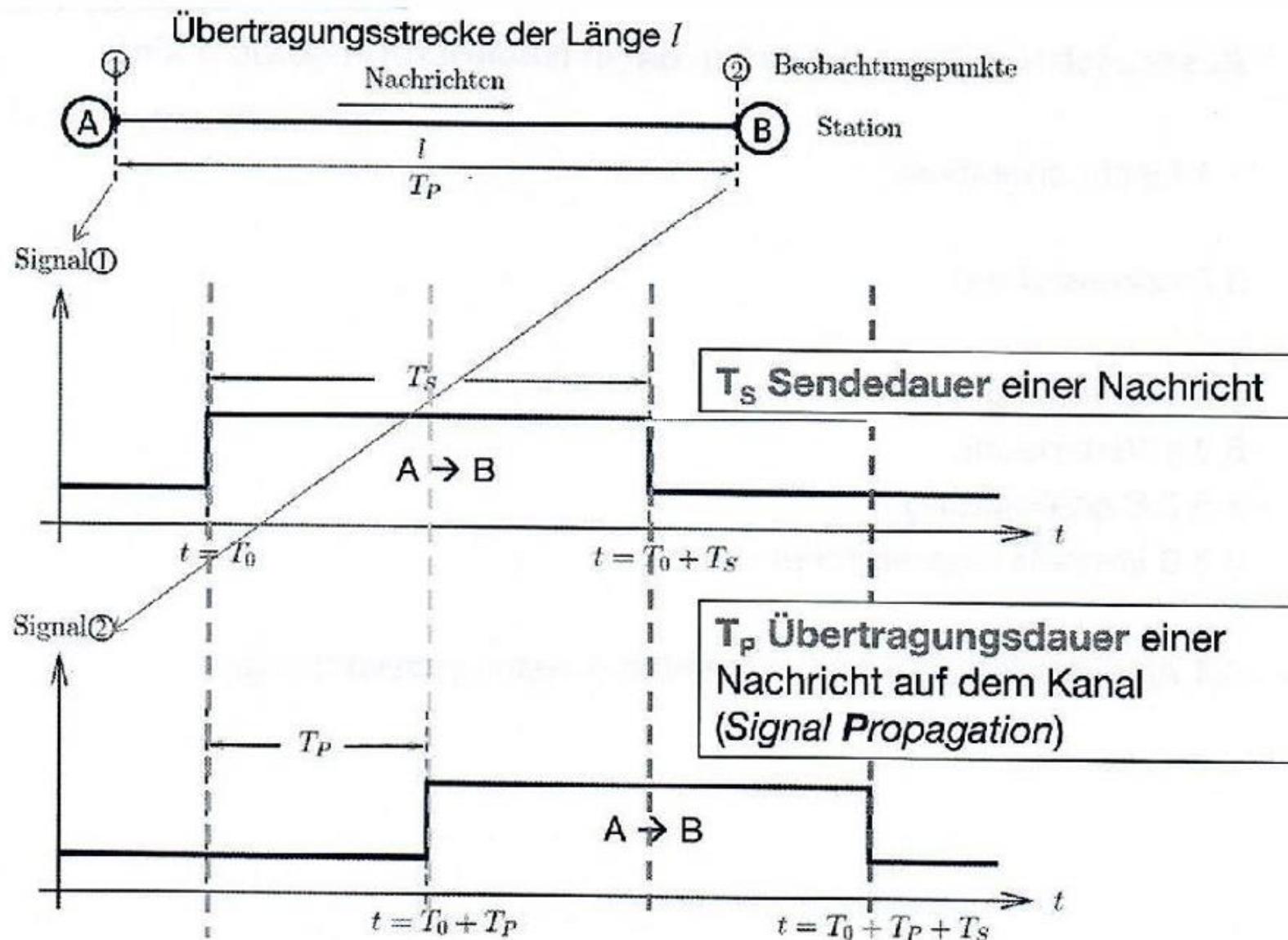
Funkübertragung

- Elektromagnetische Wellen
- Mobilfunk
 - Frequenzbereich von einige 100 MHz bis einige GHz
 - Überbrückbare Streckenlänge von mehreren km (Zellgröße)
- Richtfunk
 - Frequenzbereich von einigen GHz
 - Überbrückbare Streckenlänge ca. 50 km
- Satellitenfunk
 - Frequenzbereich von einigen GHz
 - Überbrückbare Streckenlänge tausende von km
 - Problem: Große Signallaufzeiten (250 – 300 ms je Richtung)

Quelle: [LKN]



Nachrichtenfluss

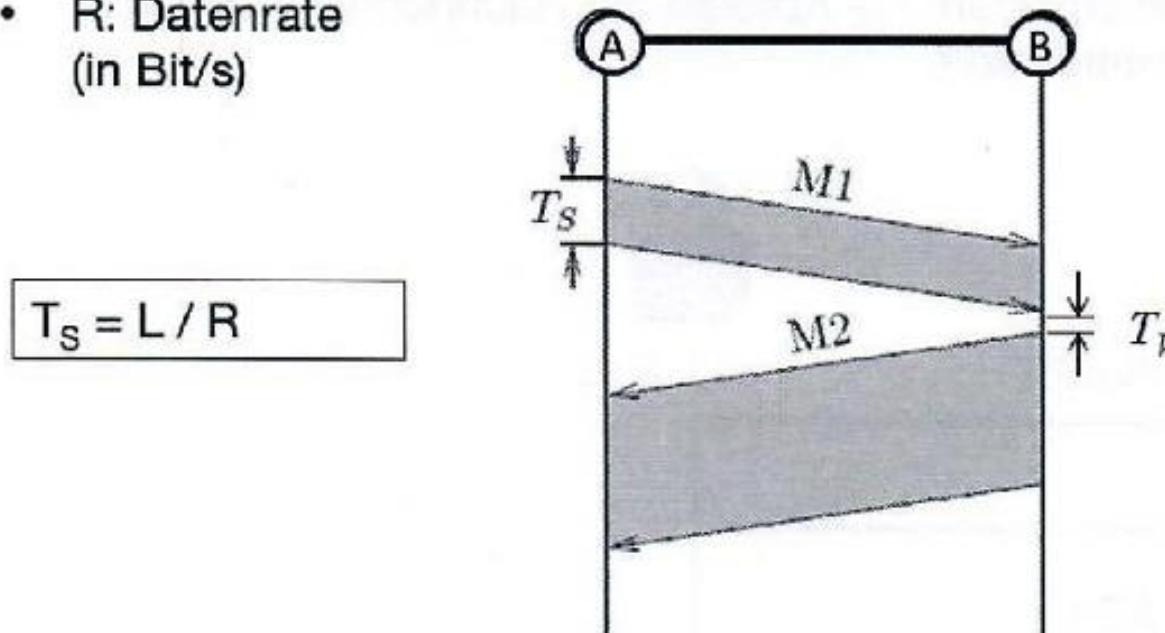


Quelle: [LKN]



Nachrichtenflussdiagramm

- Erweiterung um Nachrichtendauer T_s
- L: Länge einer Nachricht (typischerweise in Bits oder Bytes)
- R: Datenrate
(in Bit/s)



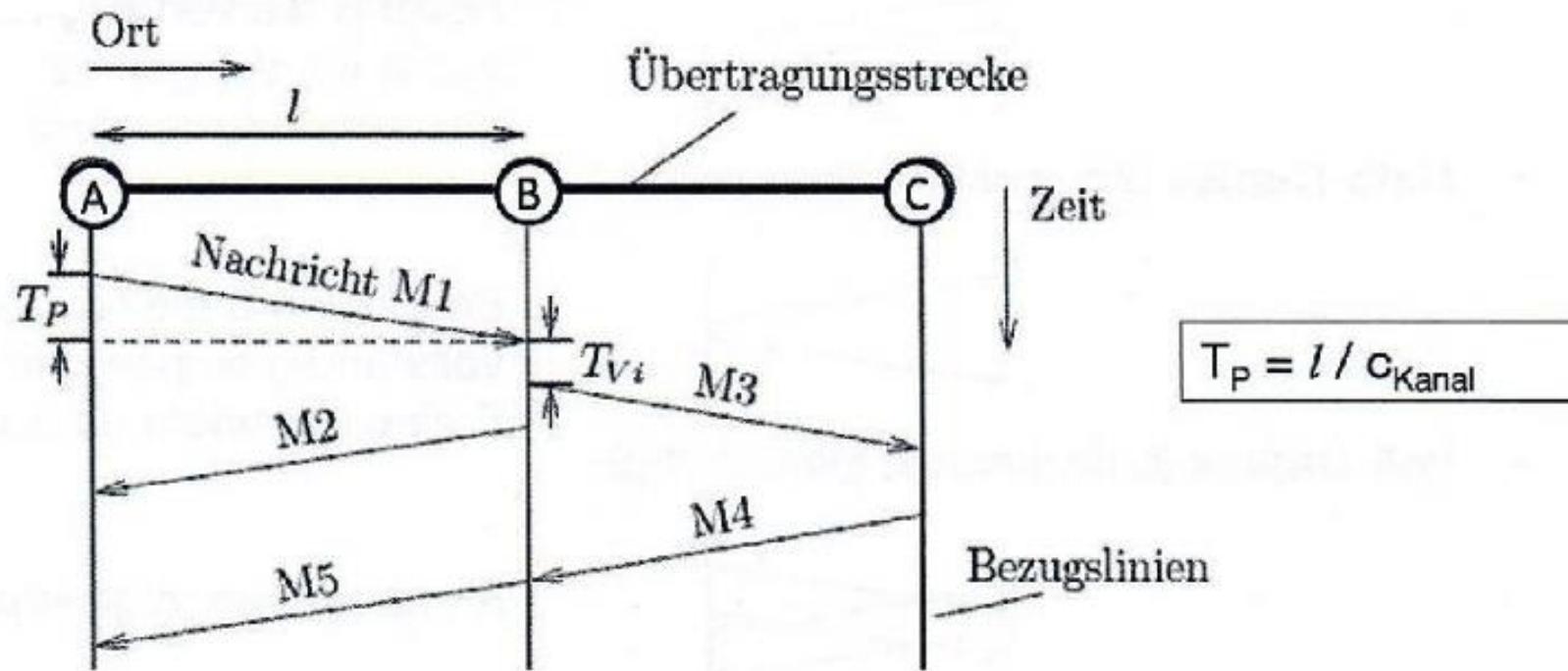
Wichtige Anmerkungen:

- Das Nachrichtenflussdiagramm zeigt nur einen ausgewählten Fall und keine allgemeine Systembeschreibung!
- Die Logik der Knoten d.h. die Regeln für den Nachrichtenaustausch (Protokolle) wird durch Zustandsautomaten beschrieben

Quelle: [LKN]



Nachrichtenflussdiagramm (Message Flow Diagram)



- Bei gleicher Signal-Ausbreitungsgeschwindigkeit c_{Kanal} (z.B. c_{Kupfer} , c_{Fiber}) auf allen Teilstrecken ist Steigung l/T_P überall gleich

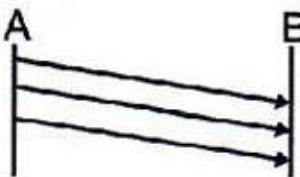
T_{Vi} : Verzögerungszeit im Netzknoten
(besteht aus Verarbeitungszeit und Speicherung im Puffer)

Quelle: [LKN]



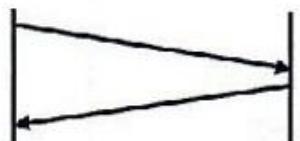
Gleichzeitigkeit der Kommunikation

- Simplex (unidirektional)



Nachricht kann abgesendet werden,
bevor B die vorherige Nachricht
von A empfangen hat
(überlappend senden)

- Halb-Duplex (abwechselnd)



Erst, wenn B die Nachricht von A
vollständig empfangen hat, kann
B eine Nachricht an A senden.

- Voll-Duplex (bidirektional, gleichzeitig)



A und B können gleichzeitig senden.

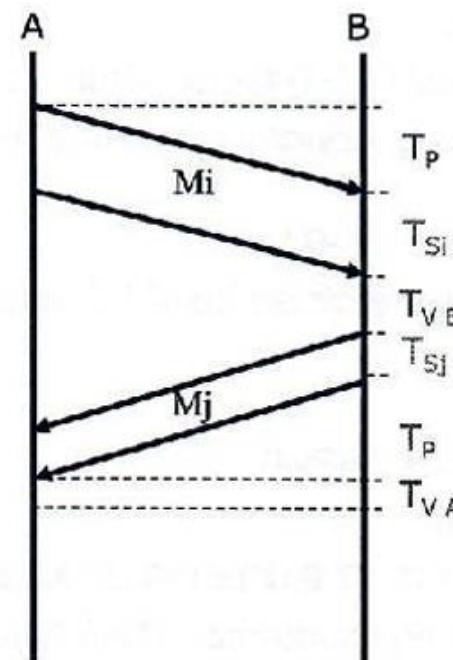
- Bei Voll-Duplex ist Richtungstrennung nötig: getrennte Leitungen oder Frequenzen, Wellenlängen, etc.

Quelle: [LKN]



NFD – Zusammenfassung

- Nachrichtensendedauer T_S
- Kanallaufzeit T_P
- Verzögerungszeit im Knoten T_V
- Zwei direkt benachbarte Netzknoten (*Single Hop*)



Quelle: [LKN]



Adressierung

- Adressierung beinhaltet alle Operationen zur eindeutigen Identifizierung einer Ressource in einem Kommunikationsnetz
- Ressource = Betriebsmittel
 - Teilnehmerendgerät
 - Server
 - Netzknoten
 - Informationspfad
 - Videofile
 - Webseite
 - Videoplayer (Anwendung)
- Beispiele für Adressen
 - Telefonnummer: +49 89 289 23500
 - Skype ID: hansmustermann
 - IP Telefonie: <sip: alice@example.com>
 - URL: <http://www.lkn.ei.tum.de/team/mitarbeiter/wolfgang-kellerer.html>
 - IP Adresse: 192.168.123.2 (IPv4)
 - MAC Adresse: 24-A3-BA-5F-77-02

→ Adressen existieren auf verschiedenen Ebenen in einem Kommunikationsnetz

Quelle: [LKN]



Adressauflösung

Abbildung zwischen unterschiedlichen Adressformaten.

- Globale Auflösung durch Index-Server (Datenbank)
 - Beispiele: DNS, Skype,...
- Lokale Auflösung durch Broadcast
 - Beispiele: ARP, Paging,...

Beispiele

- *Domain Name Service* (DNS): verteilte, hierarchisch strukturierte Datenbank
 - Auflösung des *Domain* Namens in einer URL in eine IP-Adresse
 - www.ei.tum.de → 129.187.254.81
- *Address Resolution Protocol* (ARP) im Internet (LAN)
 - Dynamische Abbildung von logischen Netzwerkadressen auf die korrespondierenden Hardware Adressen
 - 192.168.123.2 → 24-A3-BA-5F-77-02
 - Erfolgt mittels ARP Request als Broadcast in einem lokalen Netzsegment
 - Die ARP Instanz erkennt ihre IP Adresse und antwortet

Quelle: [LKN]



Adresszuweisung

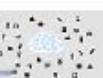
Vor der Benutzung einer Adresse muss diese dem Gerät zugewiesen werden

- Manuell
- Automatisch
- am Endgerät dezentral konfiguriert
- zentral administriert

Beispiel

- *Dynamic Host Configuration Protocol (DHCP)*

Quelle: [LKN]



Adresszuweisung

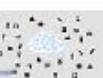
Vor der Benutzung einer Adresse muss diese dem Gerät zugewiesen werden

- Manuell
- Automatisch
- am Endgerät dezentral konfiguriert
- zentral administriert

Beispiel

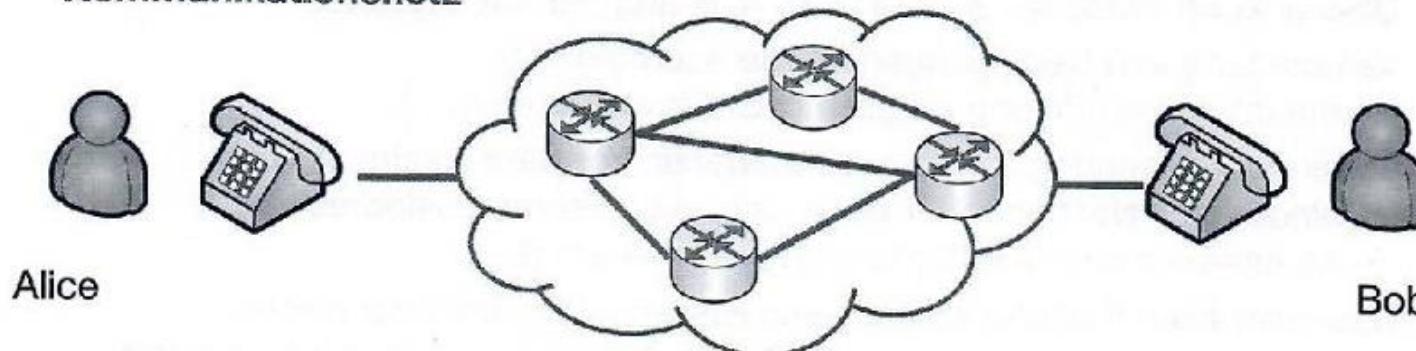
- *Dynamic Host Configuration Protocol (DHCP)*

Quelle: [LKN]



Vermittlung

Aufgabe: Nachrichtenaustausch zwischen Partnern Alice und Bob über ein Kommunikationsnetz



Schritte:

1. Herstellen einer Beziehung zwischen Alice und Bob
(Adressen müssen bekannt sein)
2. Aufbau einer Verbindung
3. Nachrichtenübertragung
4. Abbau der Verbindung

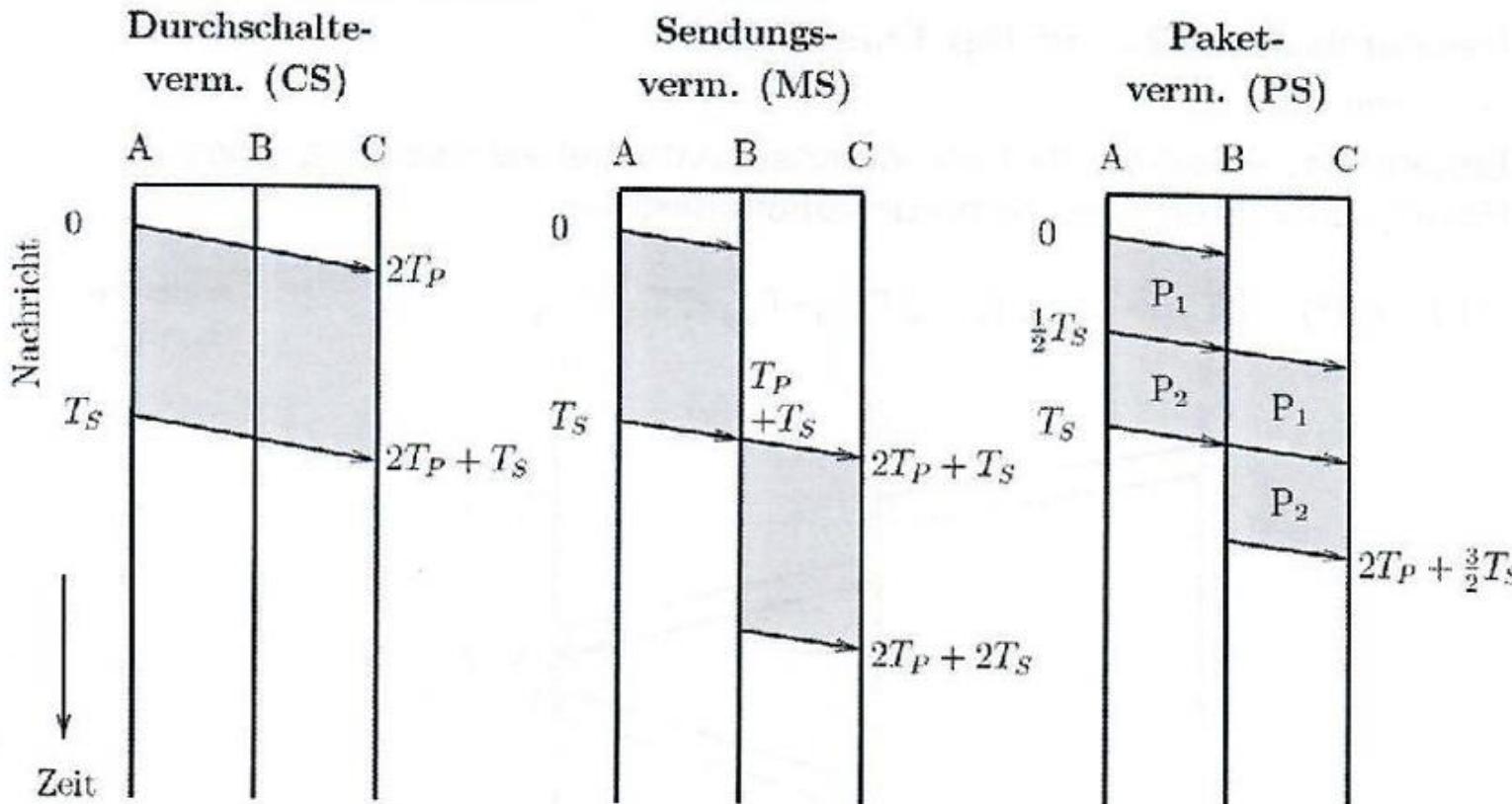
} verbindungsorientierter Betrieb
(connection oriented)
(CO)
Beispiel: Telefonie

verbindungsloser Betrieb
(connection less) (CL)
(nur 1. und 3.) – Beispiel: Briefpost

Quelle: [LKN]



Beispiel: Vergleich der Vermittlungsverfahren



Übermittlungszeit für die gesamte Nachricht:

$$T_{CS} = 2T_P + T_S$$

$$T_{MS} = 2T_P + 2T_S$$

$$T_{PS} = 2T_P + 1,5 T_S$$

$$T_{CS} < T_{PS} < T_{MS}$$

Quelle: [LKN]



Kommunikationsprotokolle: Definition & Aufgaben

Definition

- Ein Kommunikationsprotokoll ist ein im gesamten Netz gültiger Satz von Regeln zum Austausch von Nachrichten zwischen zwei oder mehr Kommunikationspartnern
- Zur Sicherung der einwandfreien Zusammenarbeit zwischen Kommunikationspartnern müssen diese dasselbe Protokoll verwenden
- Zwischen unterschiedlichen Protokollen sind fallweise Protokollwandlungen erforderlich

Aufgaben der Protokolle

- Übertragung von Nutz- und Steuernachrichten (Signalisierung, *signaling*)
- Fehlererkennung (*error detection*)
- Fehlerbehandlung (*error control*)
- Flusssteuerung (*flow control*)
- Staubehandlung (*congestion control*)

Kommunikationsprotokolle sind die Grundlage aller Kommunikationsnetze

Quelle: [LKN]



Prozessbegriff: Kommunikationsprotokolle sind kommunizierende Prozesse

- Prozess ist durch eine Folge von Zuständen definiert
 - Ähnlich: Betriebssystemprozesse in Computern
- Prozesse befinden sich in Zuständen
- Aufgrund von Ereignissen können Zustandsübergänge stattfinden
 - Zustands-Ereignis-Verknüpfung
- In einem Zustandsübergang können zusätzlich Aktionen ausgeführt werden, um auf das Ereignis zu reagieren

Beispiel für eine Zustands-Ereignis-Verknüpfung



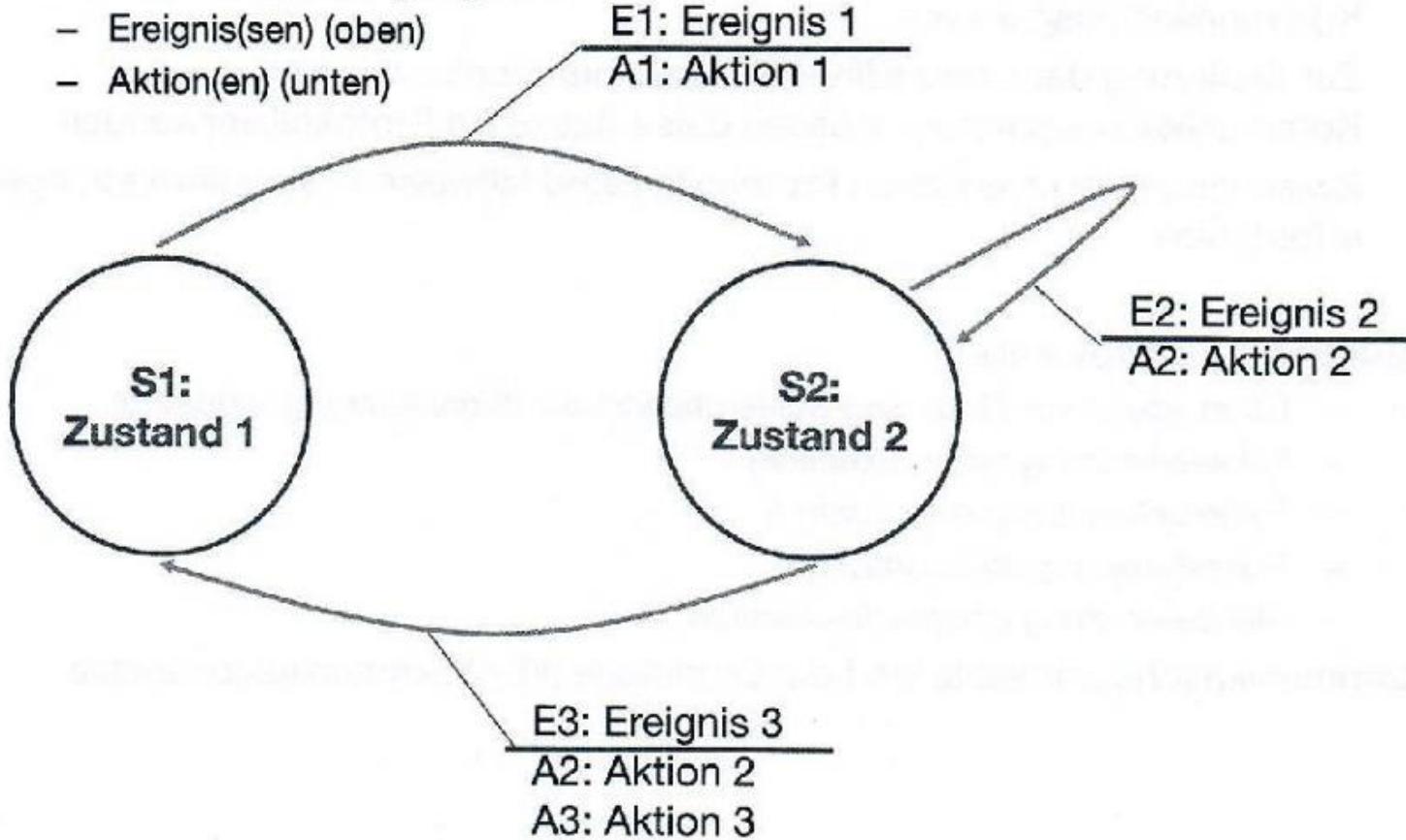
Quelle: [LKN]



Zustandsdiagramm

Notation

- Kreis: Zustand mit Nummer und Bezeichnung
- Pfeil: Zustandsübergang mit
 - Ereignis(sen) (oben)
 - Aktion(en) (unten)



Quelle: [LKN]



Protokoll-Wirkungsgrad

Ausnutzung (*Utilization*) des Übertragungskanals als Verhältnis der Nachrichtensendedauer zur der Zeit, zwischen Aussenden des ersten Bit der Nachricht bis zur Aussendung des ersten Bits einer neuen Nachricht, d.h. bis eine neue Nachricht übertragen werden kann

$$\rho = \frac{\text{(Nutz-)Nachrichtensendedauer}}{\text{Zeit bis eine neue Nachricht gesendet werden kann}}$$

mit Nachrichtensendedauer $T_S = L_M / R$

mit R : Bitrate auf den Übertragungskanal

mit Kanallaufzeit $T_P = l / c$

l : Länge der Leitung

c : Ausbreitungsgeschwindigkeit



Grundlegende Konzepte für Kommunikationsprotokolle

Kommunikationsprotokolle stellen sicher, dass

- Nachrichten übertragen werden
- Fehlübertragungen korrigiert werden (z.B. Wiederholung)
- der Kanal entsprechend der Systemumgebung ausgelastet wird (Flusssteuerung und Stauregelung)
- Die Kommunikationspartner fair auf den Kanal zugreifen können (Multiplex)

Dabei werden zusätzliche Steuernachrichten eingesetzt (z.B. ACK) und Timer-Abläufe überwacht

Basisprotokolle:

- Quittungsprotokolle zur Fehlererkennung und Fehlerkorrektur
- Protokolle mit Zeitüberwachung
- Protokolle mit Folgenummern
- Fensterprotokolle
- Protokolle mit selektiver Wiederholung

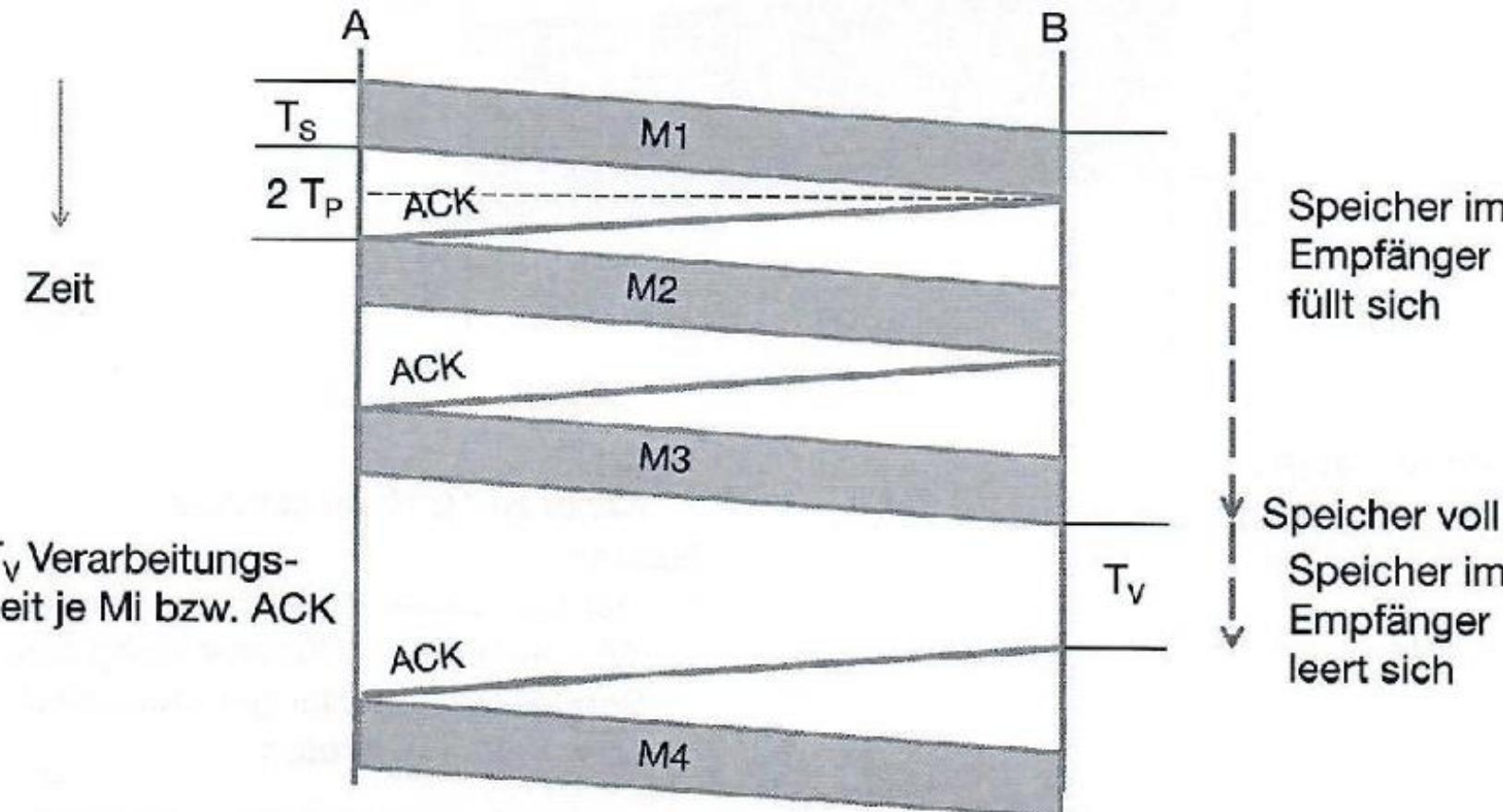
Quelle: [LKN]



B) Einfaches Quittungsprotokoll: Stop & Wait: NFD

Prinzip: Empfänger quittiert Empfang jeder Nachricht M_i mit Quittung ACK
(Länge L_A , Sendedauer T_A) „Stop and Wait“ für Modell II

Nachrichtenflussdiagramm (hier: Annahme $T_S \gg T_A$)



Quelle: [LKN]



B) Einfaches Quittungsprotokoll: Stop & Wait: Wirkungsgrad

- Annahmen: $T_V = 0$ Verarbeitungszeit
- $T_A = 0$ und $L_A = 0$ Quittung
- $L_H = 0$ Header

$$\rho = \frac{T_s}{T_s + 2T_p} = \frac{1}{1 + \frac{2T_p}{T_s}}$$

$$T_s = L / R$$

⇒ Wirkungsgrad:

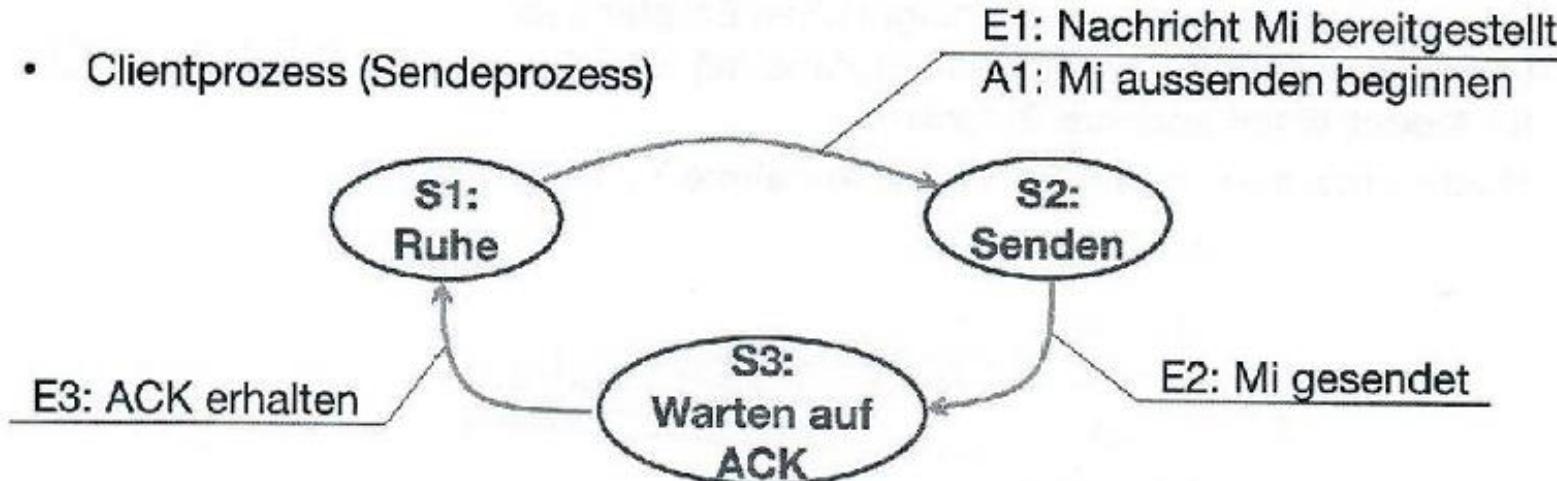
- sinkt mit steigendem R
- sinkt mit sinkendem L
- Sinkt mit steigendem T_p

Quelle: [LKN]

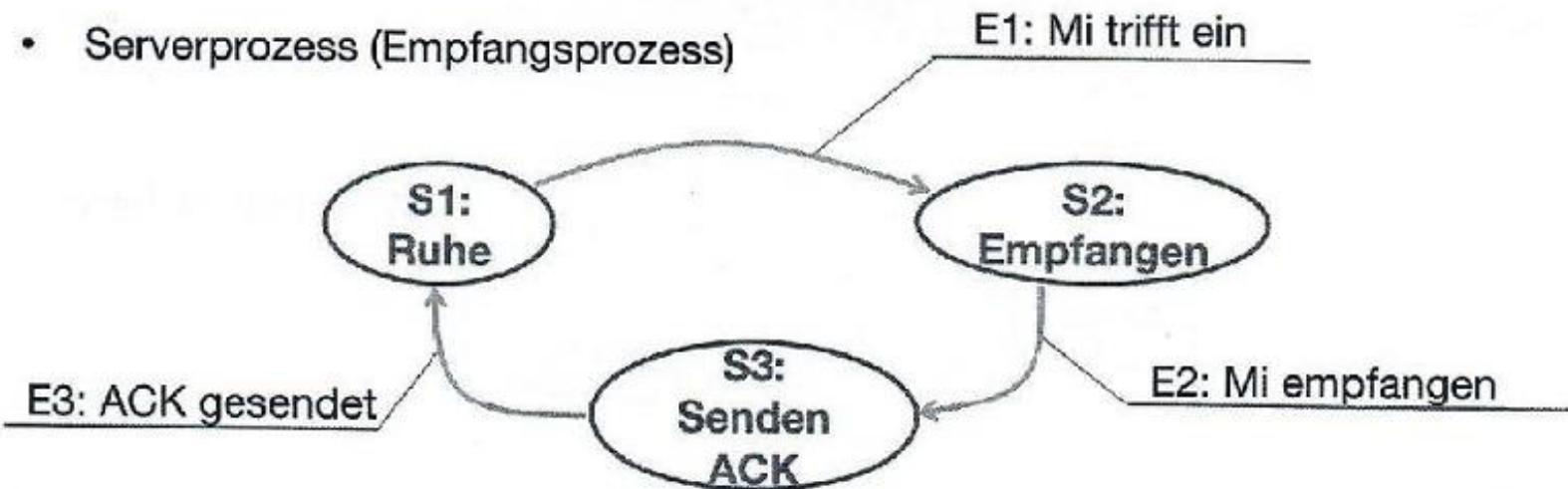


B) Einfaches Quittungsprotokoll: Stop & Wait: Zustandsdiagramm

- Clientprozess (Sendeprozess)



- Serverprozess (Empfangsprozess)



Quelle: [LKN]



F) Go-Back-N ARQ Fensterprotokoll –Fenstermechanismus

Sender:

SN_{S1} : niedrigste SendeNummer der noch nicht quittierten Nachricht

SN_{S2} : SendeNummer der zuletzt gesendeten Nachricht M

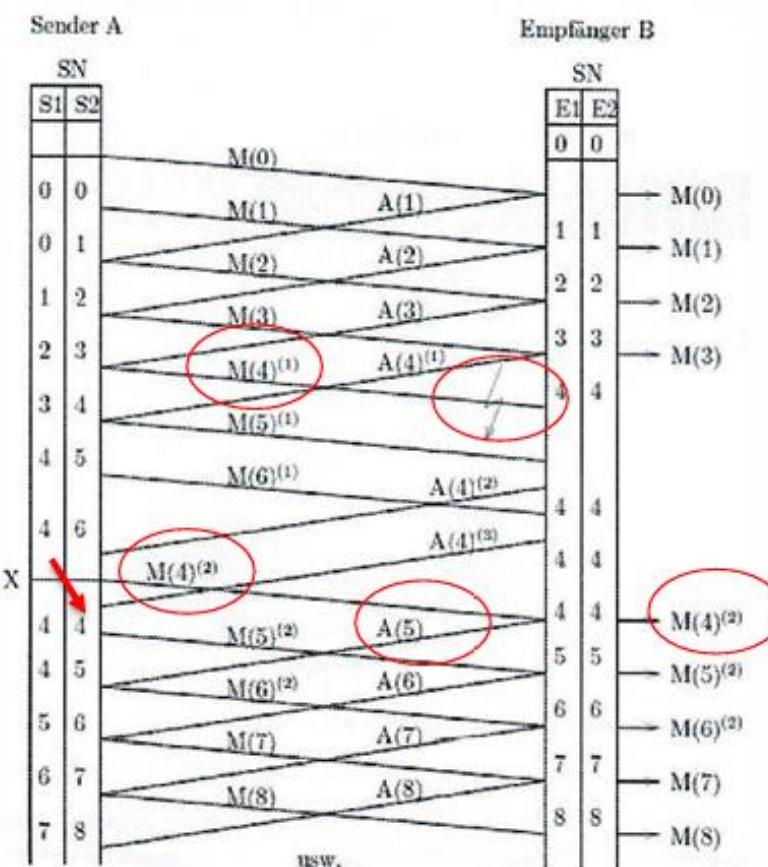
W_s : Sende-Fenster:

Anzahl ausstehender ACKs
(entspricht Anzahl zu haltender Nachrichten im Sender)

W_{Sm} : Sende-Fenster-Maximum:
Maximale Anzahl an erlaubten ausstehenden ACKs.
(Größe Sendepuffer)

Beispiel:

- $W_{Sm} = 3$
- X: Ereignis
Timer TACK für M(4) ist abgelaufen.
 $\Rightarrow M(4)$ wird wiederholt
 \Rightarrow Zurücksetzen von SN_{S2} auf SN_{S1}



Empfänger:

SN_{E1} : SendeNummer der nächsten zu empfangenden Nachricht.
Bei erfolgreichem Empfang von $M(SN_{E1}) \Rightarrow ACK(SN_{E1}+1)$

SN_{E2} : Höchste mögliche Nummer der nächsten zu empfangenden Nachricht.

Empfangs-Fenster:
Maximale Anzahl der Nachrichten die Empfänger speichern kann.

Beispiel:

- $WE = 1$

Quelle: [LKN]



G) Selective Repeat ARQ Fensterprotokoll – Fenstermechanismus

Sender:

SN_{S1} : niedrigste SendeNummer der noch nicht quittierten Nachricht

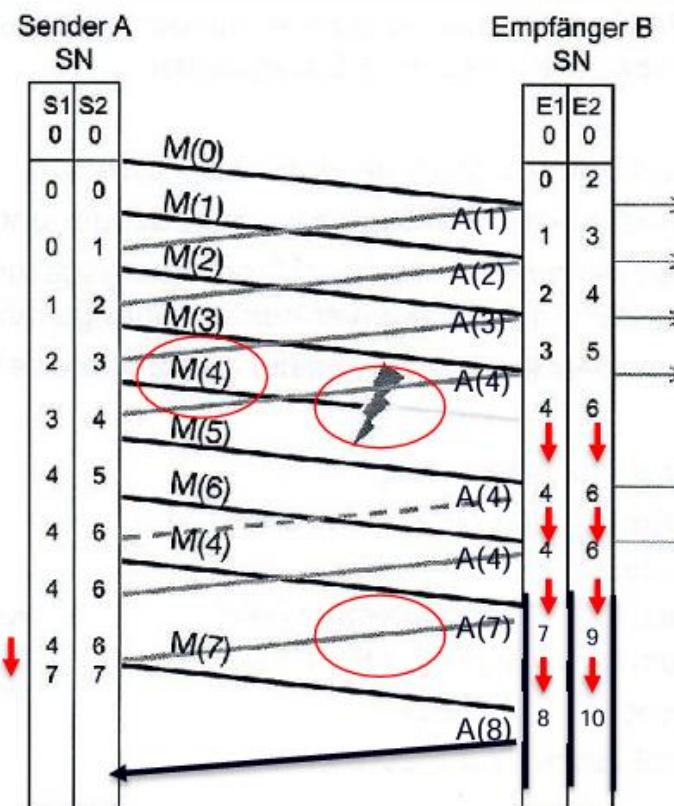
SN_{S2} : SendeNummer der zuletzt gesendeten Nachricht M

W_S : Fenster-Sender:
Anzahl ausstehender ACKs
(entspricht Anzahl zu haltender Nachrichten im Sender)

W_{Sm} : Fenster-Sender-Maximum:
Maximale Anzahl an erlaubten ausstehenden ACKs.
(Große Sendepuffer)

Beispiel:

- $W_{Sm} = 3$



Empfänger:

SN_{E1} : SendeNummer der nächsten zu empfangenden Nachricht.
Bei erfolgreichem Empfang von M(SN_{E1}) => ACK($SN_{E1}+1$)

SN_{E2} : Höchste mögliche Nummer der nächsten zu empfangenden Nachricht.

W_E : Empfänger-Fenster:
Maximale Anzahl der Nachrichten die Empfänger speichern kann.

Beispiel:

- $W_E = 3$

Quelle: [LKN]



Was ist das Internet? – eine technische Beschreibung

Mobilfunknetze

IX (Internet Exchange Point)

Internet Service Provider (ISP)

<https://de.wikipedia.org/wiki/Internetdienstanbieter>

Heimnetze

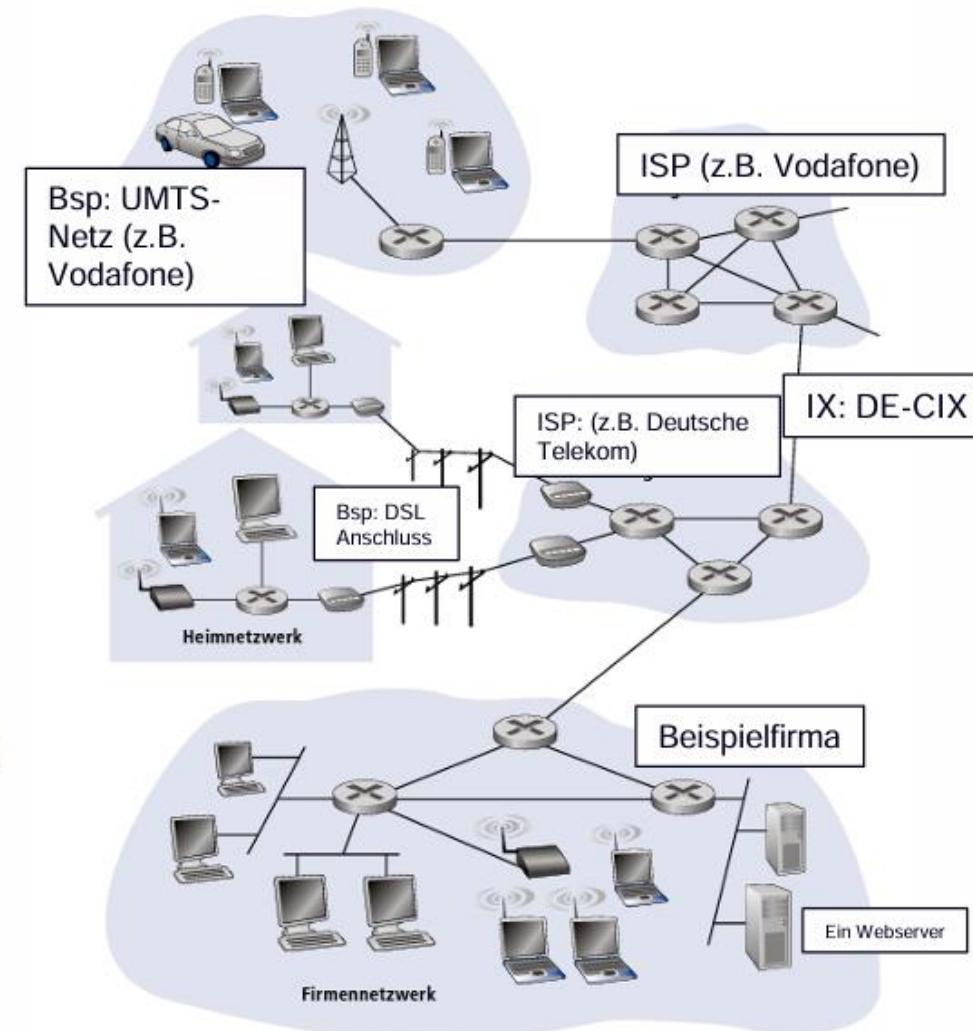
Firmennetze

Internet Exchange Point:

<https://www.de-cix.net/about/what-is-an-ix/>

<https://www.de-cix.net/customers-partners/customers/de-cix-frankfurt/>

=> Anschlüsse der ISP



Legende:



[KuRo]



Eine Dienstbeschreibung

Das Internet ist eine Infrastruktur die den Anwendungen Dienste zur Verfügung stellt.

Anwendungen im Internet:

- sind verteilt (mehrere Endsysteme)
- tauschen miteinander Daten aus
- laufen nur auf Endsystemen

Dienste des Internet:

- ⇒ das Netz erlaubt den Datenaustausch
- ⇒ die Elemente des Netzes (z.B Router) haben nichts mit den Anwendungen zu tun:
Sie erlauben lediglich den Datenaustausch.

Dienste/ Protokollfamilie:

<https://de.wikipedia.org/wiki/Internetprotokollfamilie>

- Datentransport Endsystem-zu-Endsystem: IP, IPsec, ICMP
- Innerhalb des Netzes: EIGRP, OSPF, BGP, RIP



Netzrand: Client / Server- Programme

Auf den Systemen des Netzrandes:

- **Client – Server Programme:**

- Client-Programm läuft auf einem Endgerät (Bsp Webbrowser)
- Das Server-Programm läuft auf einem anderen Endgerät (Bsp Webserver)
- Client fordert von einem Server-Programm Daten an.

- **Peer-to-Peer (P2P) Programme:**

Auf Endgeräten läuft ein P2P-Programm, dass sich gleichzeitig

- wie ein Client-Programm,
- und auch wie ein Server-Programm

verhält.

Damit kann das P2P-Programm sowohl Daten senden, als auch empfangen.
(Bsp: Filesharing Plattform Bittorrent, Skype,...)



Zugangsnetze (Access Networks)

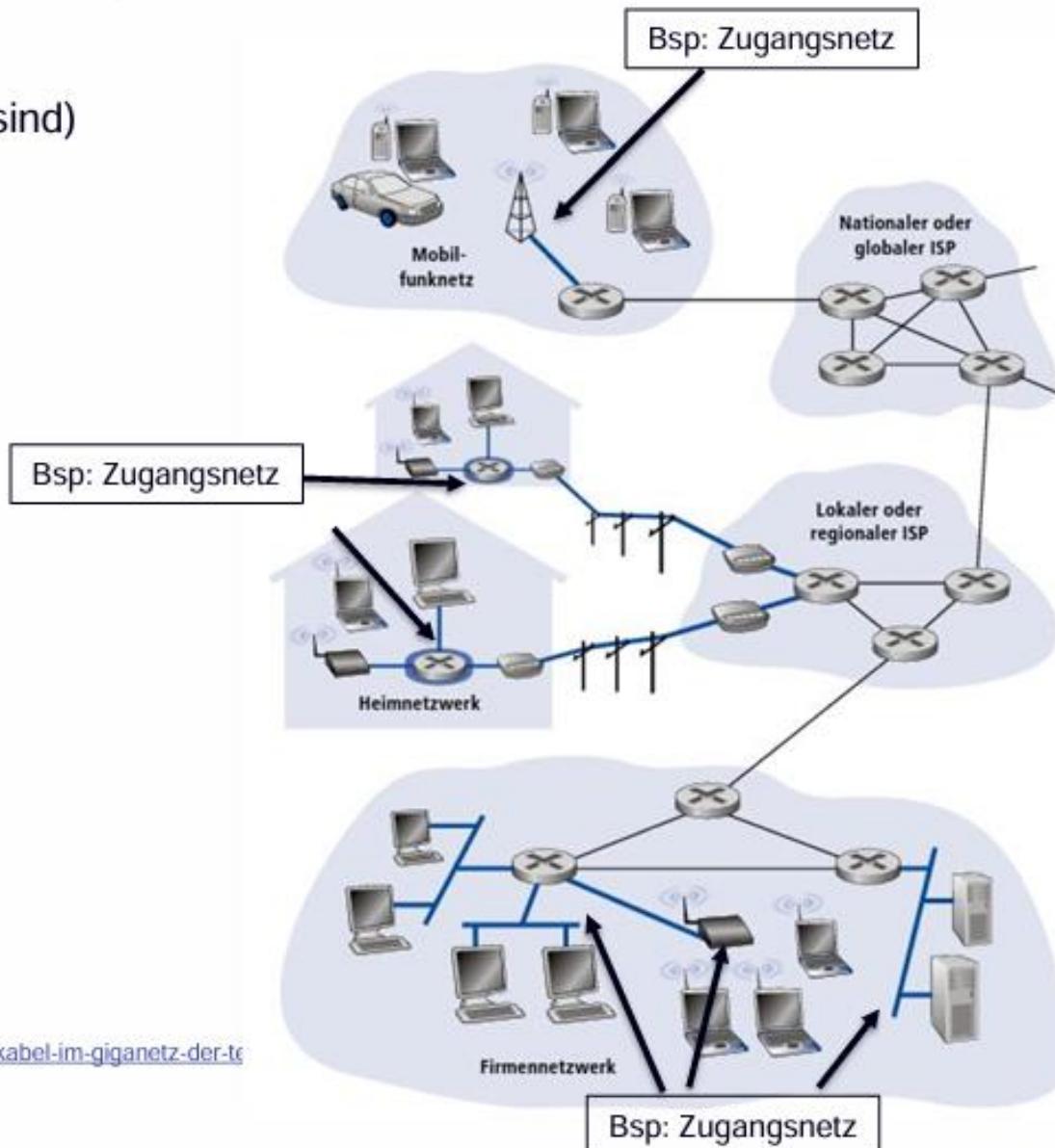
verbinden Endsysteme
(Geräte die an das Internet angeschlossen sind)
mit dem ersten Router
(Edge-Router).

Arten von Zugangsnetzen:

- Heimzugang
- Firmenzugang
- Drahtgebundener Zugang
- Drahtloser Zugang (Wireless)

Produkte:

- DSL
Bsp: <http://www.m-net.de/privatkunden/>
- Kabel
Bsp: www.unitymedia.de
- FTTH (Fiber to the Home)
Bsp: <http://tarife-und-produkte.t-online.de/ftth-internet-ueber-glasfaserkabel-im-giganetz-der-te>



[KuRo]



Zugangsnetze (Access Networks)

Drahtloser Zugang zum Internet

1. Wireless LAN (WLAN)

typischerweise:

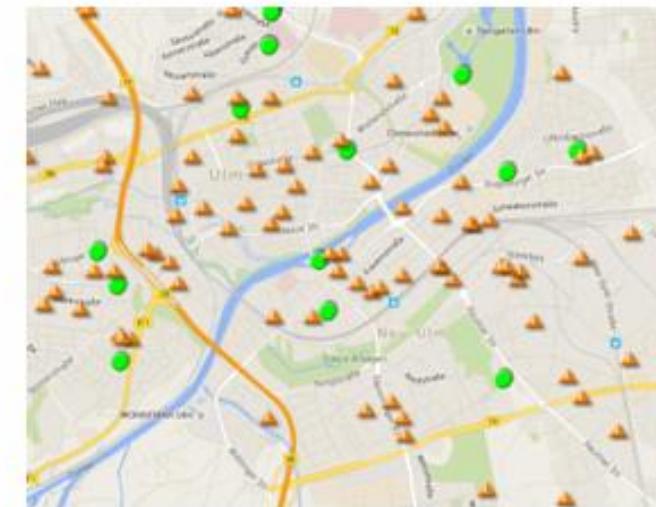
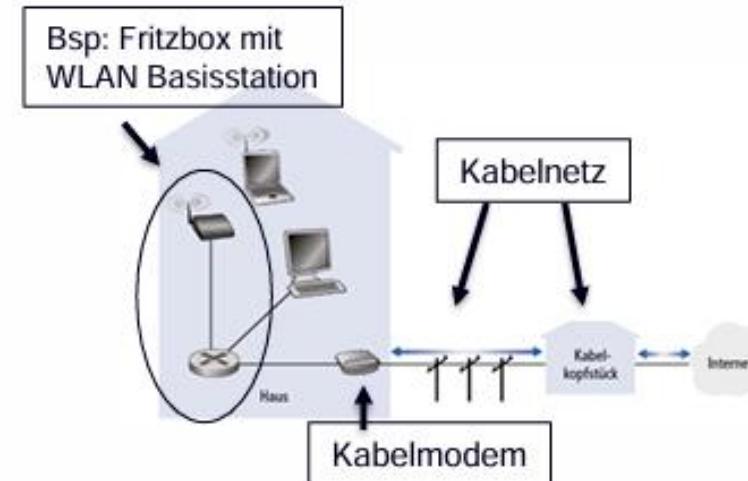
- eine Basisstation (Wireless Access Point) (z.B. Fritzbox)
- diese Basisstation wird vom Hauseigentümer betrieben
- Reichweite: wenige Meter
- WLAN-fähiges Endsystem (z.B. Laptop)

2. Wide Area Wireless Access Networks (WAWAN)

- Flächendeckende Infrastruktur
- Sehr viele Basisstationen
- diese Basisstationen werden von einem Telekommunikationsunternehmen betrieben (z.B. Dt. Telekom)
- Reichweite (UMTS) ~3km
- UMTS-fähiges Endsystem (z.B. Smartphone)
- Auch: GSM; Zukünftig: LTE

<http://emf3.bundesnetzagentur.de/karte/Default.aspx>

Bsp: Zugangsnetz:
Kabelnetz mit WLAN Basisstation



[KuRo]



Trägermedien

Drahtgebunden

- Verdrillte Kupferkabel (Twisted Pair)
- Koaxialkabel
- Glasfaser

Drahtlos (Funk)

- Lokale Umgebung (geringe Reichweite)
- Weitere Umgebung

Kommunikations-Satelliten:

<https://www.youtube.com/watch?v=hXa3bTcIGPU>

- Geostationär (~36.000 km) – spürbare Verzögerung
 - [https://de.wikipedia.org/wiki/Internetzugang %C3%BCber Satellit](https://de.wikipedia.org/wiki/Internetzugang_%C3%BCber_Satellit)
Hohe Datenübertragungsraten möglich
 - Bsp: <http://www.inmarsat.com/about-us/our-satellites/> ; <https://en.wikipedia.org/wiki/Inmarsat>
<https://www.youtube.com/watch?v=4o5fXeu0ZyY>
 - <https://www.youtube.com/watch?v=auWxYsKpEb0>
 - <https://www.youtube.com/watch?v=HCTDHBygh-0>
- Erdnahe Umlaufbahn (LEO: Low Earth Orbit)
 - Bsp: Globalstar Constellation: <http://www.globalstar.com/en/index.php?cid=8300>
Globalstar Phone Commercial: https://www.youtube.com/watch?v=pOMhMb-2_C8
 - Bsp: Iridium: https://de.wikipedia.org/wiki/Iridium_%28Kommunikationssystem%29
Bsp: Iridium ermöglicht unterbrechungsfreie Kommunikation sogar in den Polarregionen mit der Außenwelt..
Geringe Datenübertragungsraten, aber Internetzugang ist möglich.
Iridium GO commercial (hotspot) <https://www.youtube.com/watch?v=YbCakX3QdUw>



Leitungs- vs. Paketvermittlung

Wiederholung:

Es gibt grundsätzlich 2 Arten der Vermittlung:

- Leitungsvermittlung
(z.B. „Fräulein vom Amt“)
- Paketvermittlung

Das Internet baut auf der Paketvermittlung auf.

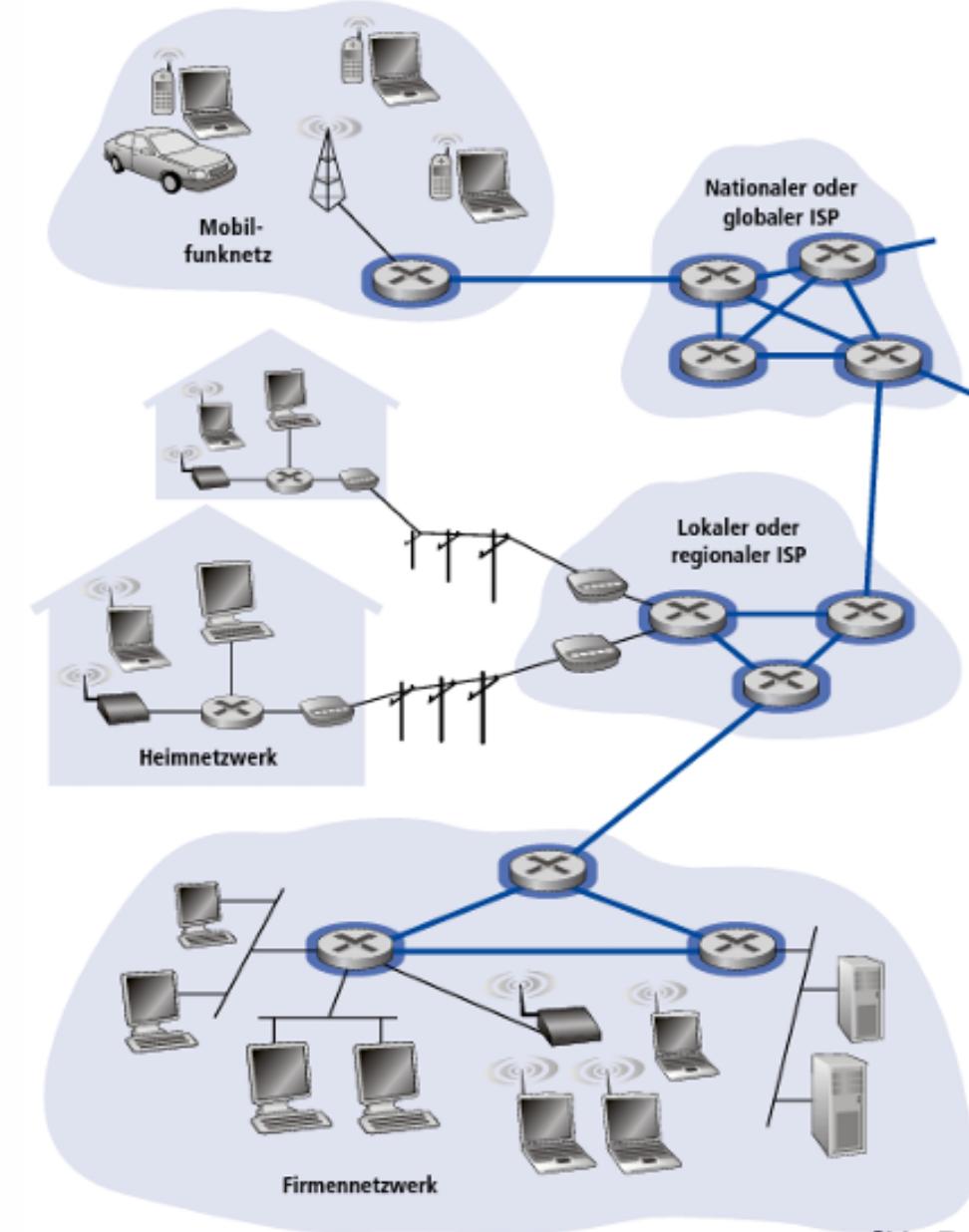
Prinzip:

- Alle Datenpakete werden ohne jegliche Reservierung ins Netz gesendet.
- Damit kann es passieren dass Pakete warten müssen, bis sie „an der Reihe“ sind, und weitergeschickt werden können.
=> Es kann somit eine Wartezeit entstehen (bei jedem Netzknoten)
- Das Internet „tut sein Bestes“ um die Pakete abzusenden, aber es kann nichts garantieren.
(Best Effort)

Analogie:

Restaurant nimmt keine Tischreservierung an:

Wenn Sie ankommen kann es passieren dass alle Tische belegt sind – und Sie müssen warten bis der erste Tisch frei wird.



[KuRo]



Wie gelangen Datenpakete durch das Internet?

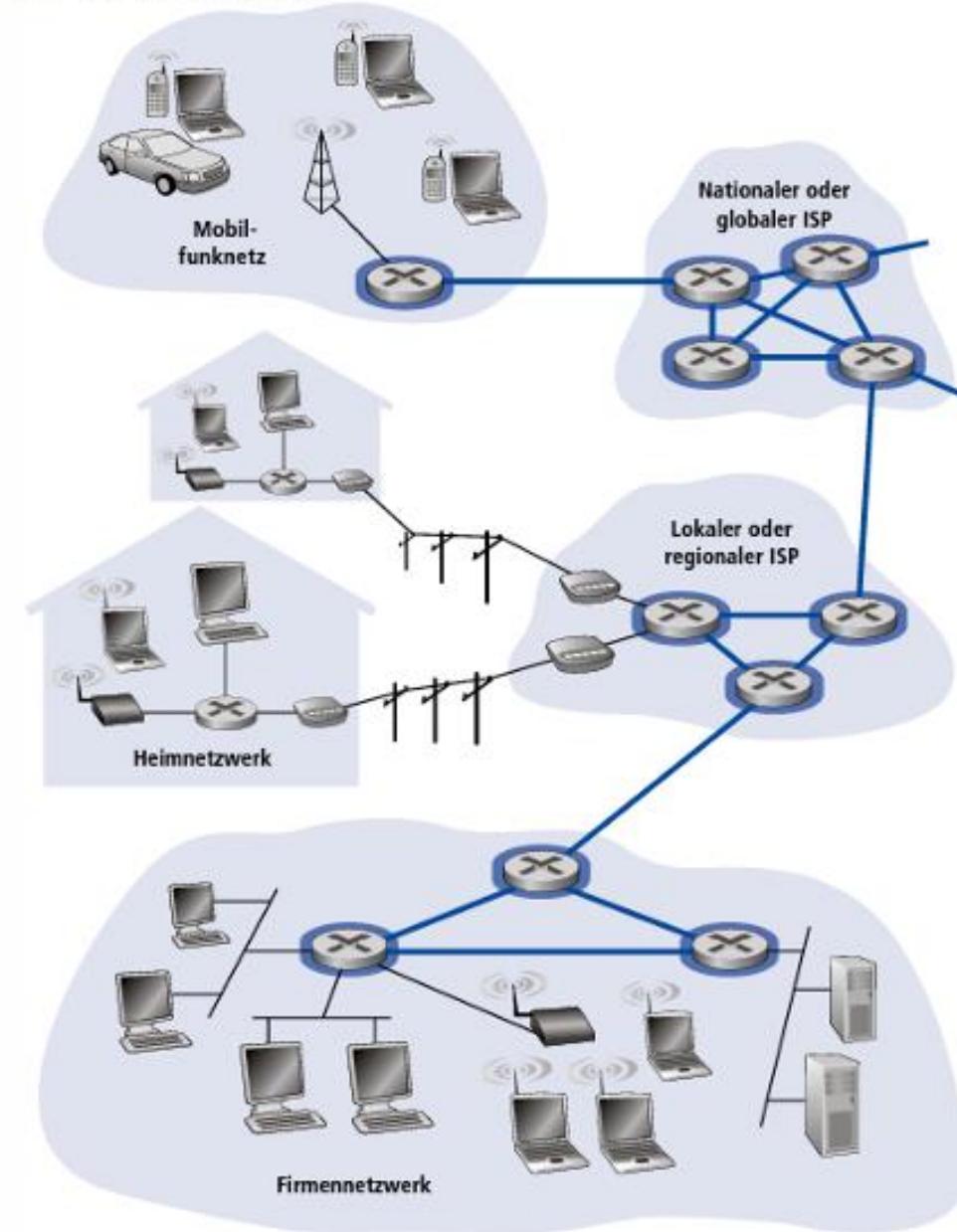
Weiterleitungstabellen (Routing Table)

Prinzip:

- Jeder Router (Netzknoten) besitzt eine Weiterleitungstabelle
- Zu allen Zieladressen (IP Adresse) ist gespeichert auf welche Ausgangsleitung der Router das Datenpaket weitersenden soll.

Analogie:

Autofahrer der ein Ziel hat und an JEDER Kreuzung nach der Richtung fragt.

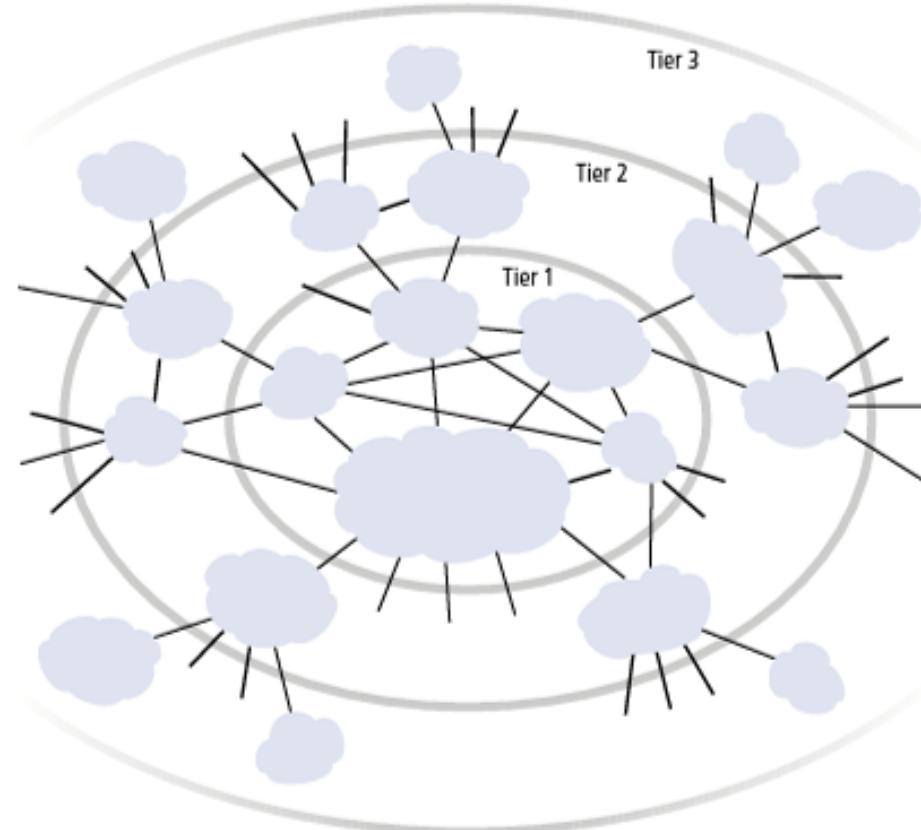


Internet Service Provider (ISP) & Internet-Backbone

Tier-1 ISP

https://en.wikipedia.org/wiki/Tier_1_network

- wird auch als Internet-Backbone bezeichnet
- Alle Tier-1 ISP sind direkt mit allen anderen Tier-1 ISP verbunden
- Alle Tier-1 ISP arbeiten international
- Tier-1 ISP sind mit vielen Tier-2 ISP verbunden
- Bsp: Deutsche Telekom
https://de.wikipedia.org/wiki/Tier_%28Netzwerke%29



Tier-2 ISP

- Ist mit einem Tier-1 ISP verbunden, selbst aber kein Tier-1 ISP
- ist ein Kunde eines Tier-1 ISP
- Kann auch mit einem anderen Tier-2 ISP verbunden sein
- Bsp: Vodafone
https://en.wikipedia.org/wiki/Tier_2_network

Tier-3 ISP:

- Zugangs-ISP
- ist mit einem Tier-2 ISP verbunden.
- i.d.R.: kleine regionale Provider: Sie sorgen für den Internetanschluß
- Bsp:
<http://www.swu.de/privatkunden/telekommunikation.html>

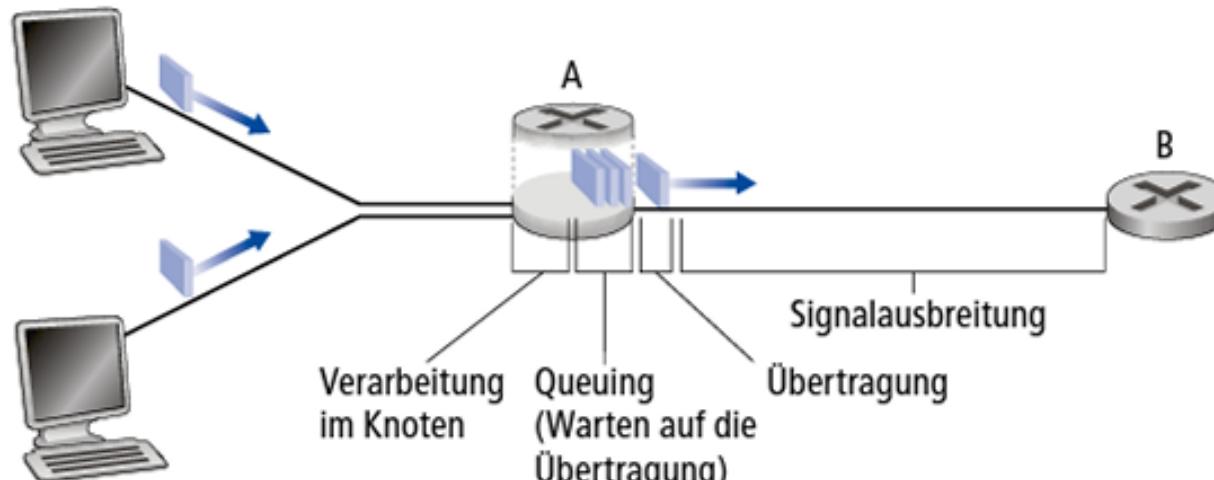
[KuRo]



Überblick über Verzögerungen in paketvermittelten Netzen

1. T_v (Verarbeitungszeit)

- Verarbeitungsverzögerung
Prüfung Paket-header & Entscheidung über den weiteren Weg.
=> Abhängig von Technik im Router
- Warteschlangenverzögerung
Paket ist der Ausgangsleitung zugewiesen - und wartet darauf versendet zu werden (bis es an der Reihe ist)
=> Abhängig von der gegenwärtigen Last



1. T_s (Sendedauer)

- Dies ist die Zeit die benötigt wird, um alle bits eines Datenpakets „auf die Leitung zu legen“
- L / R

2. T_p (Übertragungsdauer)

- Ist ein bit auf der Leitung angekommen, benötigt es T_p bis es beim Empfänger ankommt
- $l / cKanal$

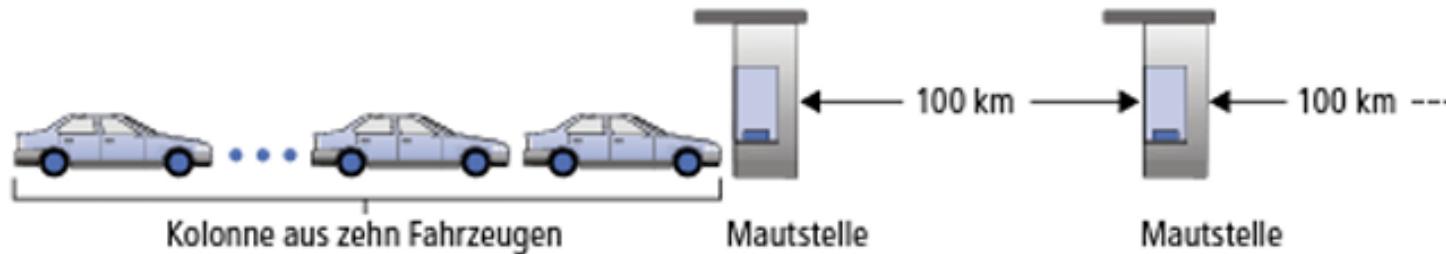
Gesamtverzögerung pro Router (inkl. zugehörigem link):
 $T_v + T_s + T_p$



Warum kommt es im Internet zu Paketverlusten?

Warteschlangenverzögerung

Analogie:



Kommen pro Zeit mehr Fahrzeuge an,
als pro Zeit die Mautstellen passieren können:

- ⇒ Dann bildet sich vor den Mautstellen eine immer größere Warteschlange – es kommt zum „Mega-Stau“.
- ⇒ Gleiches kann im Internet in einem Router geschehen.
- ⇒ Der Router verwirft dann einzelne Datenpakete – diese sind „einfach weg“. Man spricht von einem Paketverlust.



Schichtenarchitektur (Schichtenmodell)

Der fünfschichtige Internet-Protokoll-Stapel (IP-Stack)

- Netz-Protokolle werden in Schichten organisiert
- Prinzip:
 - Die höhere Schicht bedient sich der Dienste bzw. Protokolle der jeweils niederen Schicht.
- Vorteile der Schichtenarchitektur:
 - Teile & Herrsche:
Komplexität des Internet wird beherrschbar
 - Modularität:
Einzelne Komponenten sind leichter austauschbar, wartbar, erweiterbar,...



Schichtenarchitektur (Schichtenmodell)

Anwendungsschicht:

Netzbasierte Anwendungen + Protokolle

(z.B. HTTP Protokoll, als Basis für Webanwendungen)

⇒ Nachricht (Bsp: ein PDF File)

Transportschicht:

Transport von **Nachrichten** zwischen Endpunkten der Anwendungen
(TCP, UDP Protokoll)

⇒ Segment (Bsp: TCP-Segment)

Netzschicht:

Leitet **Segmente** über eine Reihe von Knoten im Netz, bis das Segment beim Ziel ankommt.

(IP-Protokoll)

⇒ Datagramm (Bsp: IP-Paket)

Sicherungsschicht:

Leitet **Datagramme** von einem Knoten zum nächsten Knoten
(Ethernet, WLAN,...)

=> Frames (Rahmen) (Bsp: Ethernet-Frame)

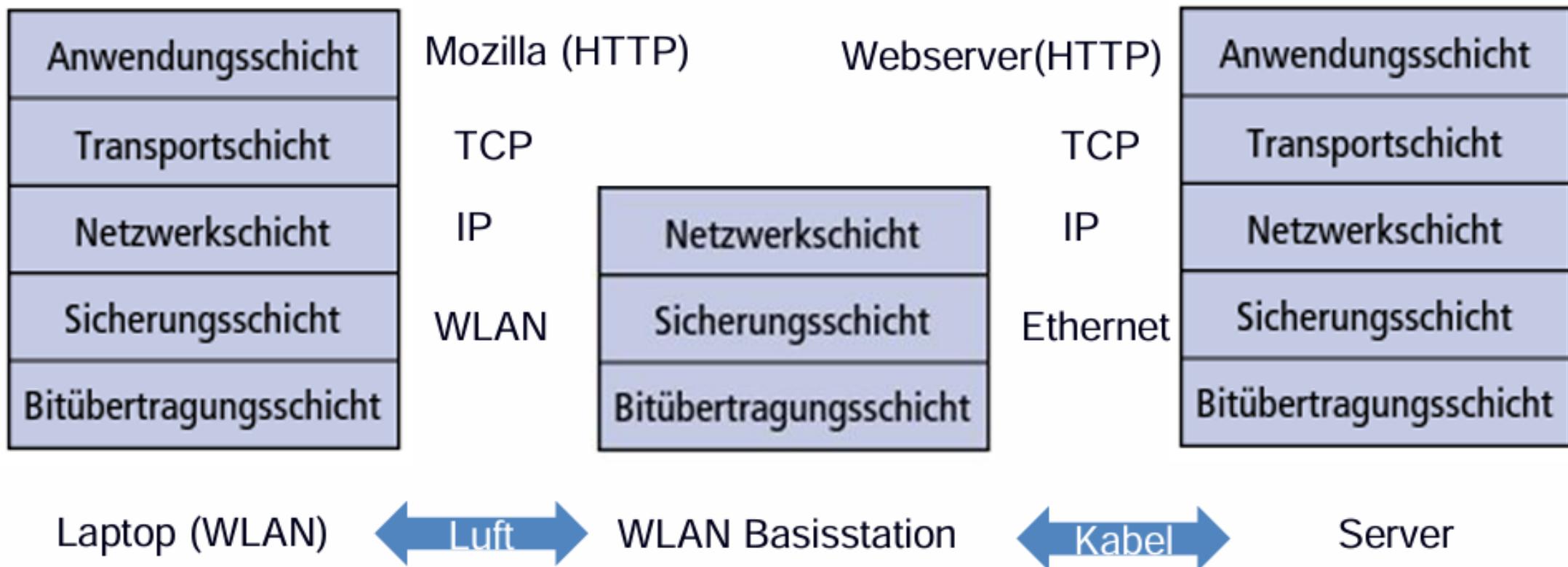
Bitübertragungsschicht:

Leitet ein einzelnes Bit eines **Rahmens** zum nächsten Knoten.
Bsp: Bitübertragung in Glasfaser, Kupfer, Luft,..



Schichtenarchitektur (Beispiel-Kommunikation)

- Laptop (WLAN)
- WLAN Basisstation (z.B. Fritzbox)
- Ein an die WLAN Basisstation mit Kabel verbundener Server



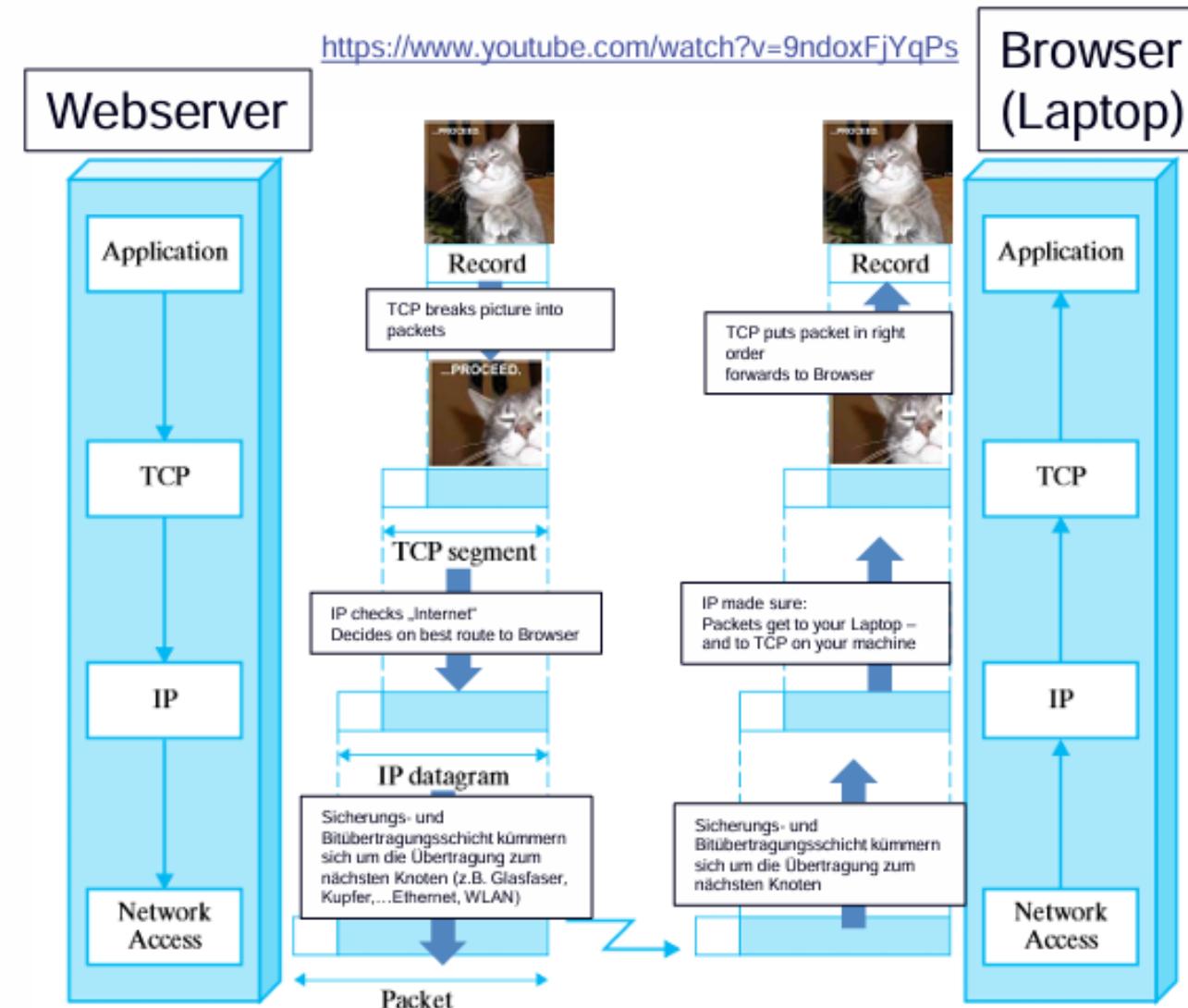
Nachrichten, Segmente, Datagramme (& Rahmen)

Prinzip:

- Die Nachricht (Bild im Bsp) wird an die darunterliegende Schicht übergeben.
- Die darunter liegende Schicht fügt in einem Header die notwendigen Informationen hinzu um die Aufgabe der Schicht zu erfüllen.

⇒ Header (Overhead) wächst kontinuierlich an bei der Übergabe an niedrigere Schichten

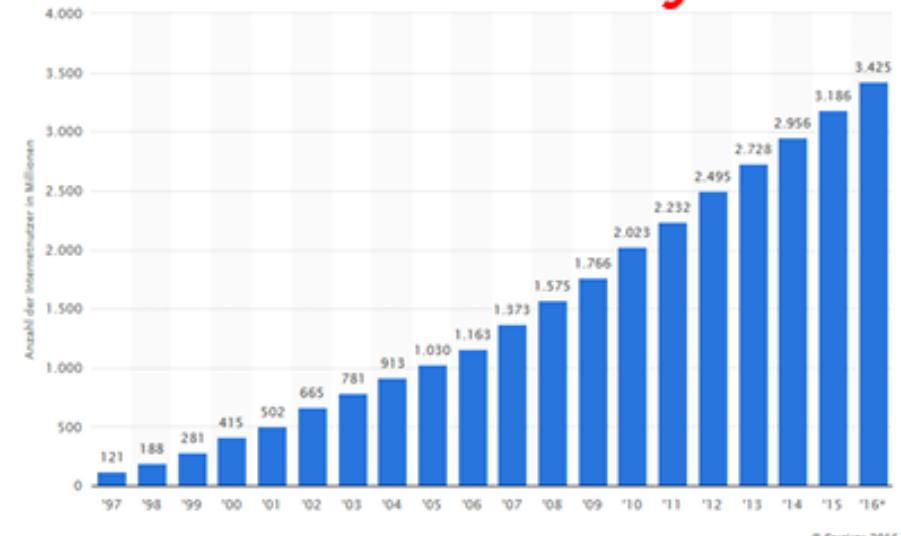
⇒ Auf Empfangsseite wird wieder schichtenweise „ausgepackt“.



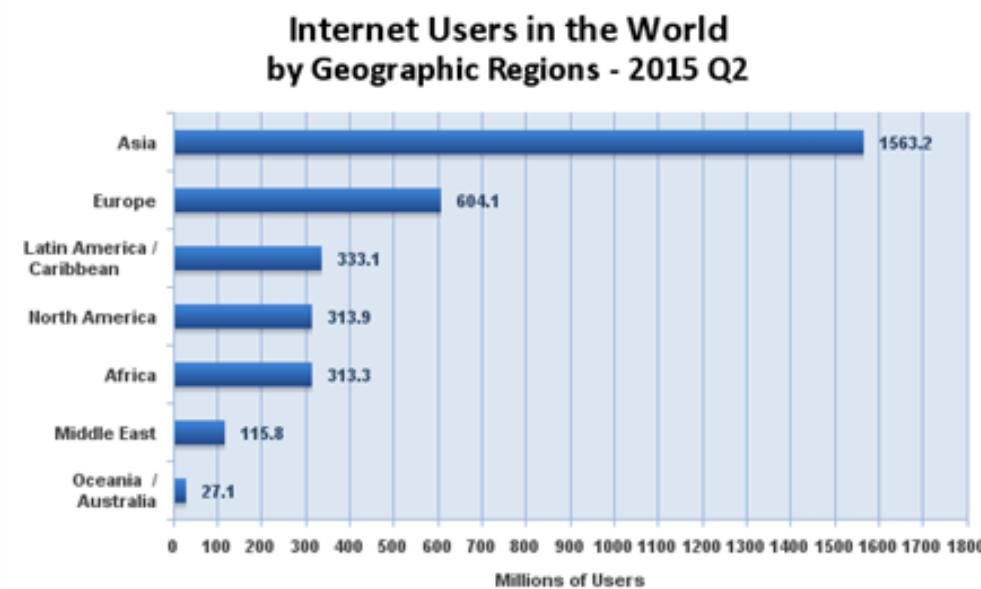
Jüngste Entwicklung

- Internet-(Bild-)Telefonie (Skype)
- Austausch von Videos (Youtube)
- Internetfernsehen
- Entwicklung im Mobilfunk ermöglicht die ununterbrochene Verbindung mit dem Internet
 - ⇒ Geburt des Smartphones: bereits ca. 1995
 - ⇒ Erste Schritte ca. bis 2005/06
 - ⇒ Smartphone Hype seit 2007: erstes iPhone...
- Peer-to-Peer: unabhängig von zentralen Servern.
- Internet der Dinge – alles wird vernetzt
- Mehr als 8 Milliarden Rechner, Endgeräte („Dinge“) sind mit dem Internet verbunden.
- <http://www.internetworldstats.com/stats.htm>
 - ca. 3,3 Milliarden Menschen haben Internetzugang
 - das sind ca. 45% der Weltbevölkerung

Statista:
Anzahl Internetnutzer weltweit von 1997 bis 2016



© Statista 2016



Selfstudium



Kommunikation zwischen Prozessen

Betriebssystem (z.B. Windows):

- Anwendungen bzw. „Programme“ werden durch das Betriebssystem als Prozesse behandelt.
- Diese Prozesse können kommunizieren:
 - auf der gleichen Maschine (Interprozesskommunikation)
 - auf unterschiedlichen Maschinen. (Netz-Anwendungen)

Eine Netzanwendung besteht aus:

- Prozesspaaren,
- die einander Nachrichten über ein Netzwerk zusenden.

Bsp: ein Client-Browser-Prozess tauscht Messages mit einem Webserver-Prozess aus.

Prozesse Leistung App-Verlauf Autostart Benutzer Details Dienste						
Name	Status	7% CPU	78% Arbeitss...	13% Datenträg...	0% Netzwerk	
System		3,1%	182,1 MB	0,4 MB/s	0 MBit/s	^
> Firefox (32 bit)		0%	79,4 MB	0 MB/s	0 MBit/s	
> Paint		0%	73,8 MB	0 MB/s	0 MBit/s	
> Adobe Digital Editions 4.5 (32 bit)		0%	66,4 MB	0 MB/s	0 MBit/s	
> Diensthost: Lokales System (Net...)		0%	55,6 MB	0 MB/s	0 MBit/s	
> Microsoft Outlook (32 bit)		0,8%	41,6 MB	0 MB/s	0 MBit/s	
> Microsoft PowerPoint (32 bit)		0%	40,0 MB	0 MB/s	0 MBit/s	
> Desktopfenster-Manager		1,6%	34,3 MB	0 MB/s	0 MBit/s	
> Microsoft Excel (32 bit)		0%	23,2 MB	0,1 MB/s	0 MBit/s	
> Windows-Explorer		0%	20,6 MB	0 MB/s	0 MBit/s	
> Diensthost: Lokales System (15)		0%	13,1 MB	0 MB/s	0 MBit/s	
> Task-Manager		1,0%	12,6 MB	0 MB/s	0 MBit/s	
> Diensthost: Lokaler Dienst (kein ...)		0%	12,2 MB	0 MB/s	0 MBit/s	
> Norton AntiVirus (32 bit)		0%	12,0 MB	0,1 MB/s	0 MBit/s	
> HP LaserJet Service (32 bit)		0%	10,2 MB	0 MB/s	0 MBit/s	▼

Weniger Details

Task beenden

Definition:

Der Prozess, der die Kommunikation eröffnet (also erstmals den anderen Prozess zu Beginn der Sitzung kontaktiert), wird **als Client bezeichnet**.

Der Prozess, der darauf wartet, zu Beginn einer Sitzung angesprochen zu werden, **ist der Server**.



Überblick HTTP (HyperText Transfer Protocol) (I)

- HTTP wird durch zwei Programme implementiert: ein Client-Programm und ein Server-Programm.
- Diese beiden, auf verschiedenen Endsystemen laufenden Programme kommunizieren miteinander durch den Austausch von HTTP-Nachrichten.
- HTTP definiert:
 - Struktur dieser Nachrichten
 - Art und Weise, wie Client und Server diese austauschen.

Terminologie:

- Eine (auch als Dokument bezeichnete) Webseite besteht aus Objekten.
- Ein Objekt ist:
 - einfach eine Datei (z.B. eine HTML-Datei, ein JPEG-Bild, ein Videoclip,...)
 - unter einer einzelnen URL (uniform resource locator) zu erreichen
- Jede URL hat zwei Bestandteile:
 - den Hostnamen des Servers, auf dem sich die Objekte befinden
 - sowie den Pfadnamen des Objekts.
- Die meisten Webseiten bestehen aus:
 - einer Basis-HTML-Datei
 - die auf mehrere weitere Objekte verweist.



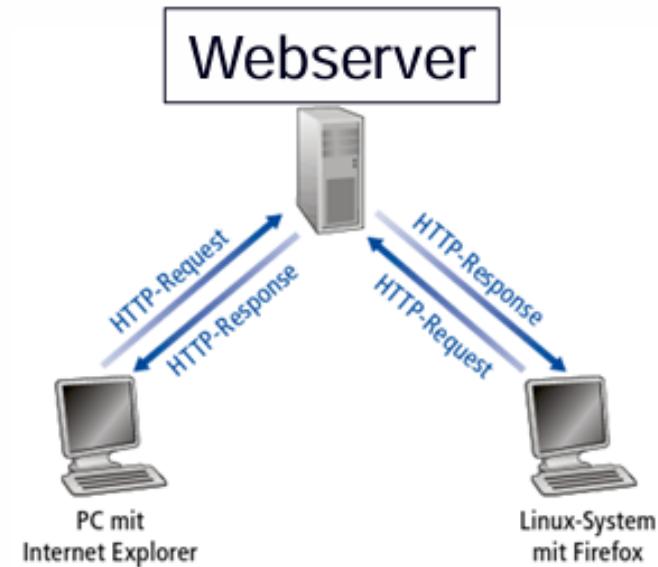
Überblick HTTP (HyperText Transfer Protocol) (II)

HTTP definiert:

- wie Webclients Webseiten von Webservern anfordern
- und wie Server die Webseiten zu den Clients übertragen.

Prinzip:

- Fordert ein Benutzer eine Webseite an, sendet der Browser **HTTP-Request-Nachrichten** für die Objekte auf der Webseite an den Server.
- Der Server erhält die Anforderungen und antwortet mit **HTTP-Response-Nachrichten**, welche die Objekte enthalten.
- HTTP verwendet TCP als zugrundeliegendes Transportprotokoll (und nicht UDP).
- Der HTTP-Client initiiert zuerst eine TCP-Verbindung zum Server.
- Sobald die TCP-Verbindung hergestellt ist, kann eine **HTTP-Response-Nachricht** gesendet werden



Grundsätzliches:

- Das Web verwendet die Client-Server-Anwendungsarchitektur
- Webserver ist immer online, besitzt eine feste IP-Adresse und beantwortet Anfragen von potenziell Millionen unterschiedlicher Clients (Bsp. www.spiegel.de)
- Webserver sendet die angeforderten Dateien an Client, ohne Informationen über den Zustand zu speichern
⇒ Fordert ein bestimmter Client innerhalb weniger Sekunden zweimal dasselbe Objekt an, sendet der Server das Objekt erneut: „Der Server merkt sich nicht, was er vorher getan hat.“
- ⇒ Weil ein HTTP-Server keine Information über die Clients behält, **bezeichnet man HTTP als zustandsloses Protokoll**.



Benutzer-Server Interaktion: Cookies

Ausgangslage:

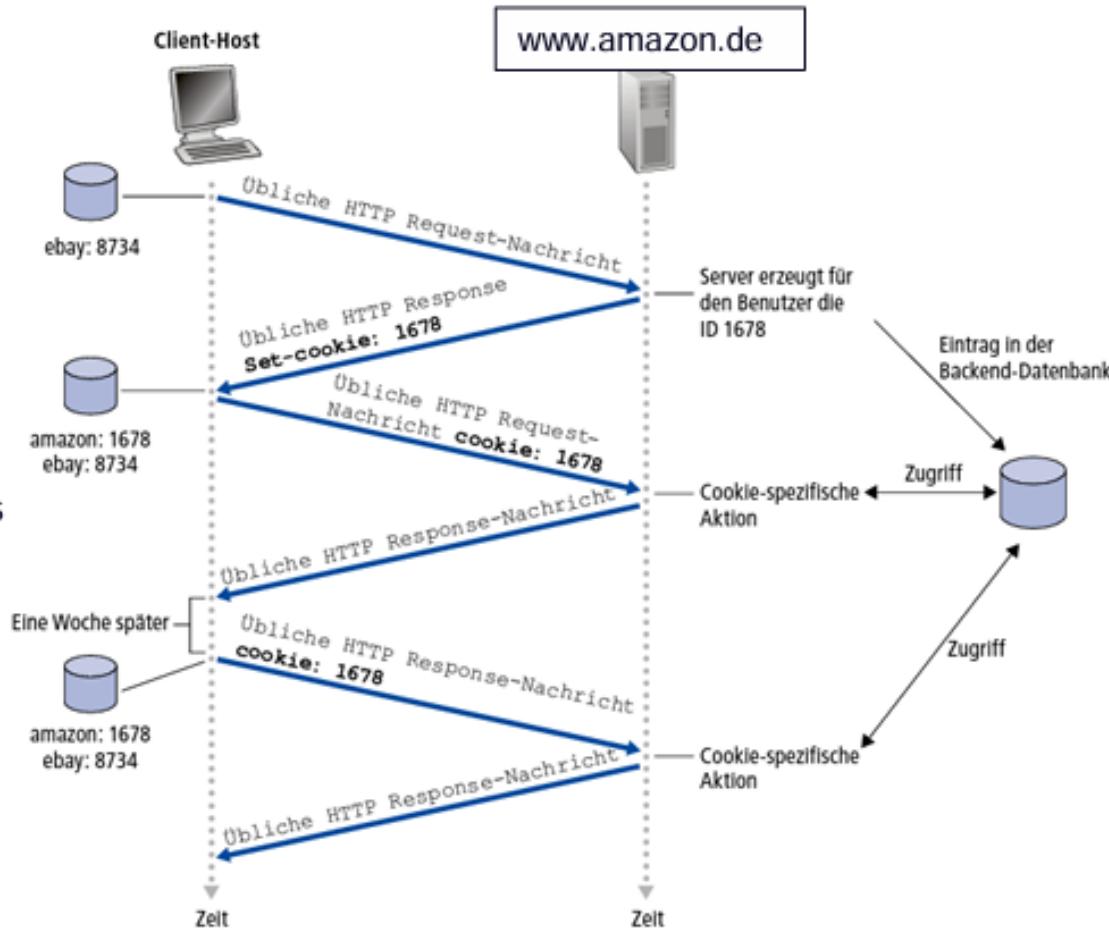
- Ein HTTP Server ist zustandslos.
„Der Server merkt sich nicht, was er vorher getan hat.“

Anforderung:

- es ist oft wünschenswert, dass eine Website Benutzer erkennt/identifiziert,
- entweder weil der Server den Benutzerzugang beschränken möchte,
- oder weil der Inhalt von der Identität des Benutzers abhängig ist.

Lösung:

- Erweiterung HTTP durch Cookies (RFC 2965)
- Cookies ermöglichen es den Websites, Benutzer wiederzuerkennen.
(Die meisten größeren kommerziellen Websites verwenden heute Cookies.)



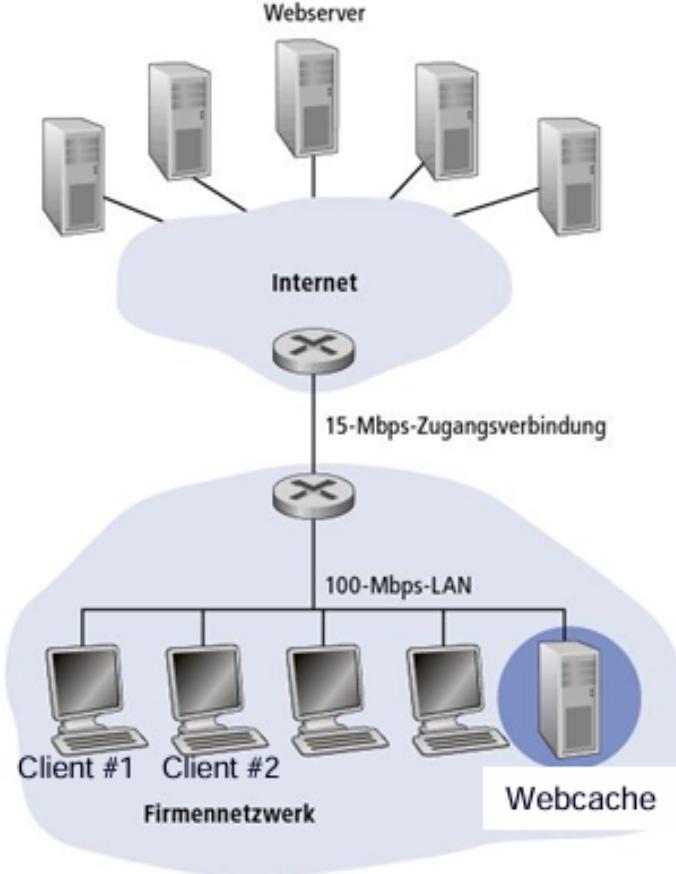
Webcaching (Proxy-Server)

Webcache:

- beantwortet im Namen des Webservers HTTP Requests von Clients
- hat eigenen Storage
- im Storage bewahrt er Kopien vor kurzem angeforderter Objekte auf („cached Objekte“)

Bsp:

1. Client #1 führt das erste mal einen HTTP Request nach einem bestimmten Objekts auf.
(z.B. ein Bild)
2. Der Webcache überprüft ob er das angefragte Objekt schon gespeichert hat.
3. Hat der Webcache das Objekt schon gespeichert, dann sendet er das Objekt direkt an den Client #1 zurück.
(ist in diesem Bsp nicht der Fall)
4. Hat der Cache das Objekt nicht gespeichert, dann führt der Webcache einen HTTP-Request aus, und holt das Objekt vom betreffenden Webserver – und speichert es lokal. Zusätzlich sendet er das Objekt an den Client #1
5. Fragt Client #2 dann nach dem gleiche Objekt => Punkt 2.



Vorteile:

- Geschwindigkeit
 - Übertragungsrate zw. Client und Cache i.d.R. hoch
 - RTT zw. Client und Cache i.d.R. klein
- Reduzierung des Datenverkehrs mit dem Internet: Firma spart Kosten (Client #1 und #2)



Bedingtes GET ?

Schwäche des Webcaching am Bsp:

1. Client #1 führt vor einer Woche einen HTTP-Request nach einem Objekt durch
2. Wie beschrieben: der Webcache holt das Objekt vom Webserver und speichert es lokal
3. Client #2 führt EINE Woche später, den gleichen HTTP-Request wie Client #1 durch.

Was ist ein möglicher Fehler ?

Deshalb: Bedingtes GET:

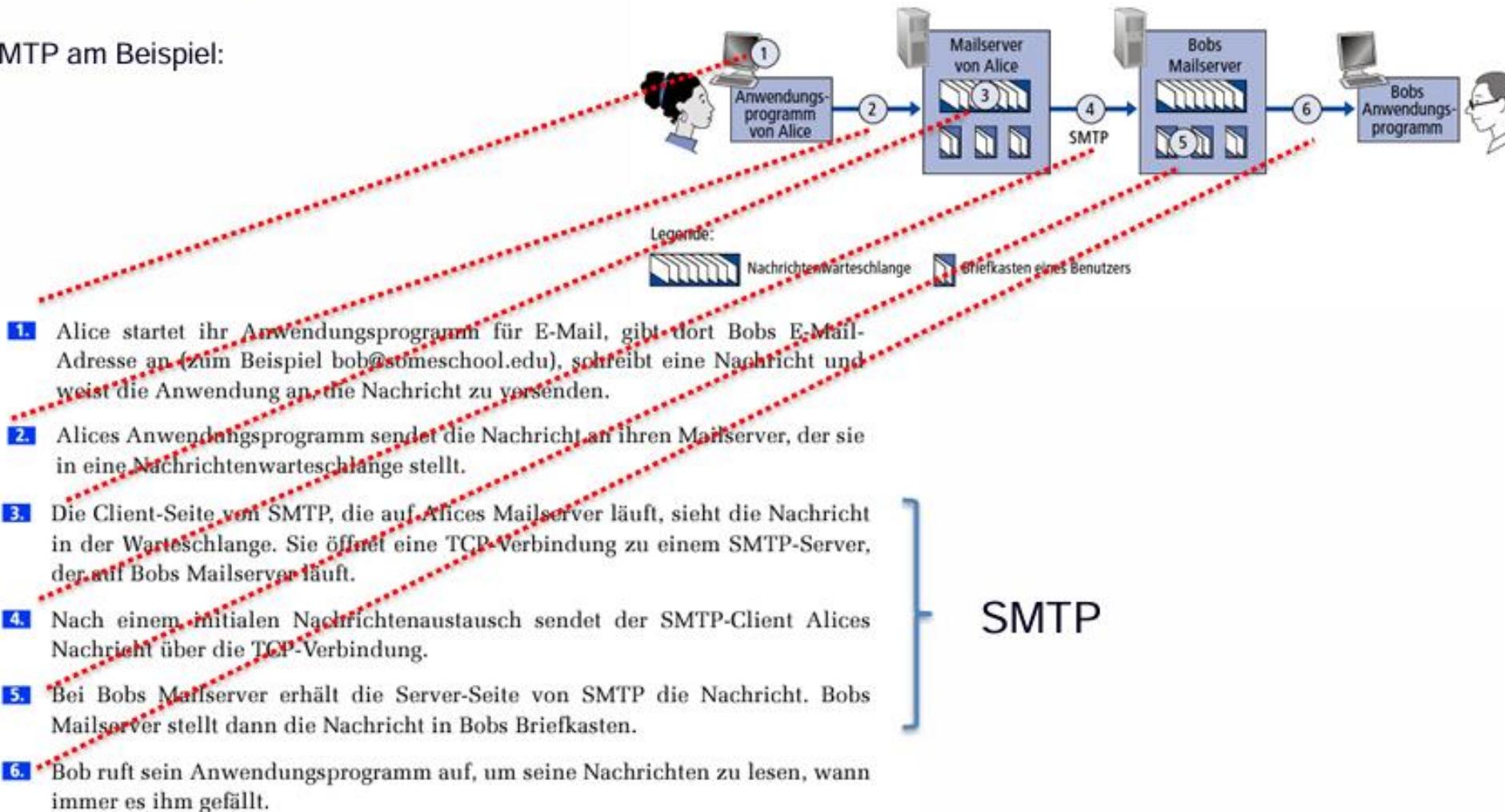
- In einem bedingten GET überprüft der Webcache beim Webserver ob das Objekt seit dem letzten download (vor einer Woche) verändert worden ist.
 ⇒ Ohne das Objekt zu übertragen!
- Ist das nicht der Fall dann nimmt der das lokal gespeicherte Objekt und schickt es an Client #2
- Ist das der Fall, dann holt der Webcache das veränderte Objekt vom Server.



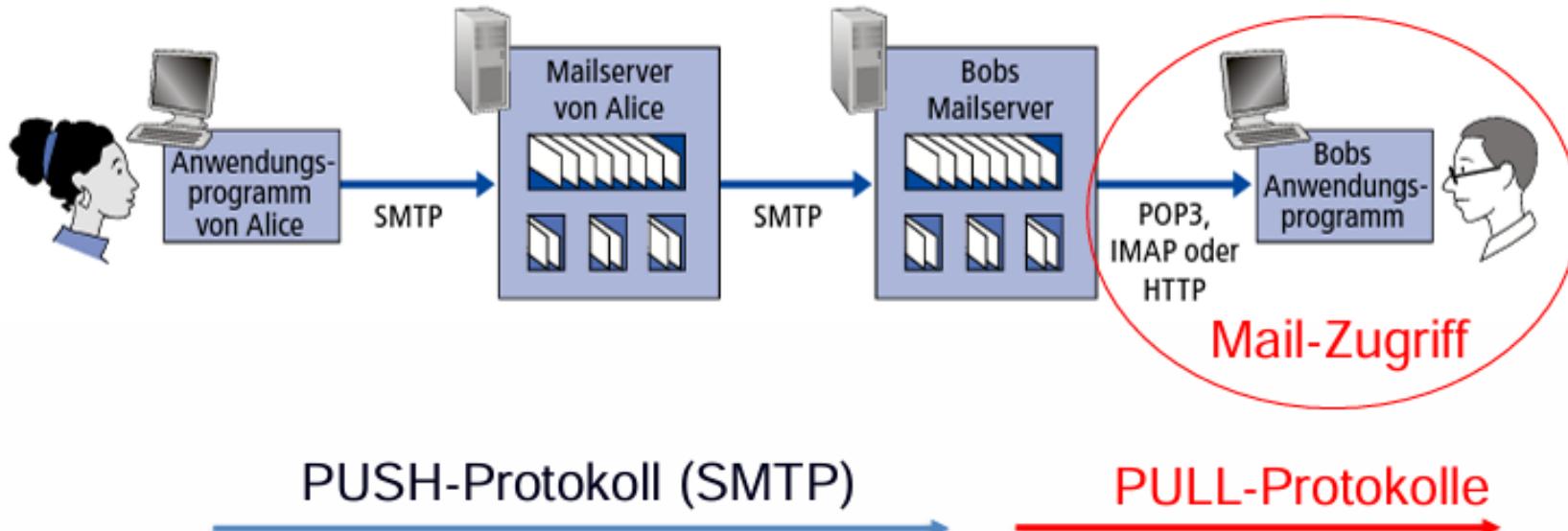
SMTP (Simple Mail Transfer Protocol)

https://de.wikipedia.org/wiki/Simple_Mail_Transfer_Protocol

SMTP am Beispiel:



Mail-Zugriffsprotokolle



Beispiel:
Konfiguration Outlook
für gmail
<https://www.youtube.com/watch?v=G9oi99--RCE>

- POP3:** https://de.wikipedia.org/wiki/Post_Office_Protocol
- Keine Benutzerordner auf Mailserver – nur lokale Ordner auf dem Endsystem
- IMAP:** https://de.wikipedia.org/wiki/Internet_Message_Access_Protocol
- Verbindet jede mail mit einem Ordner auf dem Mailserver
 - Benutzerorder auf Mailserver
 - Kann nur einzelne Bestandteile einer mail empfangen (bsp nur den header)
- HTTP:** https://de.wikipedia.org/wiki/Hypertext_Transfer_Protocol
- Hotmail startete damit Mitte der '90er
 - Heute Standard: Mails werden im Browser dargestellt



Vom DNS erbrachter Dienst

- Hauptaufgabe des Domain Name System (DNS) des Internets:
Verzeichnisdienst der Hostnamen in IP-Adressen übersetzt.
- Das DNS ist
 - eine verteilte Datenbank, die auf einer Hierarchie von DNS- Servern basiert
 - ein Anwendungsschichtprotokoll, das es Hosts ermöglicht, die verteilte Datenbank abzufragen.
- Bestandteile DNS:
 - DNS-Server
 - oft Unix-Computer, auf denen das Programm Berkeley Internet Name Domain (BIND) läuft [BIND 2007].
 - Das DNS-Protokoll setzt UDP ein und benutzt Port 53
- Nutzung:
 - DNS wird intensiv in Kombination mit anderen Anwendungsschichtprotokollen verwendet: z.B. HTTP, SMTP,...
 - Übersetzt vom Benutzer zur Verfügung gestellte Hostnamen in IP-Adressen.
- **Funktionalität aus Endbenutzer-Sicht: am Beispiel**



Weitere vom DNS erbrachte Dienste

Host-Aliasing, bzw. Mailserver-Aliasing

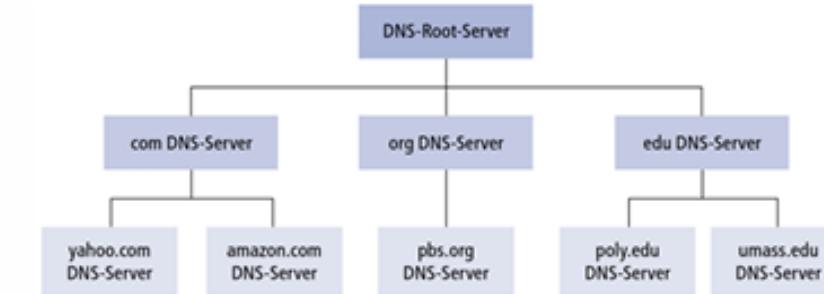
- Prinzip:
 - Ein Host (Rechner) kann mehrere Namen haben
- Bsp
 - www.hs-neu-ulm.de
 - hs-neu-ulm.de
 - 194.95.20.82

Die verteilte Datenbank des DNS enthält diese Info.



Überblick Arbeitsweise DNS (I)

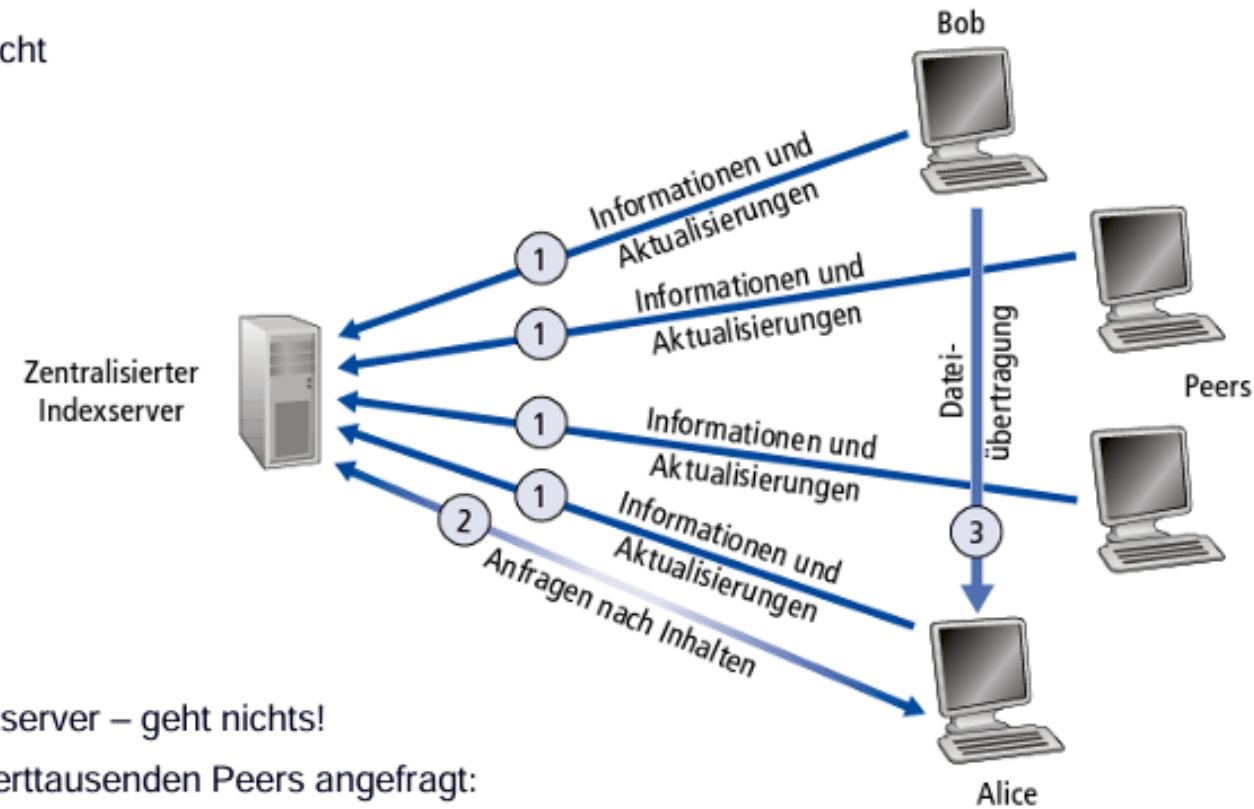
- verteilte, hierarchische Datenbank
- Skalierbarkeit:
 - DNS benutzt eine große Anzahl von Servern
 - die hierarchisch organisiert
 - über die ganze Welt verteilt sind.
 - **Es gibt keinen einzelnen DNS-Server, der alle Übersetzungen für alle Hosts im Internet enthält**
 - Stattdessen sind die Übersetzungen über alle DNS-Server verteilt.
 - **Drei Klassen von DNS-Servern**
 - DNS-Root-Server,
 - Top-Level-Domain-DNS Server (TLD, Domains der obersten Hierarchiestufe)
 - und authoritative DNS-Server.



P2P Community: Suchen von Informationen (I)

Zentralisierter Index: (Bsp: Napster)

- Indexdienst wird von einem Server erbracht
- eine neu gestartete P2P Anwendung informiert den Indexserver über:
 - die eigene IP-Adresse
 - Und welche Dateien zur Verteilung bereitstehen
- Indexserver sammelt diese Information von ALLEN Peers:
=> zentralisierter Index



Nachteile:

- Eine zentrale Schwachstelle: ohne Indexserver – geht nichts!
- Ein zentraler Server: wird evtl. von hunderttausenden Peers angefragt: Performance-Bottleneck...

Disclaimer: diese Vorlesung behandelt keine rechtlichen Themen wie Urheberschutz.

Natürlich gilt bei P2P Netzen – wie überall im Leben: Gesetze beachten!

How did Napster work?
<https://www.youtube.com/watch?v=7AF18DUIH1Y>

Official Napster Documentary Trailer
<https://www.youtube.com/watch?v=6Ai6K2VIEXM>

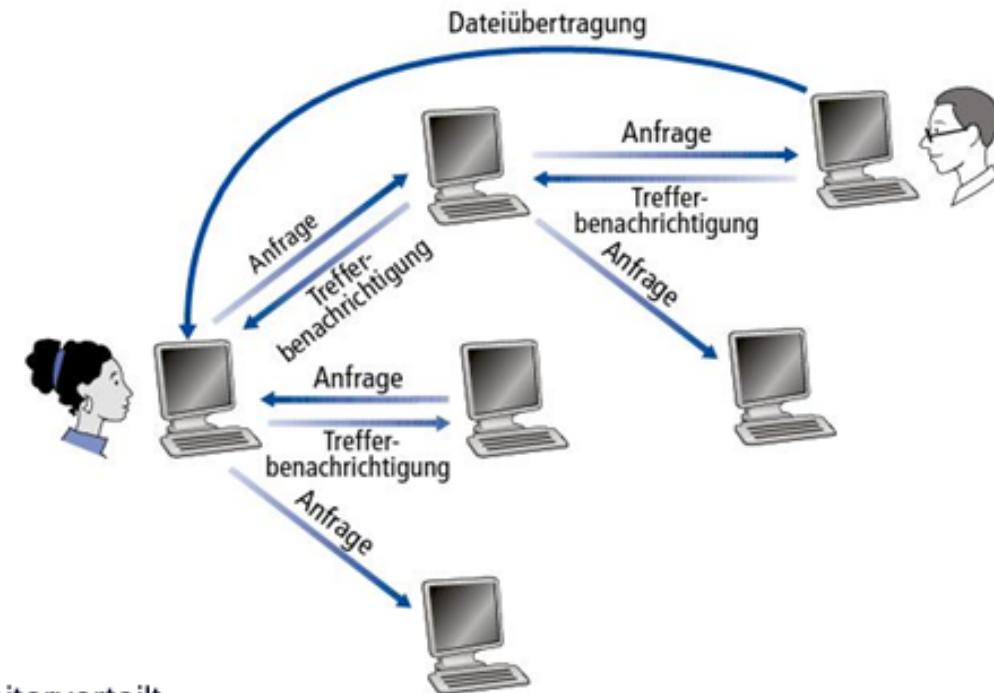


P2P Community: Suchen von Informationen (II)

Dezentraler Ansatz:

Anfrage Fluten:

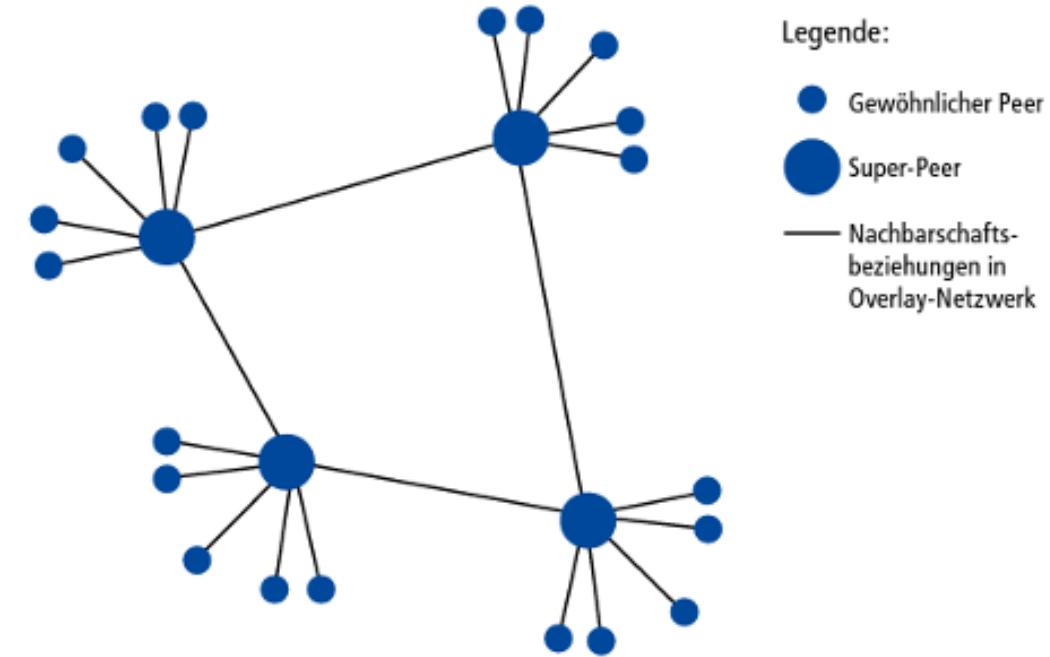
- kein zentraler Index
- Jeder Peer indiziert nur die Dateien die er selbst zur Verteilung freigibt.
- Prinzip:
 - Suchanfrage nach einer Datei wird an alle Nachbarn weitergeleitet
 - Die Nachbarn:
 - haben das gesuchte file im eigenen Index:
=> positive Antwort
 - haben das gesuchte file nicht im eigenen Index:
Suchanfrage wird ebenfalls an alle Nachbarn weiterverteilt.
- Resultat: Peers werden gefunden, die das gewünschte File lokal gespeichert haben



P2P Community: Suchen von Informationen (III)

Hierarchisches Overlay-Netz (moderne P2P-Netze)

- kein einzelner Indexserver
- unterschiedliche Arten von Peers:
 - Super-Peers
(schneller Internetanschluss & hohe Verfügbarkeit, aber ein „normaler“ Rechner, z.B. Laptop)
 - gewöhnlicher Peer
„Child eines Super-Peers“,
(ebenfalls ein „normaler“ Rechner, aber erfüllt nicht die Kriterien eines Super-Peer)
 - ein Super-Peer kann bis zu einige hundert Peers als Children verbinden
- Super-Peer hat Index über seine Children
- Super-Peers sind untereinander verbunden



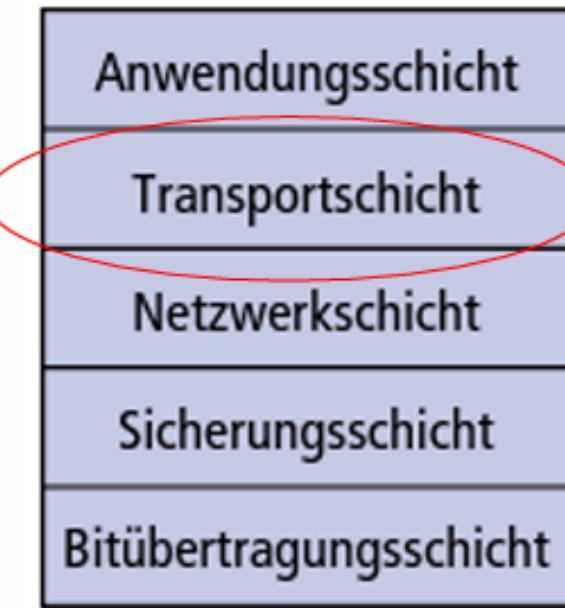
⇒ Betrifft nur die Suche von Information

⇒ Ist die Information gefunden: dann P2P Datenaustausch.



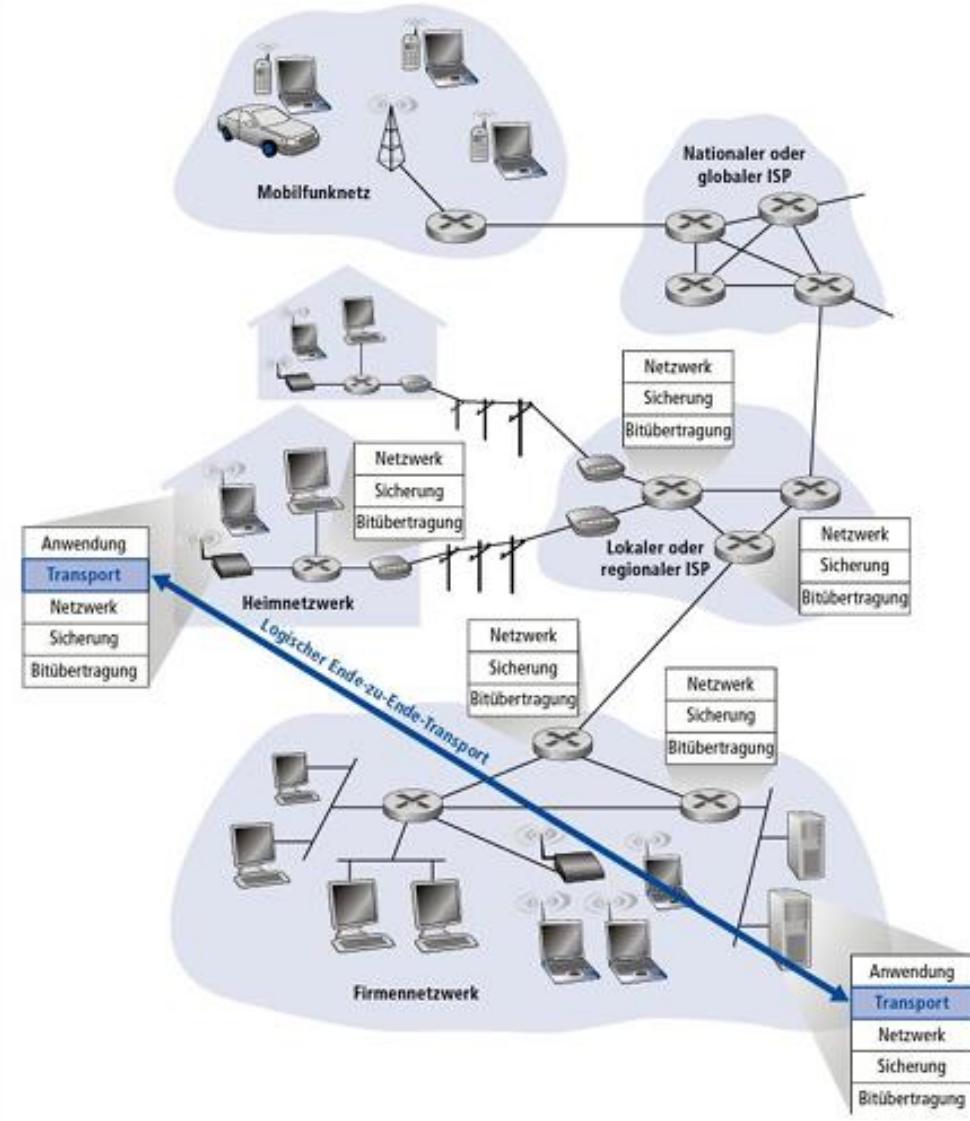
1) Einführung und Transportschichtdienste

- Transportschicht ein zentrales Element der geschichteten Netzwerkarchitektur.
- Aufgabe:
 - den **Anwendungsprozessen auf verschiedenen Hosts (Computern)** Kommunikationsdienste zu erbringen.
⇒ „Beziehung“ zur Anwendungsschicht
 - Erweiterung der Kommunikation zwischen zwei Endsystemen (wie sie die Netzwerkschicht anbietet) hin zur: Kommunikation zwischen zwei auf diesen Endsystemen ablaufenden Anwendungsschichtprozessen.
⇒ „Beziehung“ zur Netzsicht.
 - Verbindungsloses Transportprotokoll: UDP illustrieren
 - Wiederholung: Wie können zwei Kommunikationsteilnehmer sicher über ein Medium kommunizieren, das Daten verlieren oder verändern kann? (Protokolle)
 - verbindungsorientiertes Transportprotokoll des Internets: TCP
 - Kontrolle der Übertragungsrate von Instanzen der Transportschicht, um Überlast im Netzwerk zu vermeiden oder abzubauen.



Aufgabe von Transportschichtprotokollen

- logische Kommunikation zwischen Anwendungen, die auf verschiedenen Hosts (Computern) laufen:
 - Aus Sicht einer Anwendung (z.B. Browser – Webserver) erscheint es als wären die Hosts direkt miteinander verbunden
 - Hosts können aber tatsächlich durch zahlreiche Router verbunden sein.
 - Anwendungsprozesse benutzen die logische Kommunikation mittels der Transportschicht, um sich gegen seitig Nachrichten zu zusenden, ohne sich um Details der physikalischen Infrastruktur kümmern zu müssen, die diese Nachrichten übertragen.
- Transportschichtdienste/ Protokolle sind nur in den Endsystemen implementiert => nicht in Routern!
- Prinzip:
 - Senders verpackt in der Transportschicht die Nachrichten, die er von einer sendenden Anwendung (z.B. Webserver) empfängt, in Segmente.
 - Wenn nötig (fast immer) wird dabei die Nachrichten in kleinere "Stücke" aufgeteilt werden. (Bsp: Bild der Katze)
 - Anschließend wird jedem "Stück" ein Header hinzugefügt. Dadurch entsteht das Segment.
 - Auf Empfängerseite wird aus den einzelnen Segmente die ursprüngliche Nachricht wieder zusammengesetzt – und an die Anwendung auf Empfängerseite (z.B. Webbrower) weitergereicht. (Der Brower stellt dann das Katzenbild dar)



Überblick über die Transportschicht im Internet

Im Internet (einem sogenannten TCP/IP-Netz) gibt es nur zwei verschiedene Transportschichtprotokolle:

- UDP (User Datagram Protocol)
stellt den aufrufenden Anwendungen einen unzuverlässigen, verbindungslosen Dienst zur Verfügung.
- TCP (Transmission Control Protocol)
stellt den aufrufenden Anwendungen einen zuverlässigen, verbindungsorientierten Dienst an.
- Entwickler von verteilten Anwendungen müssen sie sich auf eines dieser beiden Protokolle festlegen.
- Bezeichnung eines Datenpakets der Transportschicht: **Segment** bezeichnen.

wichtigste Aufgaben von UDP und TCP:

- Erweitern des IP-basierten Zustelldienstes zwischen zwei Host (Endsystemen) auf einen Zustelldienst zwischen zwei Prozessen, die auf den entfernten Endsystemen laufen.
- Die Erweiterung der **Host-zu-Host-Zustellung** auf **Prozess-zu-Prozess Zustellung** wird als:
 - Transportschicht-Multiplexing
 - und Transportschicht-Demultiplexingbezeichnet.
- Integritätsüberprüfungen, indem sie in die Header der Segmente Felder für die Fehlererkennung einfügen.

Weitere Aufgaben von TCP (im Vgl. zu UDP):

- zuverlässiger Datentransfer:
 - Mithilfe von Flusskontrolle, Sequenznummern, Acknowledgments und Timern stellt TCP sicher, dass die Daten vom sendenden Prozess korrekt und in der richtigen Reihenfolge an den empfangenden Prozess geliefert werden.
 - TCP wandelt auf diese Art den unzuverlässigen Dienst zwischen Endsystemen (der durch IP erbracht wird) in einen zuverlässigen Datentransportdienst zwischen Prozessen um.
 - Überlastkontrolle:
 - Dienst für das Internet als Ganzes, ein Dienst für das „Allgemeinwohl“.
 - Die TCP-Überlastkontrolle hindert eine TCP-Verbindung daran, die Verbindungen und Router zwischen den kommunizierenden Hosts mit einem unverhältnismäßigen Verkehrsaufkommen zu überschwemmen.
 - Fairness: TCP ist bestrebt, jeder Verbindung, die einen überlasteten Link durchquert, einen gleich großen Anteil der Verbindungsbandbreite zuzuteilen.
- ⇒ TCP ist ein komplexes Protokoll

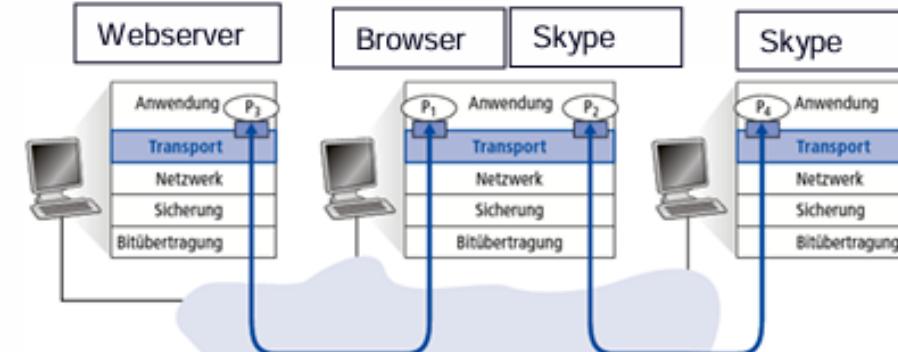
UDP hat als einfaches Protokoll keine weiteren Aufgaben!



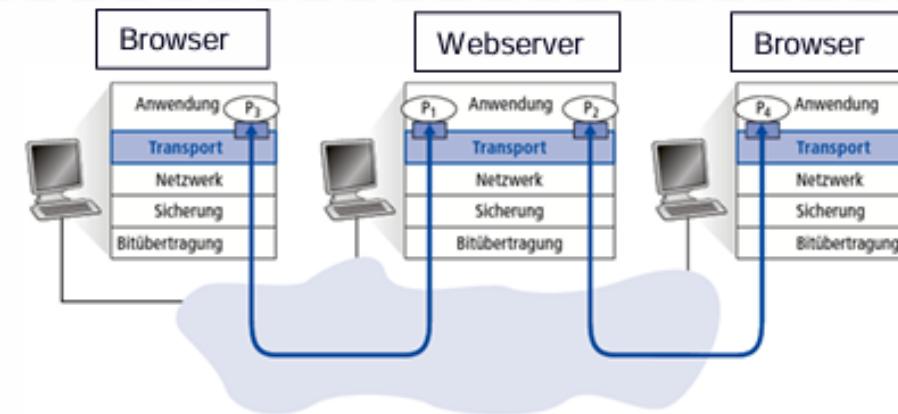
2) Multiplexing und Demultiplexing

Herausforderungen:

1. Wie kommunizieren mehrere Anwendungen auf einem Host (z.B. Browser, Skype,...) mit unterschiedlichen Hosts bzw. Anwendungen im Internet?



2. Wie kommuniziert eine Anwendung auf einem Host (z.B. Webserver) mit unterschiedlichen Hosts (bzw. den Webbrowsersn) ?



Lösung: Transportschichtprotokolle

- Erweiterung des von der Netz-Schicht angebotenen Host-zu-Host Zustelldienstes zu einem Prozess-zu-Prozess-Zustelldienst für Anwendungen, die auf den Hosts laufen.
- Am Zielhost erhält die Transportschicht Segmente von der direkt darunterliegenden Netzwerkschicht.
- Die Transportschicht hat die Aufgabe, die Daten in diesen Segmenten an die entsprechenden, im Host laufenden, Anwendungsprozesse zu liefern.



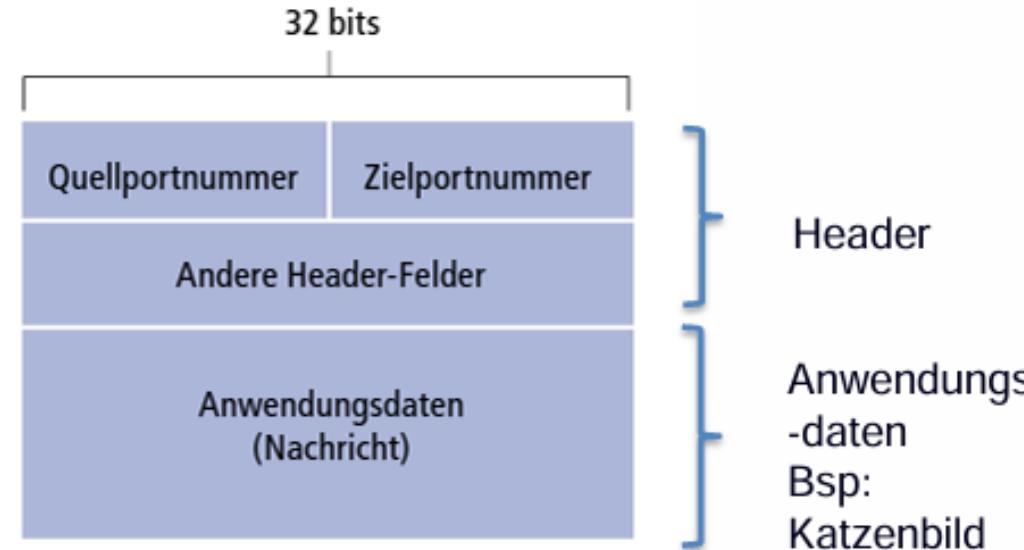
Wie läuft MUX/ DEMUX praktisch ab?

- Transportschicht-Multiplexing erfordert eine eindeutige Kennzeichnung jedes Sockets
- Diese Kennzeichnung ist in den Feldern des Headers festgelegt.
- ⇒ Header identifiziert eindeutig den Socket.
- Felder eines Headers in der Transportschicht:
- Portnummerfeld der Quelle (source port number field)
 - Portnummerfeld des Ziels (destination port number field)
 - Jede Portnummer ist eine 16 Bit-Zahl, die zwischen 0 und 65535 liegt.
 - Reservierung: Wellknown Port Numbers;
 - zwischen 0 und 1023: Reservierung für weit verbreitete Anwendungsprotokolle
 - Bsp: HTTP (Portnummer 80)
 - https://en.wikipedia.org/wiki/List_of_TCP_and_UDP_port_numbers#Well-known_ports
 - Hinweis: UDP- und TCP-Segmente: noch weitere Felder

Ablauf: DEMUX: Empfänger (Bsp. Browser)

- Dem Browser (bzw. Prozess) wird eine entsprechende Portnummer zugeteilt.
- Sobald ein Segment beim Host ankommt:
- prüft die Transportschicht die Zielportnummer im Segment
- und leitet das Segment zu dem entsprechenden Socket.
- Die Daten des Segmentes gehen dann durch den Socket in den dazugehörigen Prozess.
(bei TCP laufen noch weiter Mechanismen ab:
Reihenfolge,...)

Aufbau Transportschicht-Segment



Inversion von Quell- und Ziel-Portnummern

Entstehung der Portnummer bei Host A:

- Der Client-Prozess (dh. Ein Anwendung auf Host A, z.B. Webbrowser) instantiiert im Programm einen Socket:
- Die Anwendung weit diesem Socket eine Portnummer zu
Bsp Portnummer: 19157

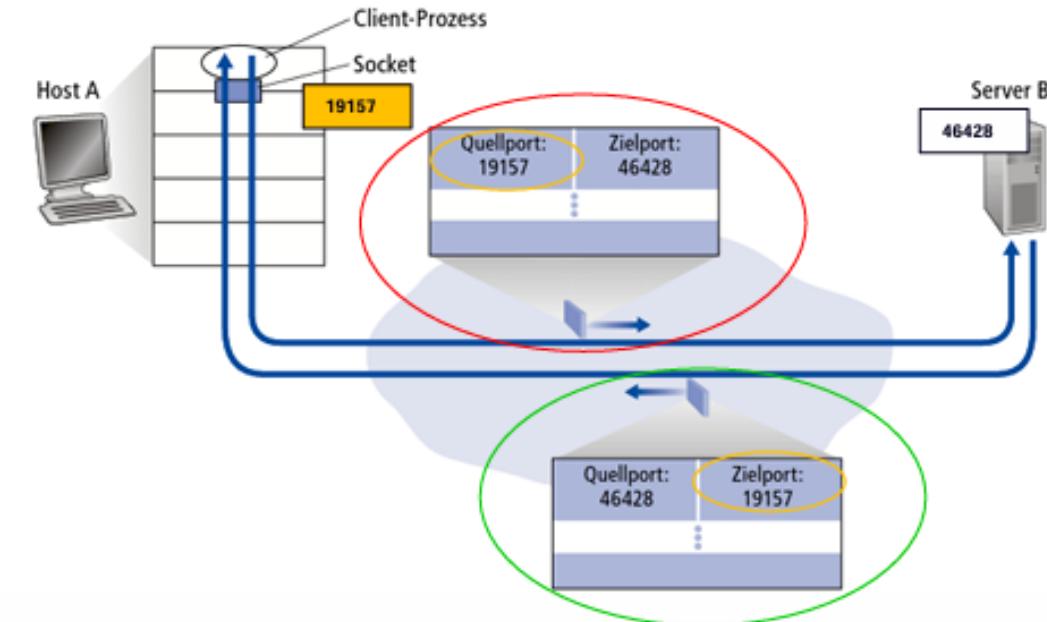
Entstehung der Portnummer beim Server B

- Der Server-Prozess (dh. Eine Anwendung auf Host A, z.B. Webserver) instantiiert im Programm ebenfalls einen Socket:
- Die Anwendung weit diesem Socket eine Portnummer zu
Bsp Portnummer: 46428

2 Arten Prozesse zu unterscheiden: („Sockets“)

1. UDP-Sockets

- werden eindeutig charakterisiert durch:
 - Eigene Portnummer
 - Eigene IP Adresse
- Bsp. Der Socket bei Host A ist eindeutig festgelegt:**
 - Port Nummer: 19157
 - IP-Adresse von Host A
- Der Socket (19157) auf Host A kann zu jedem Computer und zu JEDEM Prozess Segmente verschicken
- Der Socket (19157) auf Host A akzeptiert von jedem Computer und JEDEM Prozess Segmente (die an 19157 geschickt worden sind)



2. TCP-Sockets

- werden eindeutig charakterisiert durch:
 - Eigene Portnummer
 - Fremde Portnummer
 - Eigene IP Adresse
 - Fremde IP Adresse
- Der Socket auf Host A kann Segmente nur nach Server B UND nur zum Prozess 46428 senden
- Der Socket auf Server B akzeptiert nur Segmente von Host A UND dem Socket 19157



MUX / DEMUX in TCP Sockets

Recalling: TCP Sockets werden eindeutig charakterisiert durch:

1. Eigene Portnummer
2. Fremde Portnummer
3. Eigene IP Adresse
4. Fremde IP Adresse

Ein TCP Verbindungs-Socket wird durch diese vier Werte identifiziert:

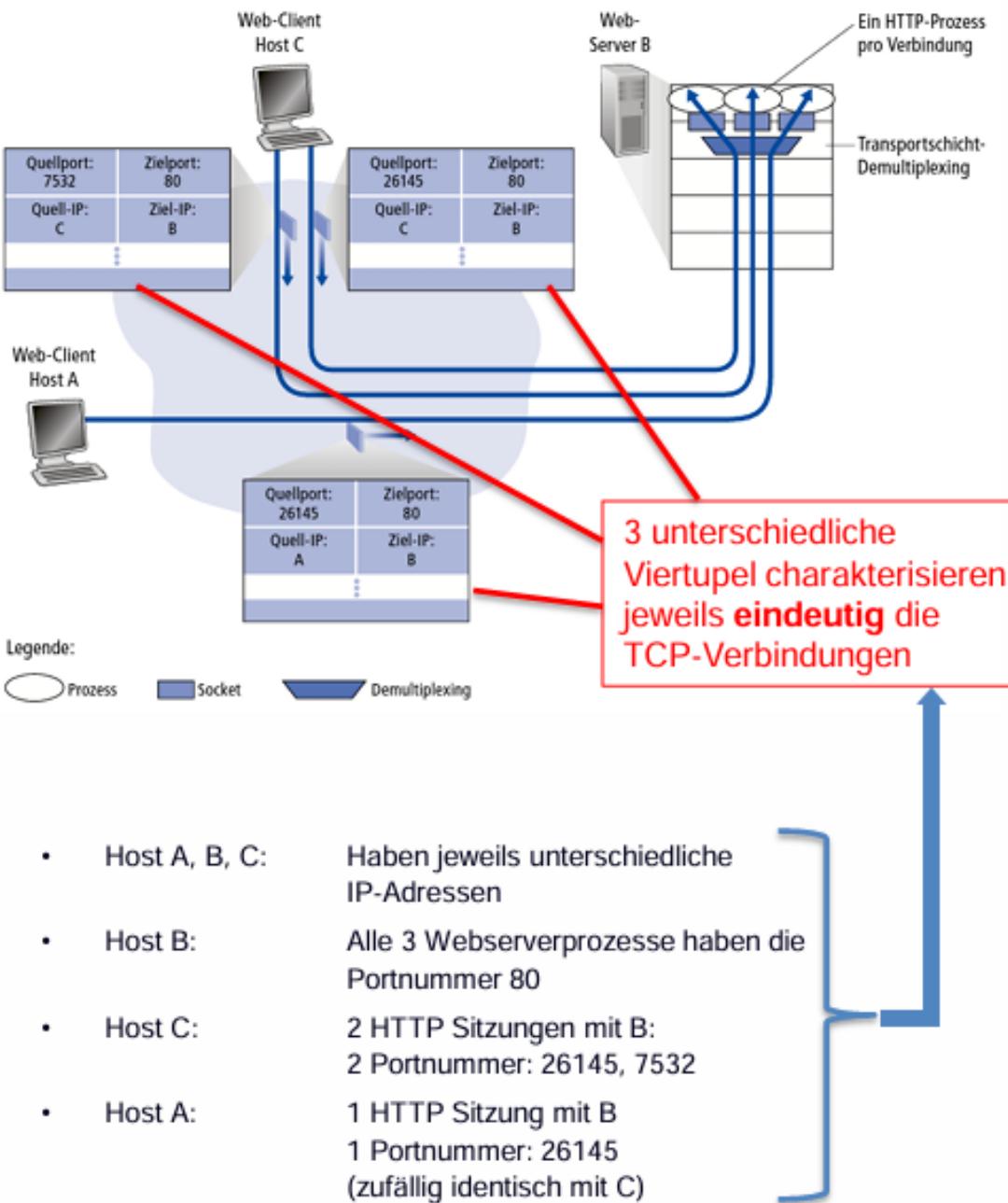
Alle eintreffenden TCP-Segmente deren:

- ⇒ Quellport,
- ⇒ Quell-IP-Adresse
- ⇒ Zielport
- ⇒ Ziel-IP-Adresse

mit diesen vier Werten übereinstimmen, werden auf diesen Socket geleitet.

- ⇒ Man spricht von einer bestehenden TCP-Verbindung. Darüber können Client-Anwendung und Server-Anwendung Daten austauschen.
- ⇒ Der Serverhost kann gleichzeitig viele TCP-Sockets verwalten, wobei jeder Socket einem Prozess zugeordnet ist und jeder durch sein eigenes Viertupel (Quell-Port, Quell-IP, Ziel-Port, Ziel-IP) identifiziert wird.

} DEMUX



Was spricht gegen TCP?

Bessere, in der Anwendungsschicht angesiedelte Kontrolle über gesendete Daten.

- Sobald ein Anwendungsprozess Daten an UDP weiterreicht, verpackt dieses die Daten in einem UDP-Segment und reicht das Segment sofort an die Netzwerkschicht weiter.
⇒ Die Anwendungsschicht behält die Kontrolle.
- Im Vergleich: TCP übernimmt die Kontrolle – die Anwendungsschicht kann nicht eingreifen:
 - Überlastkontrollmechanismus: drosselt
 - Zuverlässig: TCP sendet ein Segment auch immer wieder, bis dessen Empfang am Zielort bestätigt worden ist, egal wie lange die zuverlässige Übertragung dauert.
⇒ Echtzeitanwendungen fordern eine minimale Übertragungsrate: Segmente dürfen nicht übermäßig verzögert sein. Ein geringer Datenverlust wird verkraftet.
⇒ Dienstmodell von TCP passt nicht zu den Anforderungen von Echtzeitanwendungen
- Typisches Vorgehen von Anwendungsentwicklern:
 - wenn TCP ausscheidet: dann UDP verwenden.
 - Reicht der Umfang von UDP als einfacher Segmentzustelldienst nicht aus: dann Implementierung von zusätzlicher Funktionalität als Teil der Anwendung.



Charakterisierung UDP

1. Kein Verbindungsaufbau

1. UDP beginnt ohne formale Vorbereitungen mit der Datenübertragung
Vorteil: keine Verzögerung beim Herstellen einer Verbindung
2. Vgl: TCP baut über Three-Way-Handshake eine Verbindung auf, erst danach Datenübertragung
3. Bsp:
 1. DNS benutzt UDP => schnell.
 2. HTTP verwendet TCP => Zuverlässigkeit ist wichtig für Webseiten
 3. HTTP verwendet TCP => Verzögerung beim Download von Webseiten basiert auf dem der beim Verbindungsauf entstehenden Verzögerung

2. Kein Verbindungszustand

1. UDP speichert keinen Verbindungsstatus (keinen Parameter)
2. Vgl: TCP merkt sich den Verbindungszustand in den Endsystemen.
(Empfangs- und Sendepuffer, Überlastkontrollparameter und Acknowledgment-Nummer)
⇒ dies kostet Ressourcen
3. Konsequenz:
Daher kann ein einzelner Server normalerweise viel mehr aktive Clients unterstützen,
wenn die Anwendung über UDP statt TCP läuft.

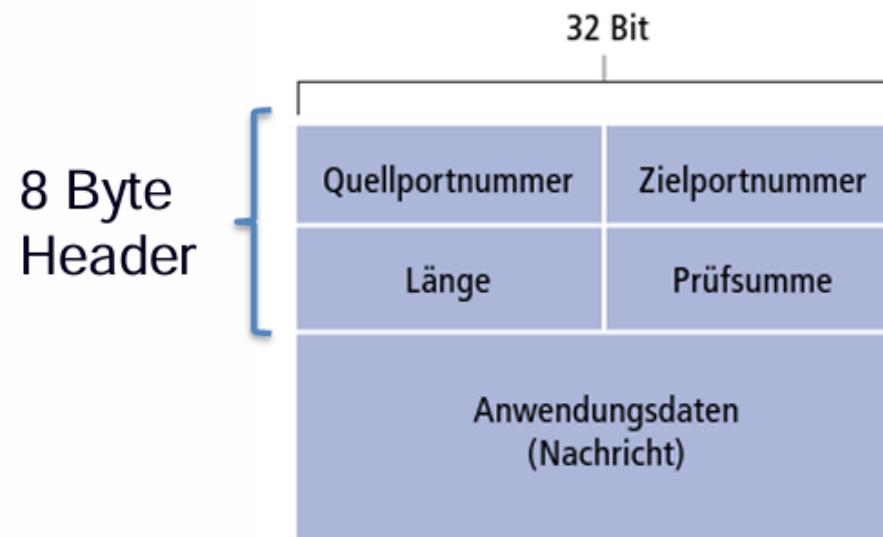
3. Geringe Header-Größe

1. UDP Header mit nur 8 Byte.
2. Vgl: TCP fügt 20 Byte zusätzliche Header-Information zu jedem Segment hinzu



UDP-Segmentstruktur

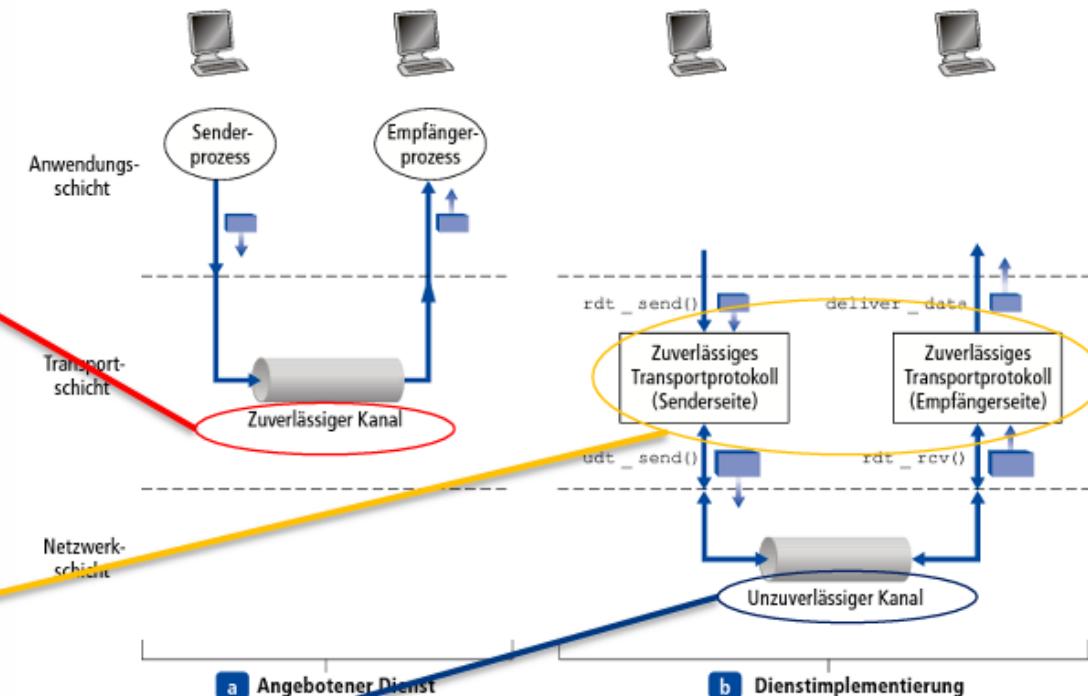
- vier Felder, jeweils zwei Byte (16 Bit)
- Quell- und Zielportnummern:
MUX/ DEMUX
- Prüfsumme (Checksum):
Für jedes Segment wird die Prüfsumme neu vom Sender berechnet.
Der empfangende Host kann damit Fehler im Segment finden.
- Längenfeld:
enthält die Länge des UDP-Segmenteinschließlich des Headers in Byte.
- Anwendungsdaten:
Enthält die zu übermittelnde Nachricht aus der Anwendungsschicht.



4) Grundlagen des zuverlässigen Datentransfers

Zuverlässiger Kanal:

- Ein Dienst der Transportschicht
- Wird der Anwendungsschicht zur Verfügung gestellt.
(Sockets)
- entspricht einem zuverlässigen Kanal zur Übertragung von Daten:
 - Auf einem zuverlässigen Kanal werden die übertragenen Bits nicht verändert (sie können also nicht von 0 auf 1 bzw. von 1 auf 0 springen)
 - Bits können nicht verloren gehen und werden alle in der Reihenfolge zugestellt, in der sie abgesandt wurden.
- ⇒ Dienstmodell das TCP den aufrufenden Anwendungen bietet.



Zuverlässiges Datentransferprotokolls (TCP):

- Liefert den Dienst „zuverlässiger Kanal“
- Herausforderung: Schicht unterhalb des zuverlässigen Datentransferprotokolls: i.d.R. unzuverlässig



5) Verbindungsorientierter Transport: TCP

- zugrunde liegende Prinzipien des zuverlässigen Datentransfers sind Basis von TCP:
Fehlererkennung, Übertragungswiederholungen, kumulative Bestätigungen, Timer und Header-Felder für Sequenznummern und Acknowledgment-Nummern: alles Bestandteile von TCP.
- TCP ist **verbindungsorientiert**, da zwei Prozesse zunächst einen „Handshake“ durchführen
- Erst nach dem 3-Way-Handshake können Anwendungsprozesse Daten austauschen.
- Im 3-Way-Handshake werden Parameter des dann folgenden Datentransfers ausgehandelt.

Unterschied zu leitungsvermittelten Netz: (Netzsicht)

- TCP arbeitet nur auf der Ebene der Transportschicht
- Daher stellt TCP keine durchgehende (TDM-, FDM-...) Leitung wie in einem leitungsvermittelten Netzwerk
- TCP stellte eine **VIRTUELLE Leitung** zur Verfügung:
 - der Zustand der Verbindung wird ausschließlich in den beiden Endsystemen gehalten.
 - Das Innere des Netzes hat keine Kenntnis der virtuellen Leitung,
und hält keinen TCP-Verbindungsstatus.
 - Die Router im Netz sind sich der Transportsicht nicht bewußt, sie sehen insbesondere keine Verbindungen;
und vermitteln (routen) ausschließlich Datagramme (IP-Pakete) durchs Netz.

Eine TCP-Verbindung bietet einen **Vollduplexdienst**:

Besteht eine TCP-Verbindung

- zwischen Prozess A auf einem Host
 - und Prozess B auf einem anderen Host
- dann können Anwendungsschichtdaten:
- sowohl von Prozess A zu Prozess B
 - als auch von Prozess B zu Prozess A fließen.

Eine TCP-Verbindung ist außerdem immer eine **Punkt-zu-Punkt-Verbindung**,

besteht also zwischen

- einem einzelnen Sender
- und einem einzelnen Empfänger.

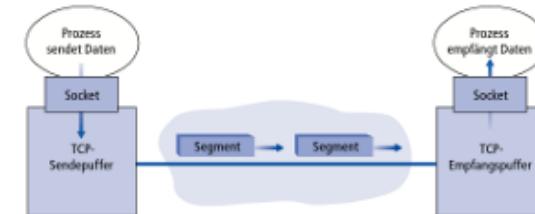
Abgrenzung: **Multicasting**:

- Datentransfer von einem Sender zu vielen Empfängern
 - in einer einzelnen Sendeoperation
- ⇒ ist mit TCP nicht möglich. Bei TCP sind drei Hosts einer zu viel!



Bsp: Senden von Daten vom Client-Prozess an den Server-Prozess

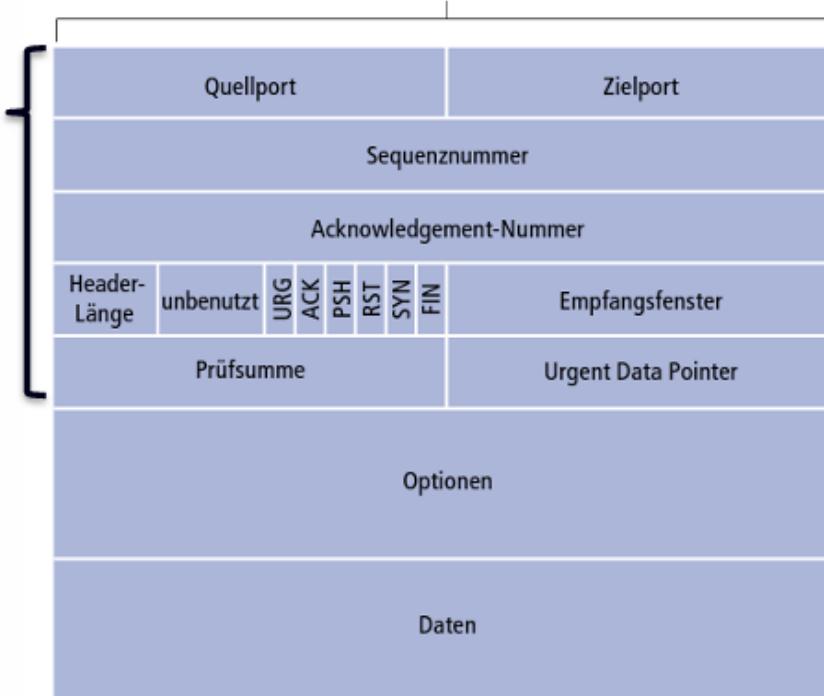
- Der Client-Prozess überträgt einen Datenstrom durch den Socket (die Tür des Prozesses)
- Sobald die Daten durch diese Tür gehen, sind sie in Händen des auf dem Client laufenden TCP.
- TCP packt diese Daten in den Sendepuffer der Verbindung:
einen der Puffer, die während des anfänglichen Drei-Wege-Handshakes reserviert wurden.
- Von Zeit zu Zeit holt TCP Teile der Daten aus dem Sendepuffer – und versendet diese.
 - Zeitpunkt:
Die TCP-Spezifikation [RFC 793] ist zurückhaltend hinsichtlich der Zeitpunkte, zu denen TCP gepufferte Daten senden sollte und legt nur fest, dass TCP „diese Daten in Segmenten nach eigenem Gutdünken senden“ sollte.
 - MSS (maximum segment size) :
Maximale Anwendungsschicht-Datenmenge die auf einmal in ein Segment gepackt werden darf.
(ohne Header!)
 - MTU (maximum transmission unit)
Länge des größtmöglichen Rahmens der Sicherungsschicht, der vom lokalen sendenden Host ausgesandt werden kann, bestimmt wird.
(inkl. aller Header)
 - Typisch: MTU wird bestimmt, dann MSS gesetzt
- TCP ergänzt jeden Block von Client-Daten um einen TCP-Header und bildet dadurch TCP-Segmente.
- Die Segmente werden zur Netzsicht hinuntergereicht, in der sie dann in Netzsicht-IP-Datagramme (IP-Pakete) verkapselt werden.
- Diese IP-Datagramme werden dann ins Netz gesandt.
- Sobald TCP am anderen Ende der Verbindung ein Segment erhält, werden die Daten des Segmentes in den Eingangspuffer der zugehörigen TCP-Verbindung eingefügt.
- Die Anwendung liest den Datenstrom aus diesem Puffer.
- Jede Seite der Verbindung besitzt ihren eigenen Sende und ihren eigenen Empfangspuffer.



TCP-Segmentstruktur

- **Quell- und Zielportnummern** (wie UDP): jeweils 16 bit MUX/DEMUX
- **Sequenznummernfeld** (sequence number field) (32 bit)
- **Acknowledgement-Nummern-Feld** (acknowledgment number field) (32 bit)
- **Header-Längenfeld** (header length field) (4 bit)
Länge des TCP-Headers in 32-Bit-Worten.
Der TCP-Header kann aufgrund des TCP-Options-Feldes unterschiedlich lang sein. (Meistens ist das Optionsfeld leer – dh. 20 Byte Länge)
- **Flag-Feld** (flag field) (6 bit)
 - URG-Bit: Wird in der Praxis nicht angewandt.
 - **ACK-Bit** gibt an, dass der im Acknowledgment-Feld eingetragene Wert gültig ist, d.h., das Segment enthält eine Bestätigung für ein Segment, das erfolgreich empfangen worden ist.
 - PSH-Bit: Wird in der Praxis nicht angewandt.
 - Die **RST-, SYN- und FIN-Bits** werden für den Auf- und Abbau der Verbindung benutzt, wie wir am Ende dieses Abschnittes noch sehen werden.
- **Empfangsfenster-Feld** (receive window field) (16 bit)
wird für die Flusskontrolle verwendet:
Teilt dem Sender die Anzahl von Bytes mit, die der Empfänger bereit ist zu akzeptieren.

20 Byte



- **Prüfsummenfeld** (wie UDP) (16 bit)
- Urgent Data Pointer: (16 bit)
Wird in der Praxis nicht angewandt.
- Länge des **Optionsfelds** (options field) ist variabel.
Dieses Feld wird benötigt für Spezialfälle:
wenn z.B. Sender und Empfänger die zu verwendende maximale Segmentgröße (MSS) aushandeln.
Meisst ist die Länge des Optionsfelds 0.
- **Daten:** Anwendungsschichtdaten



TCP-Sequenznummern und Acknowledgement-Nummern

Zwei der wichtigsten Felder im TCP-Segment-Header:

- Sequenz-Nummer
- Acknowledgement-Nummer.

Diese Felder sind ein kritischer Teil des zuverlässigen Datentransferdienstes von TCP.

Sequenz-Nummer:

- TCP betrachtet Daten als einen unstrukturierten, aber geordneten Strom von Bytes.
- Sequenznummern nummerieren den Strom der gesendeten Bytes und nicht die Folge der gesendeten Segmente:
 - Sequenznummer eines Segmentes ist deshalb die Position des ersten Bytes des Segmentes im Bytestrom.

Bsp: Annahmen:

- Prozess in Host A sendet Daten an Prozess in Host B über TCP
- TCP in Host A nummeriert implizit **jedes** Byte im Datenstrom.
- Datenstrom: eine Datei der Länge 500.000 Byte
- MSS: 1.000 Byte
- erstes Byte des Datenstroms die Sequenz-Nummer 0



Acknowledgment-Nummern.

Bsp 1: Daten: B => A ACK: B <= A

- Jedes der Segmente, die von Host B ankommen, hat eine Sequenznummer für die von B zu A fließenden Daten.
- Acknowledgment-Nummer, die Host A für sein Segment verwendet, ist die **Sequenznummer des nächsten Bytes**, das Host A von Host B erwartet.

Bsp 2:

- Annahme: A hat alle Bytes von 0 bis 535 von B erhalten.
- A sendet die SN 536 im Acknowledgment-Nummernfeld seines Segments an B.

Bsp 3:

- A erhält von B: Segment: Bytes 0 bis 535
- A erhält von B: Segment: Bytes 900 bis 1000
- A hat Bytes von 536 bis 899 noch nicht erhalten.
- A sendet die SN 536 im Acknowledgment-Nummernfeld seines Segments an B.

TCP kumulative Acknowledgments (cumulative acknowledgments):

TCP bestätigt nur Bytes bis zum ersten fehlenden Byte im Datenstrom

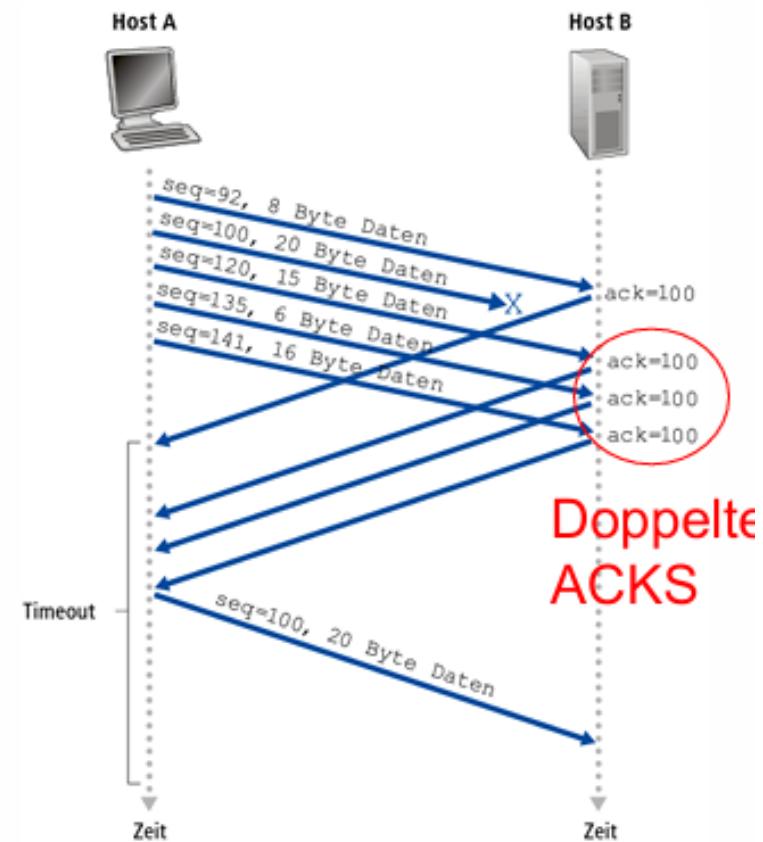
Des Weiteren: Reihenfolge: (Bsp 3): 3. Segment vor dem 2. Segment



Zuverlässiger Datentransfer: Fast Retransmit

Problem: Timeout-Periode kann relativ lang sein.

- Geht ein Segment verloren, zwingt eine Timeout-Periode den Absender dazu, die erneute Übertragung des Segmentes lange zu verzögern.
- Sender kann Paketverluste lange vor dem Eintreten des Timeout-Ereignisses erkennen:
Doppelte ACKS sind ein Hinweis!
- Fast Retransmit:
 - Falls drei doppelte ACKs empfangen werden,
 - der Timer noch nicht abgelaufen ist,
 - führt der TCP-Sender eine schnelle Übertragungswiederholung (fast retransmit) durch:
Das fehlende Segment wird nochmals übertragen.



Erzeugung von TCP ACK gem. [RFC 1122; RFC 2581]

Ereignis	Aktion des TCP-Empfängers	
Ankunft des Segmentes in der richtigen Reihenfolge mit der erwarteten Sequenznummer. Alle Daten bis zur erwarteten Sequenznummer sind bereits bestätigt.	Verzögertes ACK. Wartet bis zu 500 ms auf die Ankunft eines anderen Segmentes in richtiger Reihenfolge. Wenn das nächste Segment nicht in diesem Zeitintervall eintrifft, wird ein ACK gesendet.	http://www-01.ibm.com/support/knowledgecenter/SSQ2BZ_9.0.0/com.ibm.gdata.hc4nconfig.reference.doc/topics/delayedack.html?lang=de
Ankunft eines Segmentes in der richtigen Reihenfolge mit erwarteter Sequenznummer. Ein anderes Segment in der korrekten Reihenfolge wartet auf die ACK-Übertragung.	Sendet sofort ein einzelnes kumulatives ACK, bestätigt beide in richtiger Reihenfolge eingetroffene Segmente.	
Ankunft eines Segmentes außerhalb der Reihenfolge mit einer Sequenznummer, die größer ist als erwartet. Lücke im Bytestrom aufgetreten.	Sendet sofort ein doppeltes ACK, in dem er die Sequenznummer des nächsten erwarteten Bytes angibt.	
Ankunft eines Segmentes, das die Lücke in den erhaltenen Daten ganz oder teilweise ausfüllt.	Sendet sofort ein ACK, vorausgesetzt, das Segment beginnt mit der Sequenznummer des nächsten erwarteten Bytes. Bestätigt alle nun lückenlos vorliegenden Bytes.	



Additive-Increase, Multiplicative-Decrease: Sägezahn

→ Zusammenfassung

- ein TCP-Sender steigert sein Tempo additiv, wenn er erkennt, dass der Ende-zu-Ende-Pfad frei von Überlast ist.
- ein TCP-Sender verringert sein Tempo multiplikativ, wenn er (über ein Verlustereignis) wahrnimmt, dass der Pfad überlastet ist.
- Deshalb wird die TCP-Überlastkontrolle als
 - Additive-Increase-
 - Multiplicative-Decrease-Algorithmus (AIMD) bezeichnet.
- Konsequenz: Sägezahn
Der Wert von CongWin durchläuft immer wieder Zyklen:
 - in denen er linear zunimmt,
 - um dann plötzlich auf die Hälfte seines aktuellen Werts zu fallen (wenn ein Verlustereignis stattfindet)Dies führt in langlebigen TCP-Verbindungen zu einem Sägezahnmuster.

Typischer Verlauf eines TCP-Congestion-Window über die Zeit:



Aufgabe der Netzsicht:

Pakete von einem sendenden Host zu einem Empfängerhost (durch das Internet) bewegen

Zwei Funktionen in der Netzsicht:

1. Weiterleitung (Forwarding):

- Ein IP-Paket kommt auf einer Eingangsleitung eines Routers an.
- Der Router leitet das IP-Paket über die „richtige“ Ausgangsleitung an den nächsten Router (bzw. Host) weiter.
- Die IP-Pakete werden von Quellhost zu Zielhost damit schrittweise über einen Pfad weitergeleitet.
⇒ Weiterleitung bezieht sich nur auf lokale Aktionen des Routers

2. Routing:

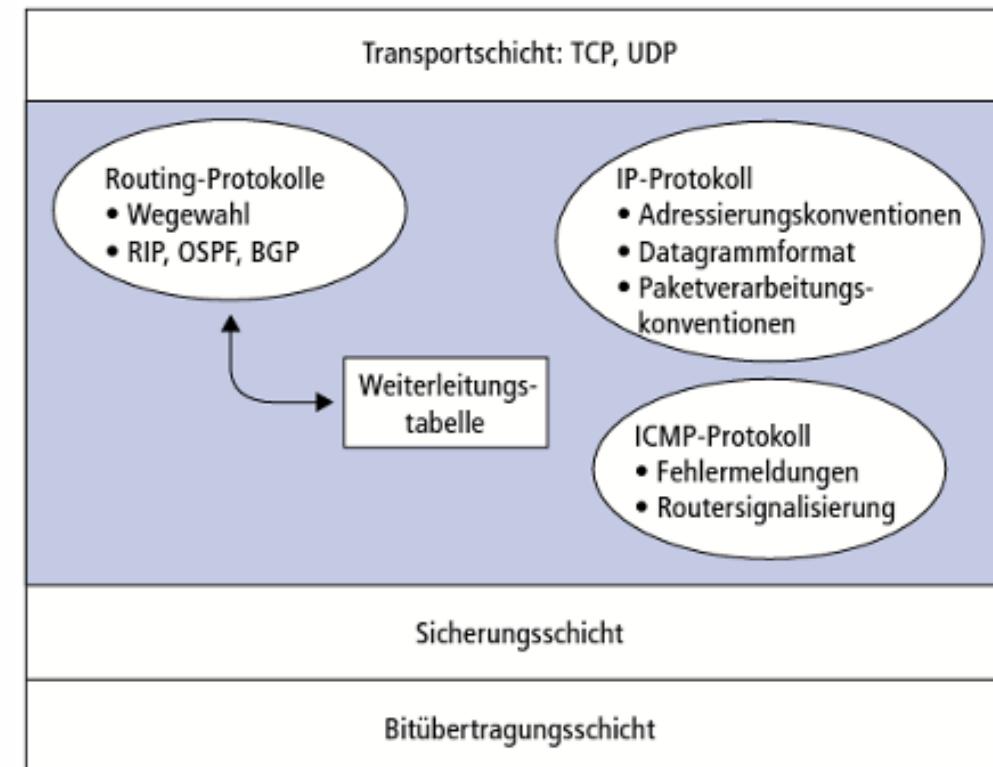
- Bestimmung des Pfads von Quellhost zu Zielhost innerhalb des Internet, der von IP-Paketen durchlaufen wird.
- Routing-Algorithmen: Berechnung der Pfade.
⇒ Routing ist ein netzweiter Prozess



Komponenten der Netzschicht

Drei Hauptbestandteile:

- **Internetprotokoll (IP-Protokoll)**
https://de.wikipedia.org/wiki/Internet_Protocol
- **Routing-Protokolle des Internet**
https://de.wikipedia.org/wiki/Routing#Routing_im_Internet
 - Bestimmen den Pfad eines Datagramms von der Quelle zum Ziel bestimmt:
Berechnung der Weiterleitungstabellen
- **Internet Control Message Protocol (ICMP)**
https://de.wikipedia.org/wiki/Internet_Control_Message_Protocol
<https://www.youtube.com/watch?v=M78kvjyMrA0>
 - Sendet Rückmeldung über aufgetretene Fehler zurück
 - Informationsaustausch über bestimmte Informationen der Netzwerkschicht.



Zusammenspiel: Weiterleitung und Routing

- Jeder Router hat eine Weiterleitungstabelle (forwarding table).
- Weiterleitung:
 - Wert eines Feldes in der Header-Zeile des ankommenden IP-Paketes
 - Wert in der Weiterleitungstabelle des Routers
 - Bestimmen die Ausgangsleitung

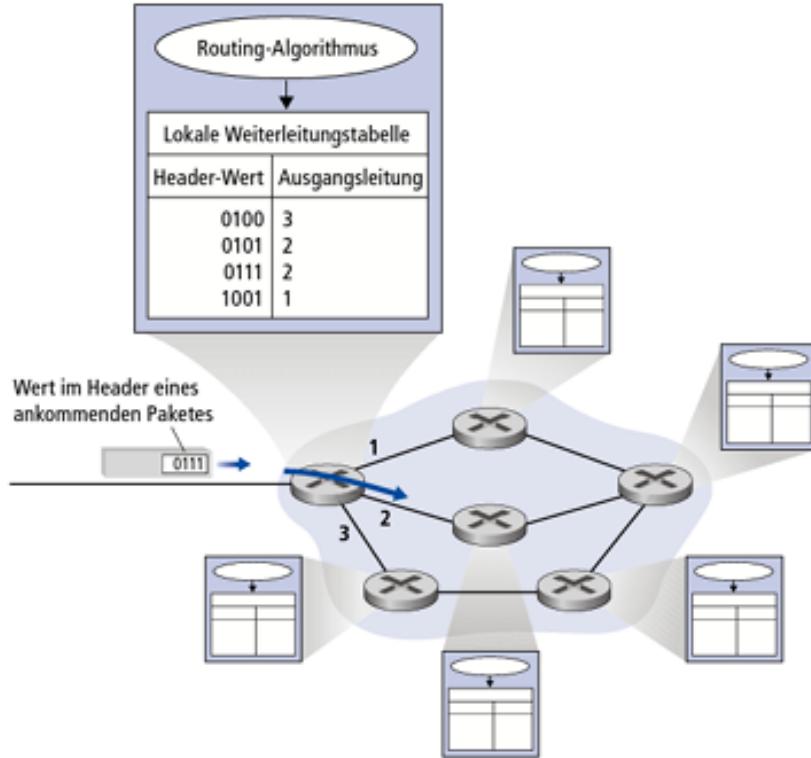
- Frage:

Wie wird die Weiterleitungstabelle im Router festgelegt ?

⇒ Routing

- Routing:

Der Routing-Algorithmus bestimmt den Eintrag der jeweiligen Ausgangsleitung in den Weiterleitungstabellen.

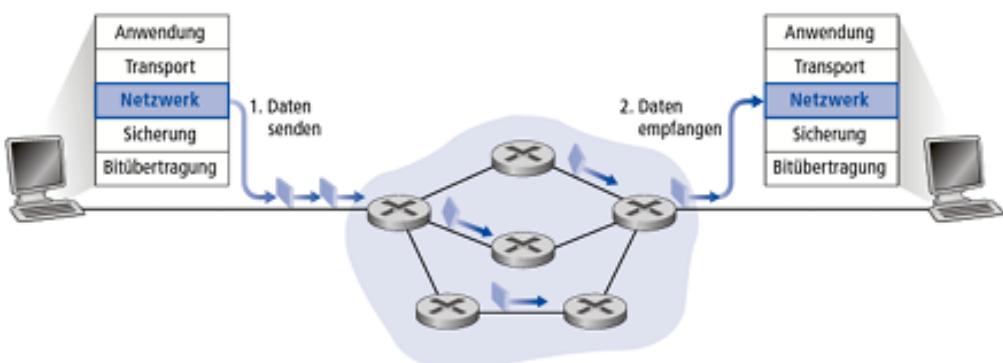


Datagramm-Netze – so ist das Internet aufgebaut

Prinzip-Endsystem:

Jedes Mal, wenn ein Endsystem in einem Datagramm-Netz ein Paket versenden will:

- markiert es das Paket mit der **Adresse des Zielendsystems**
- und schiebt das Paket dann ins Netz
- Ohne Aufbau einer virtuellen Leitung



Prinzip-Router:

- Router in einem Datagrammnetz verwalten keinerlei Statusinformationen über virtuelle Leitungen.
(es gibt keine virtuellen Leitungen)
- Auf dem Weg von der Quelle zum Ziel durchquert ein Paket eine Reihe von Routern.
- Jeder dieser Router verwendet **die Zieladresse des Paketes**, um es weiterzuleiten.
- Jeder Router hat eine **Weiterleitungstabelle**, die Zieladressen auf Ausgänge abbildet.
- Kommt ein Paket am Router an, verwendet der Router die Zieladresse des Paketes, um die entsprechende Ausgangsleitung zu identifizieren
- Der Router leitet dann das Paket an die identifizierte Ausgangsleitung weiter.



Weitere exemplarische Analyse: Weiterleitungstabellen (I)

Zieladress-Bereich (dezimale Darstellung)	Ausgangsleitung
200.23.16.0 – 200.23.23.255	0
200.23.24.0 – 200.23.24.255	1
200.23.25.0 – 200.23.31.255	2
Sonst	3



Umwandlung binäre Darstellung: Beide Tabellen identisch!

Zieladress-Bereich (binäre Darstellung)	Ausgangsleitung
11001000.00010111.00010000.00000000 – 11001000.00010111.00010111.11111111	0
11001000.00010111.00011000.00000000 – 11001000.00010111.00011000.11111111	1
11001000.00010111.00011001.00000000 – 11001000.00010111.00011111.11111111	2
Sonst	3



Longest Prefix Match – Weiterleitungstabelle im Router

Prefix (Zieladress-Bereich)	Ausgangsleitung
11001000.00010111.00010	0
11001000.00010111.00011000.	1
11001000.00010111.00011	2
Sonst	3

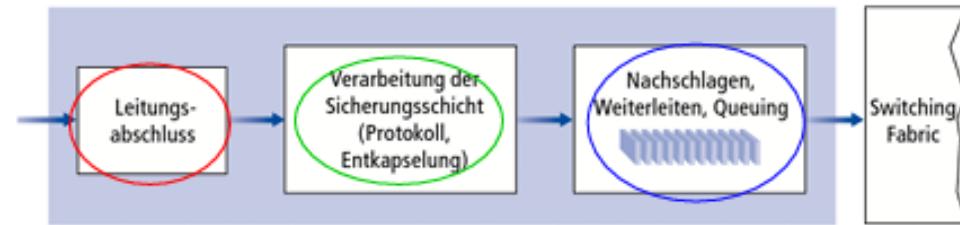
Weiterleitungsprinzip:

- Datenpaket: Zieladresse in binärer Form
 - Bitweiser-Vergleich der Zieladresse mit den Präfixes
 - Ist ein Präfix vollständig in der Zieladresse enthalten: MATCH !
 - Matchen mehrere Präfixe: der längste Präfix gewinnt.
 - Matched kein Präfix: Weiterleitung bei „Sonst“
- ⇒ longest prefix match



Eingangsports

- Endpunkt der eingehenden physikalischen Leitung die Bitübertragungs- und Sicherungsschicht.
- **Netzschicht:** Wesentlich: Such-/Weiterleitungsmodul:
 - Weiterleitungsfunktion
Bestimmung des Ausgangsport, auf den das eintreffende Paket weitergeleitet werden soll.
 - Bestimmung des Ausgangsports erfolgt mithilfe der **Weiterleitungstabelle**



Ablauf der Bestimmung des Ausgangsport:

1. Neues IP-Paket empfangen
(Bitübertragung-, Sicherungsschicht leiten an Netzschicht weiter)
2. In der Netzschicht: Suche in der Weiterleitungstabelle:
Die Ziel-IP-Adresse des IP-Pakets:
Longest Prefix Match
⇒ Ausgangsport ist bestimmt.

Einfache Abschätzung der Anforderungen am Beispiel:

- Empfang von Datenpaketen mit ca. 8 GBit/s
(dh. Empfang von ca. 1 GByte/s)
- Annahme MSS ca. 1000 Byte
- dh.ca. eine Million IP-Pakete kommen pro Sekunde an!
⇒ Anforderung an Suchalgorithmus:
Bestimmung Ausgangsport: muss schneller als 1/1.000.000 Sekunden sein! (dh. eine Mikrosekunde)

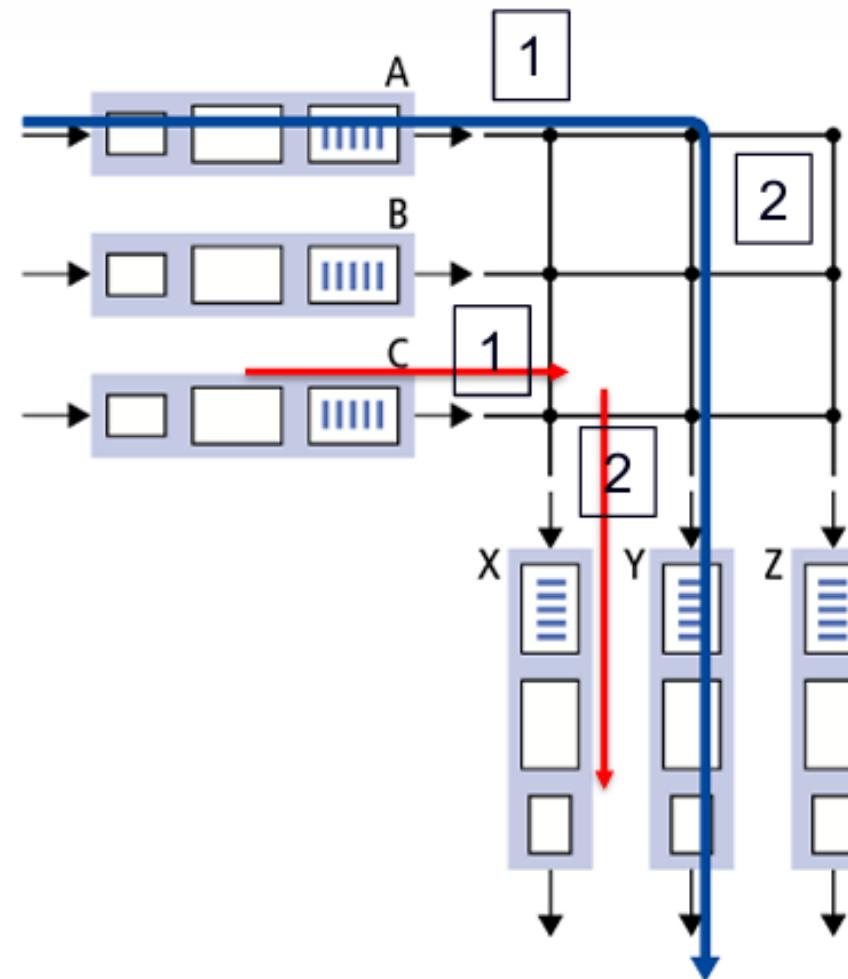
Weiterleitung des IP-Pakets an die Switching Fabric:

- Sobald der Ausgangsport bestimmt ist
- Ist die Switching Fabric gerade anderweitig beschäftigt:
Speichern des Datenpakets (Queuing)



Switching Fabric (Beispiel mittels Verbindungsnetz)

- „Herzstück eines Routers“
- Datenpakete werden von einem Eingangsport auf einen Ausgangsport weitergeleitet.
- Diese Weiterleitung nennt man Switching.
- Switching mittels Verbindungsnetz:
Parallelität möglich!
 1. Ein Paket, das an einem Eingangsport ankommt, bewegt sich über den am Eingangsport angeschlossenen **horizontalen Bus**, bis es auf die **Kreuzung** stößt, die zum gewünschten Ausgangsport führt.
 2. Wenn der dorthin führende **vertikale Bus frei ist**, wird das Paket zum Ausgangsport übertragen.
 3. Transportiert der vertikale Bus bereits ein Paket eines anderen Eingangsports zu diesem Ausgangsport: ist das ankommende Paket blockiert und es muss am Eingangsport in eine Warteschlange eingereiht werden.
- Engl: Verbindungsnetz == Crossbar

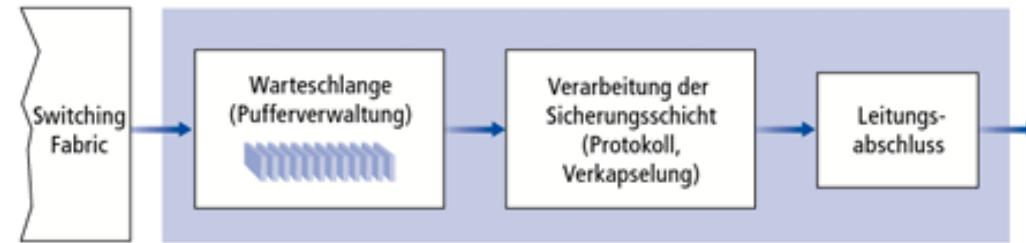


Ausgangsports

- IP-Pakete werden im Speicher des Ausgangsports abgelegt (Warteschlange)
- IP-Pakete werden aus dem Speicher aufgenommen, und über die ausgehende Leitung versandt.

Warteschlangen- und die Pufferverwaltungsfunktionalität:

- Wird benötigt, wenn aus der Switching-Fabric mehr Pakete pro Zeit angeliefert werden, als der Ausgangsport pro Zeit versenden kann.
- kann mit kurzfristigen peaks umgehen



Paketverluste

Paketwarteschlangen können sich bilden an:

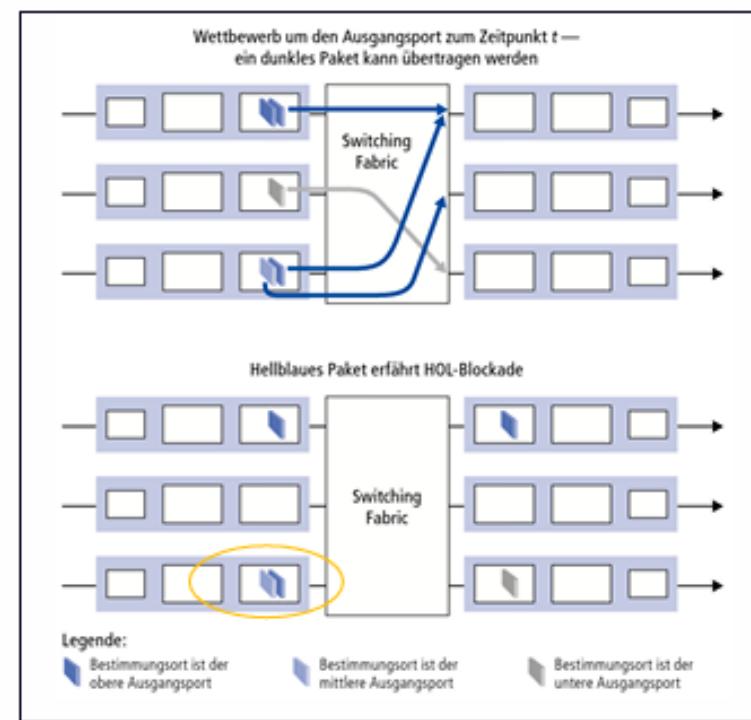
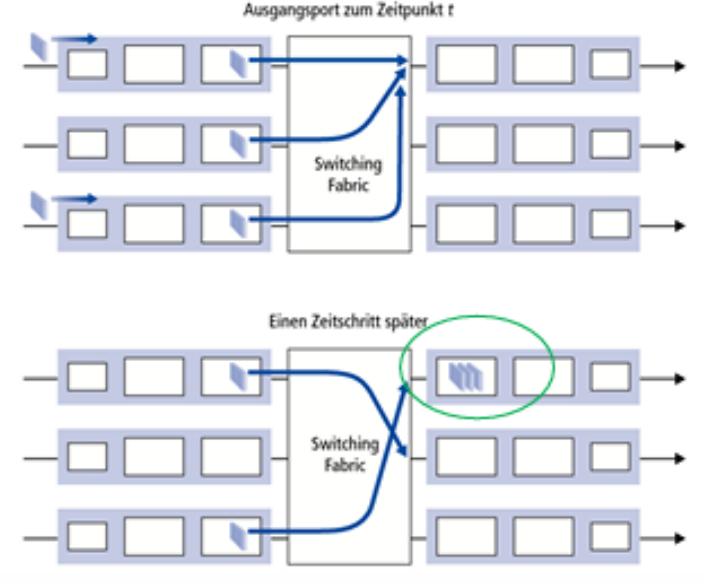
- Eingangsports
- Ausgangsports

Diese Warteschlagen können nur bis zu einem bestimmten Punkt anwachsen, dann ist der Pufferspeicher des Routers erschöpft: und **es tritt ein Paketverlust auf.**

In den Warteschlagen innerhalb eines Routers kann es zu Paketverlusten kommen.

Der genaue Ort des Paketverlustes (Eingang oder Ausgang) abhängig vom

- Verkehrsaufkommen
- Geschwindigkeit des Switching Fabric
- Leitungsgeschwindigkeit(en)



IPv4 Adressierung

Allgemeines:

Endsystem/ Host:

- typischerweise über eine einzige physikalische Leitung am Netz angeschlossen.
- Schnittstelle (Interface):
Grenze zwischen Host und physikalischen Leitung.

Router:

- ist an zwei oder mehr physikalischen Leitungen angeschlossen.
- Schnittstelle:
Grenze zwischen dem Router und jeder angeschlossenen Leitung
⇒ mehrere Schnittstellen, eine für jede Leitung.

Prinzip von IP:

- Jede Host- und Router-Schnittstelle hat eine eigene IP-Adresse
 - ⇒ IP-Adresse ist einer Schnittstelle zugeordnet (und nicht dem Host oder Router, zu dem diese Schnittstelle gehört.)
- Jede IP-Adresse ist 32 Bit lang (4 Byte).
 - ⇒ Es gibt: 2^{32} mögliche IP-Adressen (etwas über 4 Milliarden)
- IP-Adressen werden im Dezimalformat geschrieben:
 - jedes Byte der Adresse in seiner dezimalen Form
 - Trennung von benachbarten Bytes durch einen Punkt: Dezimal: 193.32.216.9
 - Binär: 11000001 00100000 11011000 00001001
- Jede Schnittstelle eines jeden Hosts und Routers im globalen Internet muss:
 - eine IP-Adresse haben
 - die weltweit eindeutig ist
- IP-Adressen können nicht beliebig gewählt werden:
Ein Teil der IP-Adresse einer Schnittstelle wird vom zugehörigen **Subnet** definiert.



IPv4 Adressierung – Beispiel (I)

Bsp: Router ist über 3 Schnittstellen mit 7 Hosts verbunden.

Auffällig:

- 3 Hosts und Router-Schnittstelle haben alle eine IP-Adresse der Form 223.1.1.xxx.
⇒ die ersten 24 Bit sind identisch
- Diese 4 Schnittstellen sind zudem miteinander über ein Netz verbunden, das keine Router enthält.
(Bsp: Ethernet-LAN)

Definition: Subnet:

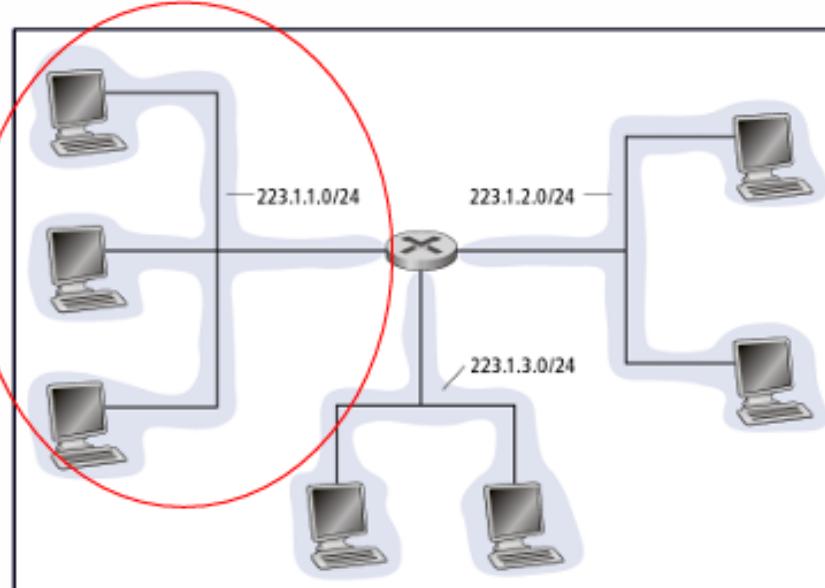
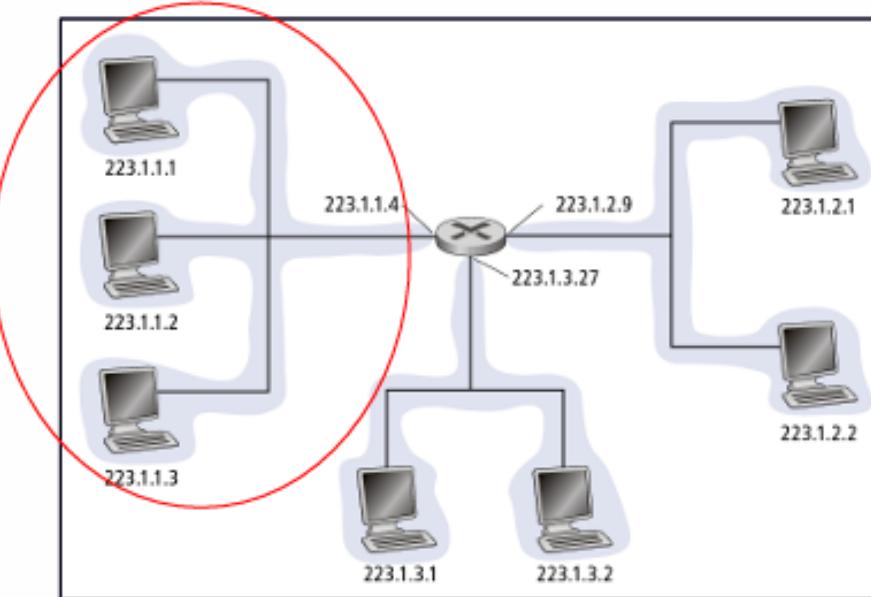
Netz aus Hostschnittstellen und/ oder Routerschnittstelle(n).

Definition: Adressierung eines Subnet am Beispiel:

- Subnetzmaske /24:
die "linken" 24 Bit (von 32 möglichen Bit für eine IP-Adresse) sind in diesem Subnet für alle Interfaces identisch.
⇒ Subnetz bekommt die Adresse: 223.1.1.0/24

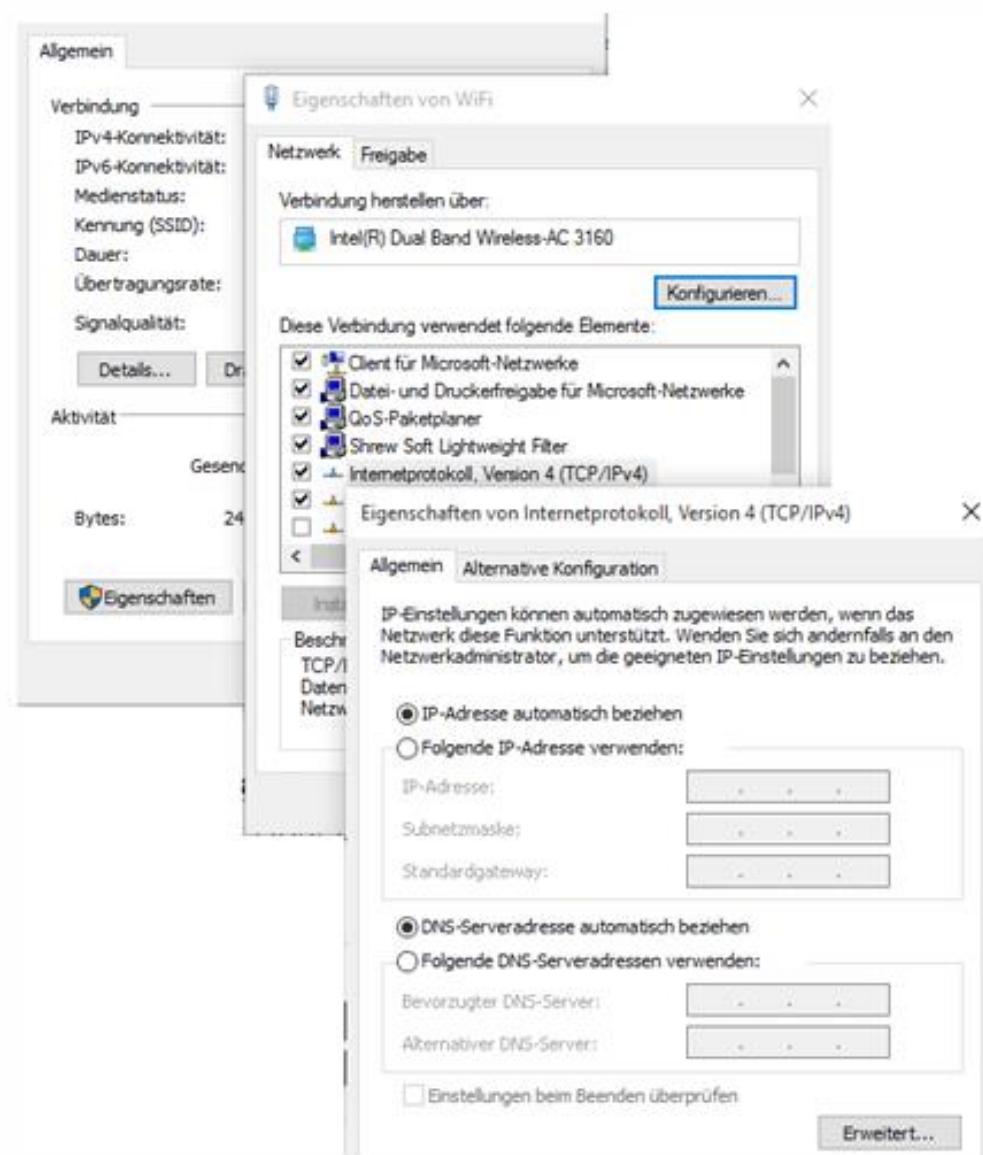
Konkret: Das Subnet 223.1.1.0/24 besteht aus:

- 3 Host-Schnittstellen (223.1.1.1, 223.1.1.2 und 223.1.1.3)
- 1 Router-Schnittstelle (223.1.1.4).
- Weitere Hosts im Subnet möglich: (223.1.1.XXX)
auch mit eindeutiger IP-Adresse



DHCP (Dynamic Host Configuration Protocol)

- DHCP ermöglicht es einem Host, eine IP-Adresse automatisch zu beziehen (zugeordnet zu bekommen)
- Zusätzlich liefert DHCP auch weitere Information:
 - Subnetzmaske
 - Adresse seines First-Hop-Routers (Standardgateway)
 - Adresse des lokalen DNS-Servers.
- 2 Konfigurations-Möglichkeiten:
 - ein bestimmter Host erhält jedes Mal, wenn er sich mit dem Netz verbündet, dieselbe IP-Adresse
 - einem Host wird eine temporäre IP-Adresse zugewiesen, diese ändert sich bei jedem Zugang zum Netz.
- DHCP wird typischerweise im Gateway konfiguriert (z.B. in der Fritzbox)



Network Address Translation (NAT) – am Beispiel

Es gibt 3 Adress-Bereiche die nicht global geroutet werden:

Private Netze:

1. 10.0.0.0 bis 10.255.255.255

10.0.0.0/8

Klasse A: 1 privates Netz mit 16.777.216 Adressen;

2. 172.16.0.0 bis 172.31.255.255

172.16.0.0/12

Klasse B: 16 private Netze mit jeweils 65.536 Adressen;

3. 192.168.0.0 bis 192.168.255.255

192.168.0.0/16

Klasse C: 256 private Netze mit jeweils 256 Adressen;

Typischer Aufbau im Haushalt:

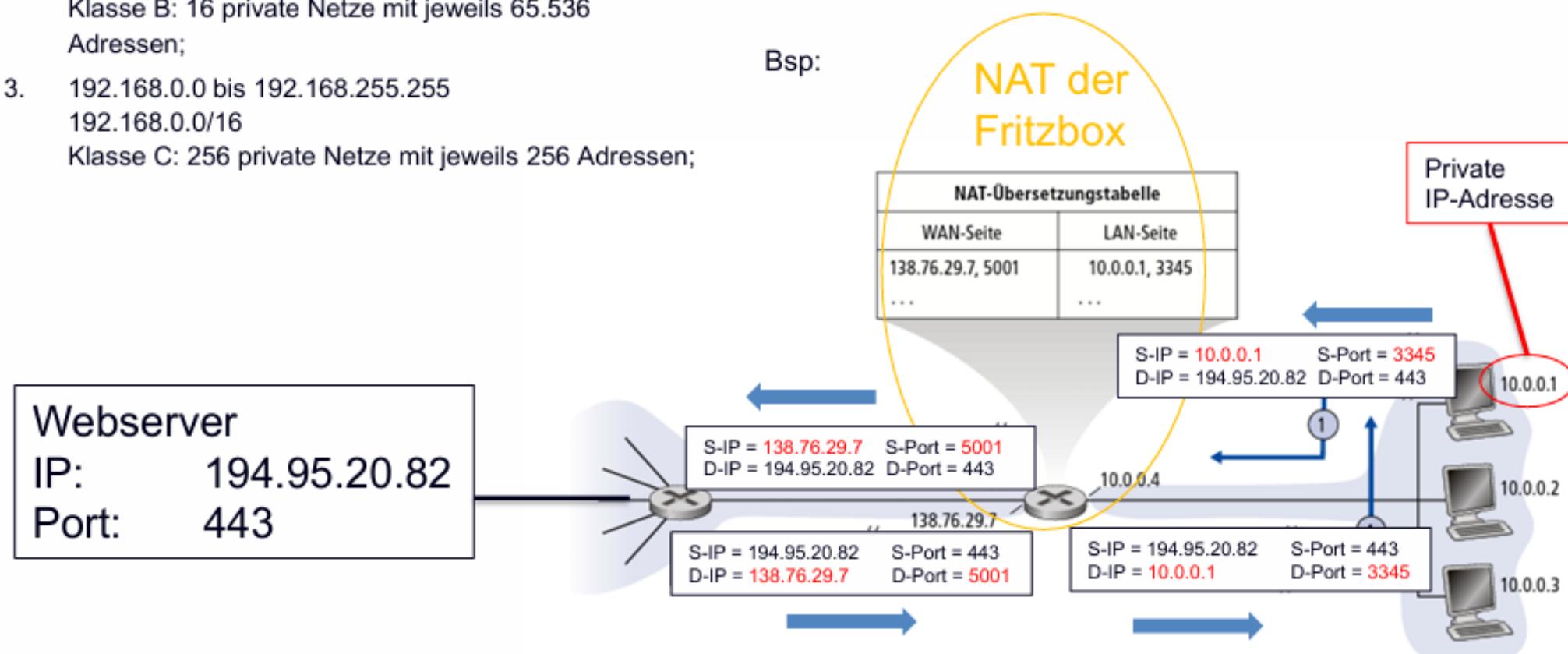
- DSL-Verbindung

- Fritzbox (mit NAT)

- Sämtliche Geräte im Haushalt:

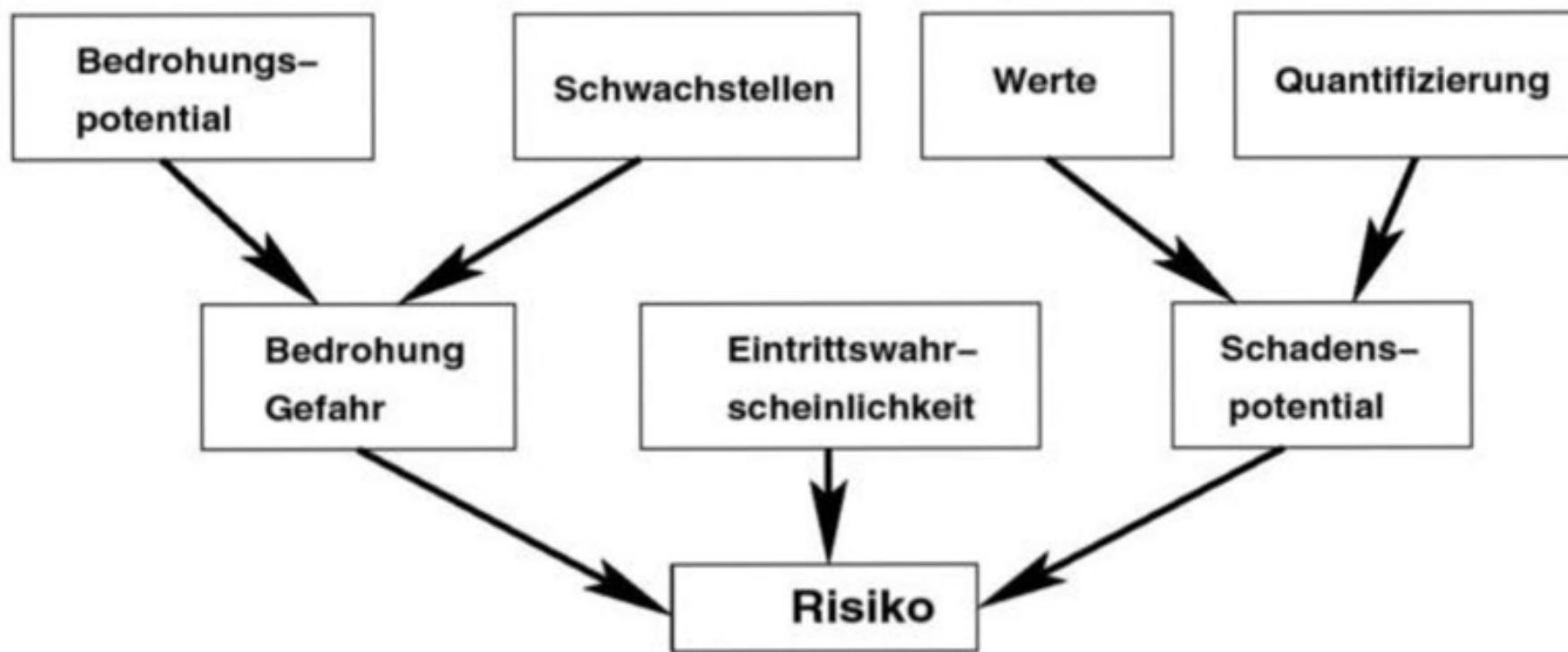
Haben eine private IP-Adresse
(diese ist aus dem Internet nicht zu erreichen)

Bsp:



Zusammenhang: Schwachstellen, Bedrohungen, Risiken

Selbststudium



[Eckert]



Angriffs- und Angreifer-Typen (I)

Selbststudium

Angriff (Attack)

Nicht autorisierter Zugriff bzw. Einen nicht autorisierten Zugriffsversuch auf das System.

- **Passive Angriffe:**

Unautorisierte Informationsgewinnung: Ziel Verlust
Vertraulichkeit eines IT-Systems

- **Aktive Angriffe:**

Unautorisierte Modifikation von Datenobjekten: Ziel
Verlust Datenintegrität und/ oder Verfügbarkeit eines
IT-Sytems

Beispiele: Passive Angriffe

- **Sniffer**

<https://de.wikipedia.org/wiki/Sniffer>

Abhören von Datenleitungen in vernetzten
Systemen

- Unautorisiertes Lesen von Daten aus Dateien
- Kritisch: z.B. Ausspähen von Passwörtern

Beispiele: Aktive Angriffe

- **Spoofing**

<https://de.wikipedia.org/wiki/Spoofing>

Täuschungsmethoden in Computernetzwerken
zur Verschleierung der eigenen Identität.

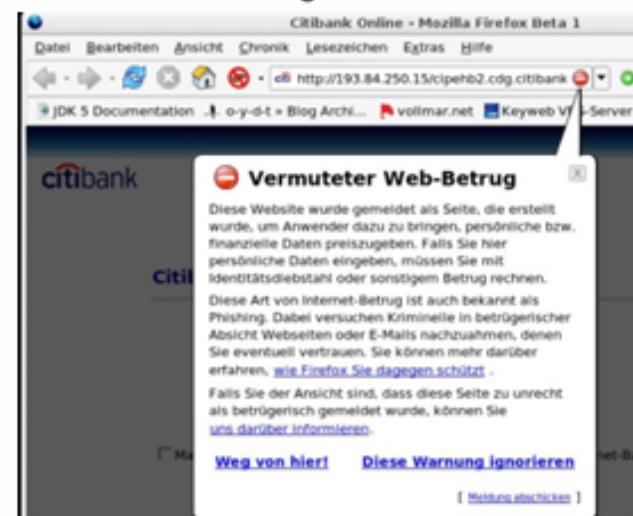
Personen werden in diesem Zusammenhang
auch gelegentlich als „Spoofers“ bezeichnet.

- 1. **Maskierungsangriffe/ Phishing**

<https://de.wikipedia.org/wiki/Phishing>

Identitätsdiebstahl:

Versuche, über gefälschte Webseiten, E-
Mails oder Kurznachrichten an persönliche
Daten eines Internet-Benutzers zu gelangen
und damit zu begehen.



[Eckert]



100 Folien schon? Aber wann hab ich es denn
endlich verstanden und bin fertig?!



That's the neat thing: you don't

Angreifer-Typen (I)

Selbststudium

Angriffs-Ursachen:

- Angriff durch Innentäter (interner Mitarbeiter/ Affiliate / Supplier)
- Angriff durch Externen (steht in keinerlei Zusammenhang zur Firma)

Angriffs-Typen:

1. Hacker

- Vorgehen:
 1. Aufdeckung von Schwachstellen & Verwundbarkeiten in IT-Systemen („Forschung“)
 2. Exploits (Angriffe) dafür entwickeln, („angewandte Forschung“)
 3. Angriff: Ausnutzung der Schwachstellen mit Exploits („praktische Umsetzung“)
- Ziel:
 1. idR: Information der Öffentlichkeit über Schwachstellen
 2. idR: kein Interesse durch der Ausnutzung einer Schwachstelle Geld zu verdienen.
(Hacker-Ethik)
- Häufig Illegal:
 1. Wird die praktische Umsetzung nicht z.B. am heimischen PC simuliert, sondern ein Unternehmen angegriffen, dann ist das strafbar.

[Eckert]



2. „Skript-Kiddies“

- Vorgehen:
 1. Technisch nicht so versiert wie Hacker / Cracker
 2. Es liegt kein Business Case zugrunde, sondern eher „Spieltrieb“
- Ziel:
 1. Angriffe sind idR nicht gezielt.
 2. Bsp: Eine Anleitung eines Exploits aus dem Internet herunterladen, und „ausprobieren“ wo dieser Exploit funktionert.
- Häufiger illegal als Hacker – „denn sie wissen nicht was sie tun“.

3. Cracker

- Vorgehen: Analog zu Hacker; nicht organisiert
- Ziel:
 1. Mit krimineller Vorgehensweise Geld verdienen.
 2. Eigener Vorteil, oder Vorteil eines Dritten (der bezahlt)
- Kriminell

[Eckert]



Angreifer-Typen (III)

4. Organisierte mafiose Strukturen:

- Vorgehen:
 1. Industrialisiert – Arbeitsteilung
 2. Global
- Ziel:
 1. Angriffe unterliegen einem Business-Case
 2. Da illegal **muss der Business-Case sehr positiv sein;**
Sonst wird ein Auftrag nicht angenommen.
- Organisiertes Verbrechen

Bsp:

<http://www.independent.co.uk/news/uk/crime/mafia-cybercrime-booming-and-with-it-a-whole-service-industry-says-study-9763447.html>

<http://www.computerworld.com/article/2503653/cybercrime-hacking/russian-cybercriminals-earned-4-5-billion-in-2011.html>

<https://www.europol.europa.eu/iocfa/2014/executivesummary.html>

Europol: The Crime-as-a-Service (CaaS) business model drives the digital underground economy by providing a wide range of commercial services that facilitate almost any type of cybercrime. Criminals are freely able to procure such services, such as the rental of botnets, denial-of-service attacks, malware development, data theft and password cracking, to commit crimes themselves. This has facilitated a move by traditional organised crime groups (OCGs) into cybercrime areas. **The financial gain that cybercrime experts have from offering these services stimulates the commercialisation of cybercrime as well as its innovation and further sophistication.**

A screenshot of a ComputerWorld news article. The headline reads "Russian cybercriminals earned \$4.5 billion in 2011". Below the headline, there is a summary: "Russian mafia took control and professionalized online crime in 2011, researchers say". The article is by Leah Ertman, published on April 26, 2012. It includes a sidebar with related topics like "Cybercrime & Marketing" and "Security". A red diagonal watermark "Selbststudium" is overlaid on the top right of the screenshot.



Mafia cybercrime booming and with it a whole service industry, says study

The Internet Organised Crime Threat Assessment (iOCTA) identifies developments and emerging threats of cybercrime affecting governments and law enforcement authorities in the EU and provides important input to the report.



The industry-wide global revenue is estimated at \$100 billion per year, with

Traditional Mafia groups are increasingly outsourcing their specialty operations to highly skilled freelance cybercriminals who promote their services on hidden websites, says a new report.

The market in online criminal services has boomed with bespoke money laundering schemes and factoring on a payment-by-results basis.



[Eckert]



Angreifer-Typen (IV)

Selbststudium

5. Nachrichtendienste

https://en.wikipedia.org/wiki/List_of_government_mass_surveillance_projects

<http://www.handelsblatt.com/politik/international/neue-snowden-enthuellung-mit-programm-xkeyscore-ueberwachung-in-echtzeit/8577568-all.html>

Handelsblatt

Home Digitalpass Finanzen Unternehmen Politik Technik Auto Sport

https://en.wikipedia.org/wiki/United_States_intelligence_budget

<https://www.washingtonpost.com/world/national-security/black-successes-failures-and-objectives/2013/08/29/7e57bb78-10ab>

Wieder war es die britische Tageszeitung „The Guardian“, die die Snowden-Enthüllungen öffentlich machte. Sie stellte eine NSA-Präsentation ins Netz, nach der Mitarbeiter über ein Programm namens „XKeyscore“ Zugriff auf gewaltige Datenmengen haben. Dieses Programm setzt auch das deutsche Bundesamt für Verfassungsschutz testweise ein. Dem Dokument von 2008 zufolge können Geheimdienstler in den „enormen Datenbanken“ der NSA nach Namen, E-Mail-Adressen, Telefonnummern und Schlagworten suchen. Für die einzelnen Anfragen bräuchten sie keine gesonderte Zustimmung eines Richters oder eines anderen NSA-Mitarbeiters, schreibt der „Guardian“.



NSA-AFFÄRE
Snowdens Vater rät zu Aufenthalt in Russland

Auch die Beobachtung der Internetaktivität einzelner Menschen in Echtzeit sei mit „XKeyscore“ möglich. Unter anderem könne man die IP-Adresse jedes Besuchers einer bestimmten Website erfassen. Inhalte der Kommunikation würden drei bis fünf Tage lang gespeichert, Verbindungsdaten 30 Tage. Innerhalb eines solchen

‘op-line figure of aggregate NIP and available aggregate MIP budget, FY 2006-pr

Fiscal Year	NIP in Billion \$ appropriated	MIP in Billion \$ appropriated
2006	40.9 ^[14]	not disclosed
2007	43.5 ^[15]	not disclosed
2008	47.5 ^[16]	not disclosed
2009	49.8 ^[1]	not disclosed
2010	53.1 ^[2]	27.0 ^[3]
2011	54.6 ^[17]	not disclosed
2012	53.9 ^[18]	not disclosed
2013	52.7 ^[19] 49.0 corrected amount	not disclosed
2014	TBA	?

[Eckert]



Bot Nets – „Industrialisierte“ Plattformen für Cybercriminals

Selbststudium

<https://en.wikipedia.org/wiki/Botnet>

Industrialisierung/ Arbeitsteilung:

Betreiber von Bot Nets **vermieten** „ihre“ Plattform an andere Cybercriminals.

Mögliche Angriffe:

Gegenüber Dritten:

- Proxy
- Spam-Mails/ Phishing
- DDoS
- Klickbetrug (um Online Ads Revenue Stream zu pushen)

Gegenüber dem Host:

- Spionage, Manipulation



Ransomware / Erpressung

Erpresserische Malware

1. Ransomware infiziert z.B. den Laptop des Opfers
2. Verschlüsselt alle Daten auf der lokalen Festplatte des Opfers
3. Das Opfer kann jetzt auf seine Daten nicht mehr zugreifen.
4. Gegen Bezahlung „Lösegeld“, werden die Daten wieder entschlüsselt.

⇒ Offene Erpressung des Opfers



Selbststudium

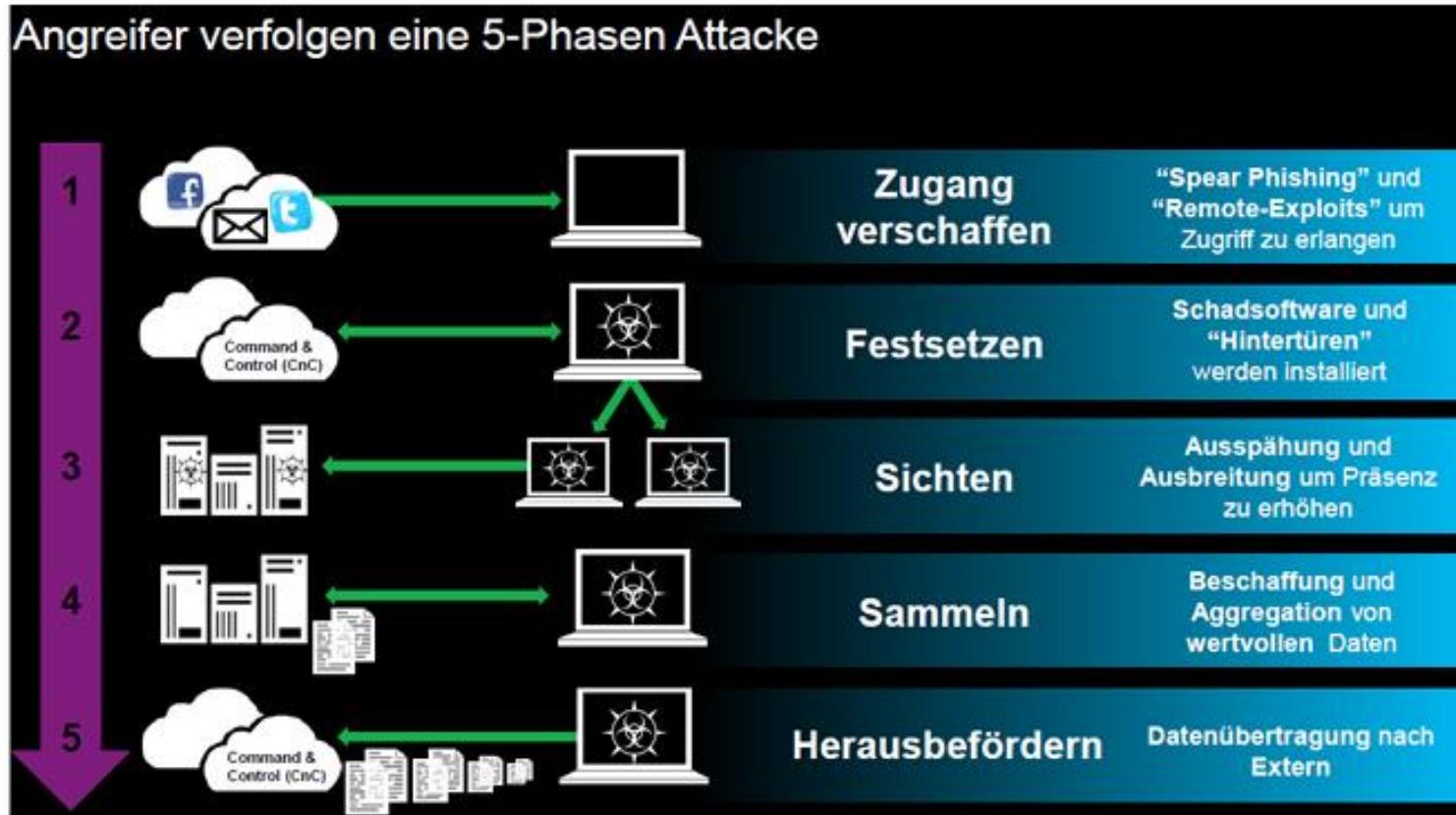


Advanced Persistent Threat (APT)

Selbststudium

https://en.wikipedia.org/wiki/Advanced_persistent_threat

Angreifer verfolgen eine 5-Phasen Attacke



Internet-Bankraub: Onlinegang stiehlt eine Milliarde Dollar

Map of Carbanak targets

Up to 100 financial institutions were hit at more than 300 IP addresses in almost 30 countries worldwide.



[IBM, Spiegel]

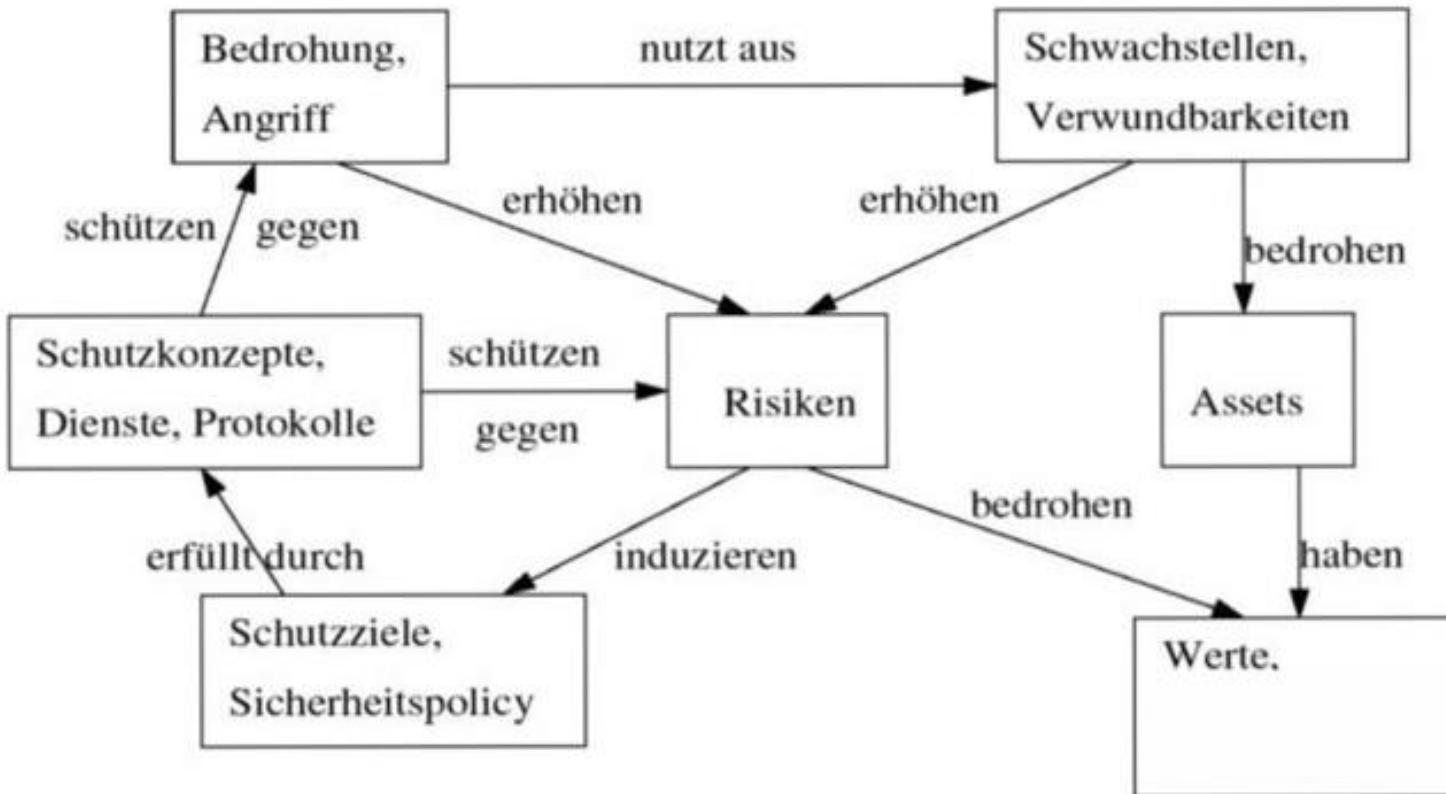
Bsp für einen APT: Carbanak

<https://securityintelligence.com/carbanak-how-would-you-have-stopped-a-1>



Zusammenhang

Selbststudium



[Eckert]



**THINK MARK THINK!
THE EXAM IS IN 2 DAYS!**



**YOUR PASSWORD SHOULDNT BE PASSWORD!
STUPID CUNT**

Komponenten einer Sicherheitsarchitektur

Selbststudium

Zugangskontrolle

- kontrollierter Zugang zu einem System
- Identifiziert & Authentifiziert das Subjekt (User)

Zugriffskontrolle:

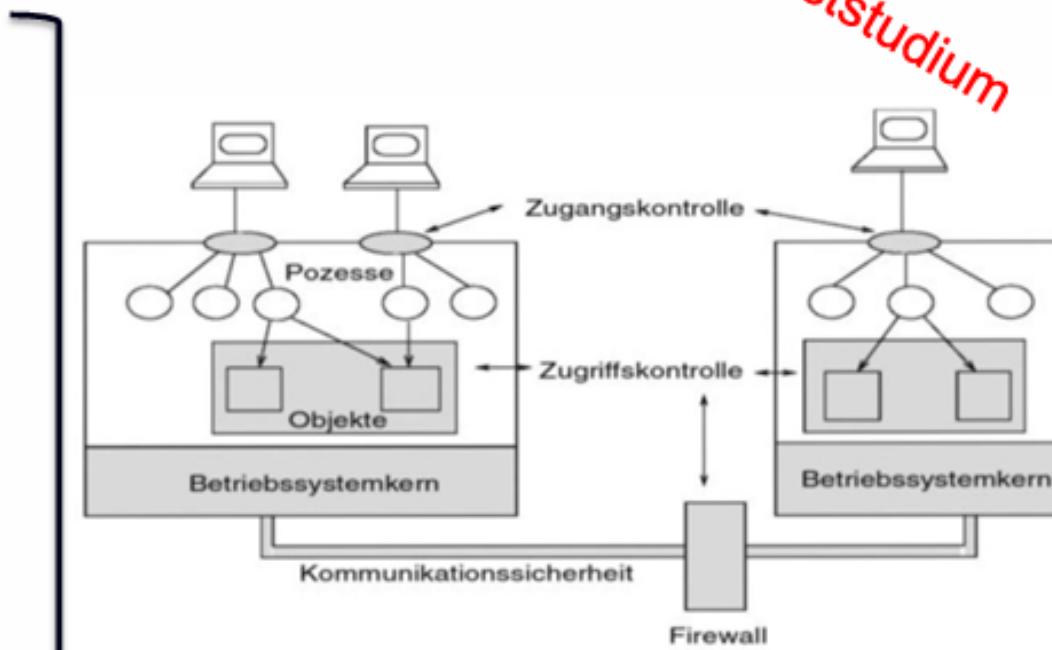
- Nur berechtigten Subjekte wird Zugriff auf zu schützende Objekte gewährt
- Gewährleistung der Datenintegrität
- Informationsflusskontrolle (falls in Policy gefordert)

Kommunikationssicherheit

- Ziel: Sichere Informationsübertragung in Rechnernetzen (bzw. Internet).
- Sichere Kommunikationsprotokolle

Weitere Komponenten

- Organisatorische Maßnahmen (z.B. Verantwortung)
- Bauliche Maßnahmen (z.B. Abtrennung)
- Administrative Maßnahmen (z.B. Audits)



Ganzheitliche Sicherheit

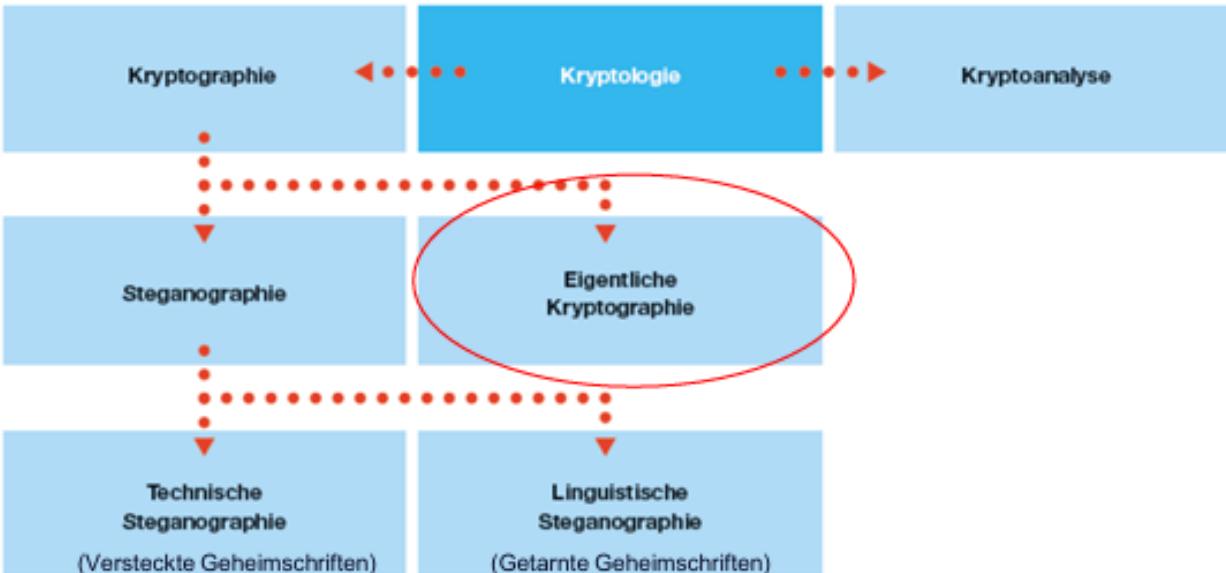
[Eckert]



Kryptologie

- Kryptologie beschäftigt sich als Wissenschaft mit der Sicherheit von Informationen.
- Der Begriff wurde bereits 1932 von dem Schweden Yves Gyldén in dem Buchtitel „cryptologue“ verwendet und hat sich über die Jahre verfeinert und verfestigt.
- Die Kryptologie gliedert sich auf in:
 - **Kryptografie:** Verschlüsselung von Informationen
 - **Kryptoanalyse:** „Entzifferung“ – das Brechen von Kryptosystemen, das Lesen von Nachrichten, ohne im Besitz des Schlüssels zu sein – auf der anderen Seite.

„Der Schutz sensibler Information ist ein Anliegen, das bis in die Anfänge menschlicher Kultur reicht.“
Otto Horak, 1991



Steganographie

- Methode die Existenz einer Nachricht zu verbergen.
- N.B. Unterschied zur Kryptographie: Veränderung einer Information, sodass sie für den Unbefugten unlesbar wird.

Bauer, Friedrich L. 2000. Entzifferte Geheimnisse: Methoden und Maximen der Kryptologie. 3., überarb. u. erw. Aufl. Springer Verlag: Heidelberg.

[IBM]



Abgrenzung: Steganografie vs. Kryptografie

- Kunst oder Wissenschaft der verborgenen Speicherung oder Übermittlung von Informationen in einem Trägermedium (Container).
- Das modifizierte Medium wird als **Steganogramm** bezeichnet.

Ziele der Steganografie:

- Verbergen von Informationen, dass ein Dritter bei Betrachtung des Steganogramms keinen Verdacht schöpft.
- Schöpft ein Dritter keinen Verdacht, dann sucht er nicht aktiv nach Informationen; das Steganogramm wird als „harmloser“ Gegenstand eingeordnet, und erfährt keine weitere Beachtung.
- Gelingt diese Täuschung: ist zugleich erreicht, dass die verborgenen Informationen nicht Dritten bekannt werden, d. h., die Geheimhaltung ist dann gewährleistet.

Unterschied Kryptografie u. Steganografie:

- Funktionsprinzip der Steganografie: ein Außenstehender erkennt die Existenz der steganografierten Information **nicht**.
- Funktionsprinzip der Kryptografie: ein Außenstehender weiß zwar um die Existenz von Informationen; aber aufgrund der Verschlüsselung ist er nicht in der Lage ist, den Inhalt zu verstehen.

<https://de.wikipedia.org/wiki/Steganographie>



Was ist Kryptografie bzw. Verschlüsselung? (engl. encryption)

- Umwandlung eines lesbaren Textes (Klartext) in einen nicht mehr les- und interpretierbaren Text (Geheimtext).
- Mithilfe eines Verschlüsselungsverfahrens, bestehend aus:
 - mathematischer Algorithmus
 - Schlüssel
- Die Wiederherstellung des Klartextes nennt man Entschlüsselung.
- Abhängigkeit der Güte einer Verschlüsselung:
 - Qualität des verwendeten Algorithmus
 - Schlüssellänge



Verschlüsselungsalgorithmus & Schlüssel

- Der Verschlüsselungsalgorithmus
 - ist eine Regel die den Verschlüsselungsprozess beschreibt.
 - gewährleistet alleine keine „Sicherheit“. „Sicherheit“ ist Abhängigkeit vom Schlüssel (bzw. Zugriff)
- Analogie: Algorithmus – Schloss
- Allgemeine Empfehlung:
 - Offenlegung des verwendeten Algorithmus.
Bildlich gesprochen: Man zeigt wie das Schloss funktioniert.
Das birgt kein Risiko, denn das Wissen über die Güte des Schlosses bedeutet nicht, dass man es öffnen kann.
Die Sicherheit wird NUR durch den Schlüssel gewährleistet.
 - Gute Algorithmen zeichnen sich dadurch aus, dass der passende Schlüssel nicht hergeleitet werden kann.



„Sicherheit“ vs. Attacke

Mögliche Attacken:

- Nicht technische Attacken – hier nicht weiter betrachtet.
- Technische Attacke: **Brute-Force-Angriff:**
Alle möglichen Schlüssel werden getestet, bis der korrekte Schlüssel gefunden ist.

Sicherheit einer Verschlüsselung:

- Einen Schlüssel zu testen kostet Rechenleistung, bzw. Zeit und Energie.
- Als „sicher“ gilt eine Verschlüsselung,
wenn die Anzahl der potenziellen Schlüssel so hoch ist, dass der Schlüssel
aufgrund mangelnder Zeit/ Rechnerkapazitäten/ Energie nicht erraten werden kann.



Ziele der Kryptografie

1. Vertraulichkeit/ Zugriffsschutz:

Nur dazu berechtigte Personen sollen in der Lage sein, die Daten oder die Nachricht zu lesen oder Informationen über ihren Inhalt zu erlangen.

2. Integrität/ Änderungsschutz:

Die Daten müssen nachweislich vollständig und unverändert sein.

3. Authentizität/ Fälschungsschutz:

Der Urheber der Daten oder der Absender der Nachricht soll eindeutig identifizierbar und seine Urheberschaft sollte nachprüfbar sein.

4. Verbindlichkeit/ Nichtabstreitbarkeit:

Der Urheber der Daten oder Absender einer Nachricht soll nicht in der Lage sein, seine Urheberschaft zu bestreiten, d. h., sie sollte sich gegenüber Dritten nachweisen lassen.



Warum Verschlüsselung ?

1. Umsetzung regulatorischer Vorgaben:

- z.B. Unternehmen, die in stark regulierten Branchen tätig sind (Gesundheits-, Finanzsektor,...)
 - Gesetze die Vorschriften für die Datensicherung verbindlich vor.
Bsp: „nur ärztliches Personal“ darf Zugriff auf medizinische Daten haben.
 - Industry Standards:
Bsp: Payment Card Industry Data Security Standards (PCI-DSS):
Gespeicherte Daten von Kreditkarteninhabern:
 - müssen sicher verschlüsselt werden
 - und Schlüssel müssen sicher aufbewahrt werden.

2. Eigeninteresse der Unternehmen:

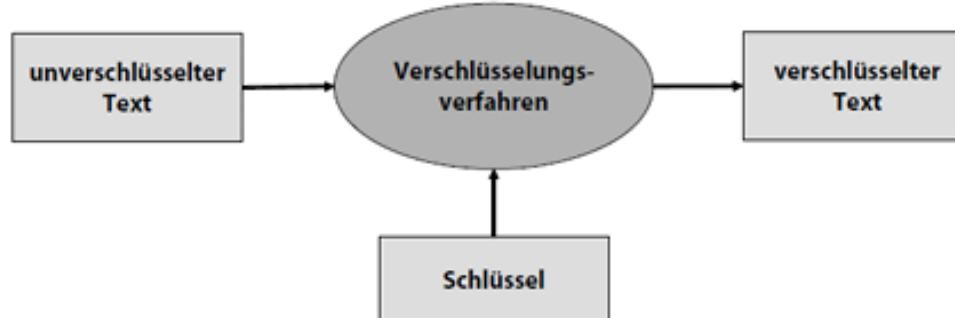
- Schutz gegen Datendiebstahl / „Cyber-Attacken“:
 - Imageschaden/ Vertrauens- und Bonitätsverlust bis hin zu strafrechtlicher Verfolgung, Haftungsschäden und Schadensersatzforderungen.
 - Die Geschäftsführung ist gesetzlich zur Sicherstellung der Vertraulichkeit bestimmter Informationen verpflichtet. Die Verantwortung für den Datenschutz kann man nicht delegieren.
=> Die Geschäftsführung haftet (siehe auch § 11 BDSG).



Verschlüsselungsverfahren

Ein Verschlüsselungsverfahren (Verschlüsselungsalgorithmus oder auch Chiffre) beinhaltet:

- eine Geheiminformation (der sogenannte Schlüssel)



Je nach Verfahren ist der Schlüssel:

- ein Passwort
- eine Geheimnummer
- oder einfach nur eine Folge von bits

Ziel eines Verschlüsselungsverfahren (am Model-Bsp):

- Alice und Bob können die ausgetauschten Nachrichten schnell und einfach verschlüsseln und wieder entschlüsseln
- Für Mallory ist es unmöglich die ausgetauschten Nachrichten ohne Kenntnis des Schlüssels zu entschlüsseln.

Funktionsweise eines Verschlüsselungsverfahren:

- wird in der Regel veröffentlicht!
- Die Sicherheit eines Verschlüsselungsverfahrens soll nur vom Schlüssel abhängen

[KRY]



Kryptografische Fachbegriffe

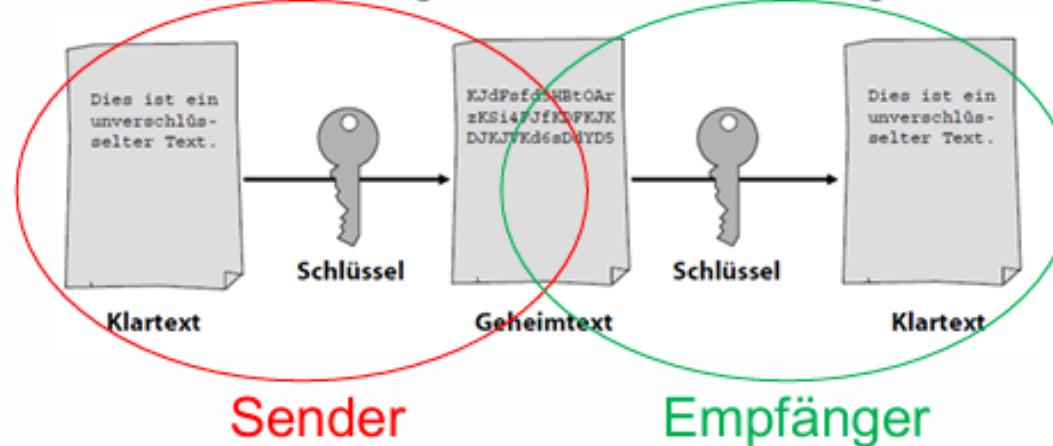
Symmetrische Verschlüsselungs-Verfahren (Secret-Key-Verfahren):

- Ver- und Entschlüsselung mit demselben Schlüssel
- Sender und Empfänger müssen vorab diesen geheimen Schlüssel vereinbart haben
- Schlüssel muss geheim gehalten werden, um die Information zu schützen.
- Sicherheitsproblem: Übergabe des geheimen Schlüssels
- Das erste bekannte dieser Art stellt die Caesar-Chiffre dar.

Asymmetrische Verschlüsselungs-Verfahren (Public- und Private-Key Verfahren)

- Verschlüsselungsverfahren, bei denen ein Teil des Schlüsselmaterials nicht geheim ist (Public-Key). Der andere Teil ist nicht nur geheim, sondern private (private key)

Verschlüsselung & Entschlüsselung:



Mathematische Darstellung

- Funktionen:
 - Verschlüsselung: e (encrypt)
 - Entschlüsselung: d (decrypt)
- Schlüssel: k (key)
- Klartext: m (message) [auch Plain Text]
- Geheimtext c (cipher)
- **Darstellung durch Formeln:**
 - $c = e(m, k)$
 - $m = d(c, k)$

[KRY]



Kryptoanalyse & Angriffe auf Verschlüsselungsverfahren

Kryptoanalyse:

Wissenschaft der Angriffe auf Verschlüsselungsverfahren

Angriff, Attacke:

Mallory versucht:

- einen Geheimtext zu entschlüsseln, den Alice an Bob schickt
- oder den Schlüssel herauszufinden

Kategorien von Angriffen:

1. Ciphertext-Only-Attacke:

1. Mallory kennt den Klartext nicht
2. Er hat nur den Geheimtext abgehört

2. Known-Plaintext-Attacke:

1. Mallory kennt den Klartext
2. Mallory kennt den Geheimtext
3. Ziel ist den Schlüssel zu berechnen

3. Chosen-Plaintext-Attacke:

1. Mallory kann den Klartext beliebig vorgeben
2. Mallory hat Zugang zur Verschlüsselung, und kann sich den Geheimtext berechnen lassen.
3. Ziel ist den Schlüssel zu berechnen

Voraussetzung für Angriffe:

Mallory kennt das Verschlüsselungsverfahren.
(Das ist eine realistische Annahme)

Bewertung:

- Ciphertext-Only-Attacke:
ist die aufwendigste Attacke, und auch die Häufigste.
- Known-Plaintext-Attacken:
 - Sind möglich, wenn sich Nachrichten oder Teile davon wiederholen.
 - Bsp: Alice E-Mails beginnen immer mit dem gleichen „Briefkopf“ etc.:
Ist dieser Briefkopf Mallory bekannt: kann mit einer Known-Plaintext-Attacke evtl. der Schlüssel ermittelt werden.
Der Schlüssel kann dann verwendet werden um den Rest der E-Mail zu entschlüsseln.
- Chosen-Plaintext-Attacke:
weniger aufwendig

[KRY]



Verschiebechiffren: Caesar-Chiffre

- war schon zu Julius Caesar's Zeiten bekannt.
- Verschlüsselungsverfahren das jeden Buchstaben durch einen anderen ersetzt.
- Jeder Buchstabe wird um eine Zahl n verschoben werden:
 - Ist $n=1$, dann gilt: aus A wird B, aus B wird C, aus C wird D, ...
 - Ist $n=2$, dann gilt: aus A wird C, aus B wird D, aus C wird E, ...
- **Verschlüsselung:**
 - Klartext:
DER MENSCH MACHT FEHLER, FUER
KATASTROPHEN IST DER COMPUTER
ZUSTAENDIG
 - Schlüssel $n=5$
 - Geheimtext:
IJW RJSXHM RFHMY KJMQJW, KZJW
PFYFXYWTUMJS NXY IJW HTRUZYJW
EZXYFJSINL
 - kennt 26 verschiedene Schlüssel
 - Besonderheit: $n=26$

Praktische Anwendung:

- Um die Caesar-Chiffre einzusetzen, kann man eine **Chiffrierscheibe** oder einen **Chiffrierschieber** verwenden. Beide Verschlüsselungswerzeuge haben eine lange Geschichte und wurden in vielen Varianten gebaut.



Einordnung:

- symmetrisches Verschlüsselungsverfahren
- monografisches Verschlüsselungsverfahren
- monoalphabetische Substitution
- Heute: Eines der einfachsten und unsichersten Verfahren - dient zur Darstellung von Grundprinzipien der Kryptologie.

<https://de.wikipedia.org/wiki/Caesar-Verschl%C3%BCsselung>



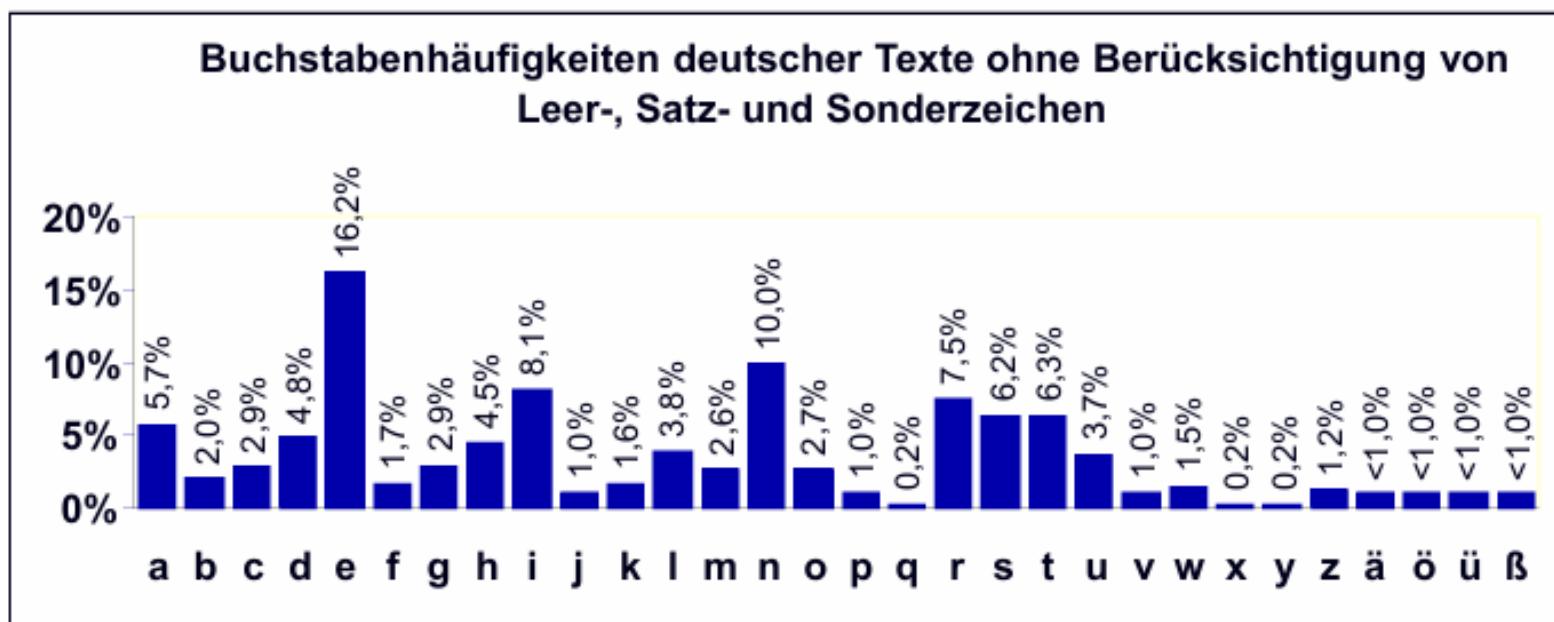
Angriffe: die Caesar Chiffre ist unsicher

Ciphertext-Only-Attacke:

- Mallory testet alle n von 1 bis 25 durch, und kann so den Schlüssel schnell ermitteln.
(mit Computerunterstützung noch schneller)
- ⇒ **vollständige Schlüsselsuche** oder auch **Brute-Force-Attacke**.

Häufigkeitsanalyse:

- Je länger der Text desto besser.
- Prinzip:
in einem Text kommen normalerweise nicht alle Buchstaben gleich häufig vor.
Bsp:
In der deutschen Sprache ist das E mit über 16,2% Prozent der häufigste Buchstabe, usw.



XOR - Verschlüsselung

- Verschiebechiffre auf dem Raum (als Alphabet interpretiert) von L-Bit-Blöcken
- Als Schlüssel dient ein fester Block k.
- Schlüssellänge ist L
- Jeder Block des Klartextes wird mit k bitweise per XOR verknüpft

Code	...0	...1	...2	...3	...4	...5	...6	...7	...8	...9	...A	...B	...C	...D	...E	...F
0...	nicht belegt															
1...	nicht belegt															
2...	SP	!	"	#	\$	%	&	'	()	*	+	,	-	.	/
3...	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
4...	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
5...	P	Q	R	S	T	U	V	W	X	Y	Z	[\]	^	-
6...	.	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
7...	p	q	r	s	t	u	v	w	x	y	z	{		}	~	
8...	nicht belegt															
9...	nicht belegt															
A...	NBSP	ı	¢	£	¤	¥		§	“	©	”	«	¬	SHY	®	-
B...	°	±	²	³	·	µ	¶	·	,	·	·	»	¼	½	¾	¼
C...	À	Á	Ã	Ä	Å	Æ	Ç	È	É	Ê	Ë	Í	Í	Í	Í	Í
D...	Đ	Ñ	Ò	Ó	Ô	Õ	Ö	×	Ø	Ù	Ú	Û	Û	Û	Û	Û
E...	à	á	ã	ä	å	æ	ç	è	é	ê	ë	í	í	í	í	í
F...	ó	ñ	ò	ó	ô	õ	ö	ø	ú	û	û	û	û	û	û	û

Klartext L = 8	D	u		b	i	s	t		g	u	t
Klartext Hexadezimal	0x44	0x75	0x20	0x62	0x69	0x73	0x74	0x20	0x67	0x75	0x74
Klartext Binär	0100 0100	0111 0101	0010 0000	0110 0010	0101 1001	0111 0011	0111 0100	0010 0000	0101 0111	0111 0101	0111 0100
Schlüssel	1001 0110	1001 0110	1001 0110	1001 0110	1001 0110	1001 0110	1001 0110	1001 0110	1001 0110	1001 0110	1001 0110
Bitweise XOR – Verschlüsselung											
Geheimtext binär	1101 0010	1110 0011	1011 0110	1111 0100	1100 1111	1110 0101	1110 0010	1011 0110	1100 0001	1110 0011	1110 0010
Geheimtext hexadezimal	0xD2	0xE3	0xB6	0xF4	0xCF	0xE5	0xE2	0xB6	0xC1	0xE3	0xE2
Geheimtext	ò	ã	¶	ó	í	å	â	¶	À	ã	â



XOR - Entschlüsselung

Klartext L = 8	D	u		b	i	s	t		g	u	t
Klartext Hexadezimal	0x44	0x75	0x20	0x62	0x69	0x73	0x74	0x20	0x67	0x75	0x74
Klartext Binär	0100 0100	0111 0101	0010 0000	0110 0010	0101 1001	0111 0011	0111 0100	0010 0000	0101 0111	0111 0101	0111 0100
Schlüssel	1001 0110	1001 0110	1001 0110	1001 0110	1001 0110	1001 0110	1001 0110	1001 0110	1001 0110	1001 0110	1001 0110
Bitweise XOR – Verschlüsselung											
Geheimtext binär	1101 0010	1110 0011	1011 0110	1111 0100	1100 1111	1110 0101	1110 0010	1011 0110	1100 0001	1110 0011	1110 0010
Geheimtext hexadezimal	0xD2	0xE3	0xB6	0xF4	0xCF	0xE5	0xE2	0xB6	0xC1	0xE3	0xE2
Geheimtext	ø	ã	¶	ó	í	å	â	¶	Á	ã	â
Schlüssel	1001 0110	1001 0110	1001 0110	1001 0110	1001 0110	1001 0110	1001 0110	1001 0110	1001 0110	1001 0110	1001 0110
Bitweise XOR – Entschlüsselung											
Klartext Binär	0100 0100	0111 0101	0010 0000	0110 0010	0101 1001	0111 0011	0111 0100	0010 0000	0101 0111	0111 0101	0111 0100

- Entschlüsselung: Gleiches Vorgehen wie bei Verschlüsselung:
Bitweises XOR



Bsp: möglicher Angriffe auf XOR: Kryptoanalyse

- alle Zeichen in der oberen Hälfte der möglichen 256 Bytes
=> Vermutung:
ein ASCII-Text mit einem Schlüssel behandelt wurde, dessen Leitbit eine 1 ist.
- Ist der Text länger, dann wird sehr oft das Leerzeichen (verschlüsselt: `\n`) dargestellt:
Folgende Annahmen führen dann sofort zum Schlüssel:
 - Es handelt sich um das Leerzeichen:
Damit ist der Klartext bekannt: 0x20 0010 0000
 - Der Ciphertext ist gegeben: 0xB6 1011 0110
⇒ Daraus lässt sich der Schlüssel k: 1001 0110 berechnen: bitweises XOR
- Häufigkeitsanalyse



Verbesserung Caesar: Freie Buchstabensubstitution

Unterschied zur Caesar-Chiffre:

Anstatt jeden Buchstaben im Alphabet zu verschieben, können Alice und Bob auch eine Tabelle aufstellen,
in der jeder Buchstabe in der oberen Zeile auf den darunter stehenden abgebildet wird.

Bsp-Tabelle:

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
N	E	U	Z	Y	O	V	D	K	T	M	F	J	R	L	B	G	H	A	C	P	W	S	Q	I	X

Klartext: GEBEN IST SELIGER DENN NEHMEN

Geheimtext: VYEYR KHC AYFKVYH ZYRR RYDJYR

Angriff:

- Eine vollständige Schlüsselsuche (Brut-Force) macht keinen Sinn – es gibt 26! mögliche Schlüssel
- Häufigkeitsanalyse führt zum Ziel

[KRY]



Zusammenfassung: Monoalphabetische Substitutions-Chiffren:

Monoalphabetische Verfahren haben alle folgende Eigenschaften

- ein bestimmter Geheimtextbuchstabe ist
immer auf den gleichen Klartextbuchstaben abgebildet
(gilt auch für Gruppen von Buchstaben – siehe Bigramm-Subst als Bsp.)
- Die Abbildung ist ein-ein-deutig: dh. Es gilt immer beides:
(auch für Gruppen von Buchstaben):
 - Verschlüsselung
Ein Klartextbuchstabe wird immer auf den gleichen **Geheimtextbuchstaben** abgebildet
 - Entschlüsselung:
Ein **Geheimtextbuchstabe** wird immer auf den gleichen Klartextbuchstaben abgebildet
- Geheimtext und Klartext setzen sich aus dem gleichen (einen) Alphabet zusammen

Schwachpunkt der Monoalphabetischen Chiffren:

Wenn Mallory im Geheimtext mehrfach den Buchstaben G findet, dann weiß er, dass sich jedes Mal der gleiche Buchstabe im Klartext dahinter verbirgt.

⇒ Häufigkeitsanalysen sind das Mittel der Wahl bei einem Angriff auf Monoalphabetische-Chiffren.

[KRY]



Vigenère-Chiffre – am Beispiel

Klartext: ALLES IST EINE FOLGE VON BITS

Schlüssel: ALICE

Vorgehen Vigenère-Verschlüsselung:

Klartext: ALLES IST EINE FOLGE VON BITS

Schlüssel: ALICE ALI CEAL ICEAL ICE ALIC

Spaltenweise -----

Addition

Geheimtext: AWIGW IDB GMNP NQPGP DQR BTÖU

Ver- und Entschlüsselung mit dem Vigenère-Tableau:

- Ein Schlüsselwort bestimmt,
wie viele und welche Alphabete genutzt werden.
- Die Alphabete leiten sich aus der Caesar-Substitution ab.

Eigenschaft: Polyalphabetische Substitution:

- Der gleiche Geheimtext-Buchstabe kann für verschiedene Klartext-Buchstaben.
- Der gleiche Klartext-Buchstabe kann auf verschiedene Geheimtext-Buchstaben abgebildet werden

Bsp: Buchstabe L in ALLES wird auf W und T abgebildet.

Buchstabe T im Geheimtext geht auf L oder I zurück

	a b c d e f g h i j k l m n o p q r s t u v w x y z ä ö ü s	
A	A B C D E F G H I J K L M N O P Q R S T U V W X Y Z Ä Ö Ü S	
B	B C D E F G H I J K L M N O P Q R S T U V W X Y Z Ä Ö Ü S A	
C	C D E F G H I J K L M N O P Q R S T U V W X Y Z Ä Ö Ü S A B	
D	D E F G H I J K L M N O P Q R S T U V W X Y Z Ä Ö Ü S A B C	
E	E F G H I J K L M N O P Q R S T U V W X Y Z Ä Ö Ü S A B C D	
F	F G H I J K L M N O P Q R S T U V W X Y Z Ä Ö Ü S A B C D E	
G	G H I J K L M N O P Q R S T U V W X Y Z Ä Ö Ü S A B C D E F	
H	H I J K L M N O P Q R S T U V W X Y Z Ä Ö Ü S A B C D E F G	
I	I J K L M N O P Q R S T U V W X Y Z Ä Ö Ü S A B C D E F G H	
J	J K L M N O P Q R S T U V W X Y Z Ä Ö Ü S A B C D E F G H I	
K	K L M N O P Q R S T U V W X Y Z Ä Ö Ü S A B C D E F G H I J	
L	L M N O P Q R S T U V W X Y Z Ä Ö Ü S A B C D E F G H I J K	
M	M N O P Q R S T U V W X Y Z Ä Ö Ü S A B C D E F G H I J K L	
N	N O P Q R S T U V W X Y Z Ä Ö Ü S A B C D E F G H I J K L M	
O	O P Q R S T U V W X Y Z Ä Ö Ü S A B C D E F G H I J K L M N	
P	P Q R S T U V W X Y Z Ä Ö Ü S A B C D E F G H I J K L M N O	
Q	Q R S T U V W X Y Z Ä Ö Ü S A B C D E F G H I J K L M N O P	
R	R S T U V W X Y Z Ä Ö Ü S A B C D E F G H I J K L M N O P Q	
S	S T U V W X Y Z Ä Ö Ü S A B C D E F G H I J K L M N O P Q R	
T	T U V W X Y Z Ä Ö Ü S A B C D E F G H I J K L M N O P Q R S	
U	U V W X Y Z Ä Ö Ü S A B C D E F G H I J K L M N O P Q R S T	
V	V W X Y Z Ä Ö Ü S A B C D E F G H I J K L M N O P Q R S T U	
W	W X Y Z Ä Ö Ü S A B C D E F G H I J K L M N O P Q R S T U V	
X	X Y Z Ä Ö Ü S A B C D E F G H I J K L M N O P Q R S T U V W	
Y	Y Z Ä Ö Ü S A B C D E F G H I J K L M N O P Q R S T U V W X	
Z	Z Ä Ö Ü S A B C D E F G H I J K L M N O P Q R S T U V W X Y	
Ä	Ä Ö Ü S A B C D E F G H I J K L M N O P Q R S T U V W X Y Z	
Ö	Ö Ü S A B C D E F G H I J K L M N O P Q R S T U V W X Y Z Ä	
Ü	Ü S A B C D E F G H I J K L M N O P Q R S T U V W X Y Z Ä Ö	
ß	ß A B C D E F G H I J K L M N O P Q R S T U V W X Y Z Ä Ö	



Vernam-Chiffre: Weiterentwicklung der Vigenère-Chiffre

Verbesserung der Vigenère-Chiffre:

- Je länger der Schlüssel gewählt, desto aufwendiger ist es die Chiffre zu brechen.
- **Spezialfall: Schlüssellänge einer Vigenère-Chiffre ist exakt so lang wie der Klartext.**
⇒ Vernam-Chiffre

Vorteil Vernam-Chiffre: NICHT zu brechen mit:

- einer **einfachen** Häufigkeitsanalyse
- einer vollständigen Schlüsselsuche

Schwachstelle Vernam-Chiffre:

- Falls der Schlüssel ein natürlichsprachliches Password ist,
ist **eine komplexe** Häufigkeitsanalyse möglich.

[KRY]



Von der Vernam-Chiffre zum One-Time Pad

Sicherheit der Vernam-Verschlüsselung:

hängt entscheidend davon ab, **wie** der Schlüsselstrom erzeugt wird.

1. Wird als Schlüssel lediglich ein langes natürlichsprachiges Passwort gewählt, so stellt die Vernam-Chiffre eine Erweiterung der Vigenère-Chiffre dar, bei der der **geheime Schlüssel die gleiche Länge wie der Klartext hat**.
⇒ Ein solches Verfahren ist nach heutigen Standards unsicher.

} Siehe vorherige Seite

2. Wird der Schlüssel von einem kryptographisch sicheren **Zufallszahlengenerator** erzeugt, so ist das resultierende Verfahren eine Stromverschlüsselung, deren Sicherheit von der des Zufallszahlengenerators abhängt:
 - Wird der Schlüssel echt zufällig erzeugt, so wird das Verfahren auch One-Time-Pad genannt.
 - **Dieses Verfahren ist perfekt sicher**

} One-Time-Pad

Bemerkung: Da das One-Time-Pad von Vernam mitentwickelt wurde, wird es oft ebenfalls als Vernam-Chiffre bezeichnet.

[KRY]



Der One-Time Pad

Mathematische Grundlage für Angriff auf Vernam-Chiffre:

1. Klartext hat eine **ungleichmäßige** Buchstabenverteilung
(Das ist bei jedem natürlichen Text der Fall)
2. Schlüssel hat ebenfalls eine **ungleichmäßige** Buchstabenverteilung
⇒ Auch beim Schlüssel wurde natürliche Sprache verwendet

Ist dass der Fall dann ist eine komplexe Häufigkeitsanalyse möglich-
(nicht in Scope dieser Vorlesung)

Vorteil One-Time-Pad:

- Schlüssel hat eine rein zufällige Buchstabenfolge,
dh. eine **gleichmäßige** Verteilung
⇒ Häufigkeitsanalyse kann keinen Erfolg haben

One-time pad

<https://www.youtube.com/watch?v=FIIG3TvQCBQ>

Gibt es überhaupt eine Methode der Kryptoanalyse, die gegen den One-Time-Pad Erfolg verspricht?

- **Nein !**
(wenn der Schlüssel wirklich zufällig gewählt ist,
und es sich damit tatsächlich
um einen One-Time-Pad handelt)
- der Geheimtext ist ebenfalls vollkommen zufällig,
und kann nicht gebrochen werden,

Mit anderen Worten:

Warum ist der One-Time-Pad unbreakable?:

- Jeder mögliche Klartext
kann in jeden möglichen Geheimtext
verschlüsselt werden,
- und das mit jeweils gleicher Wahrscheinlichkeit
⇒ Kein Ansatz für einen Angriff in der Kryptoanalyse
- ⇒ Der One-Time-Pad ist das einzige
Verschlüsselungsverfahren,
für das diese Eigenschaft gilt.
- ⇒ Es ist absolut sicher.

[KRY]



Der One-Time Pad - Einordnung

Universell verwendbar:

- Der One-Time-Pad funktioniert nicht nur mit den 26 Buchstaben des Alphabets.
- Genauso gut können auch die 256 ASCII-Zeichen oder eine andere Menge von Zeichen verwendet werden.

Praktische Anwendung des One Time Pad:

In der modernen Kryptografie werden nur die zwei Zahlen 0 und 1 verwendet.

- ⇒ Der Klartext ist in diesem Fall eine Bit-Folge, der Schlüssel ebenfalls.
- ⇒ Die Addition von einem Klartext-Bit mit einem Schlüssel-Bit entspricht dabei einer sogenannten **XOR (Exklusiv-oder-Verknüpfung)**.

Nachteile des One-Time-Pad:

1. Unhandlich:
Der Umgang mit einem Schlüssel, der genauso lang ist wie die Nachricht, ist nicht besonders handlich.
(Wollen Alice und Bob dieses Verfahren im Internet anwenden, dann muss zu jedem verschickten Bit ein Schlüssel-Bit existieren, das nur Alice und Bob bekannt ist.)
2. Erstellung von Zufallszahlen ist aufwendig!
(Es ist wesentlich schwieriger, als man denkt, große Mengen an Zufallszahlen herzustellen, die wirklich zufällig sind.)

Deshalb:

Diese Nachteile sind so gravierend, dass der One-Time-Pad in der Praxis kaum eingesetzt wird.

[KRY]



Permutationschiffren

Bei einer Permutationschiffre werden die Buchstaben des Klartexts nicht durch andere ersetzt (Substitutionschiffre), sondern in ihrer Reihenfolge vertauscht.

Beispiel

- fünf Klartextbuchstaben
 - folgende Vorschrift definiert Permutationschiffre:
 - Buchstabe 4 kommt auf Position 1,
 - 1 auf 2,
 - 2 auf 3,
 - 5 auf 4
 - und 3 auf 5.
- ⇒ Der Schlüssel ist hierbei: (4, 1, 2, 5, 3).

Klartext: ES GIBT ZWEI ARTEN

Nr: 12 3451 2345 12345

Schlüssel: 41 2534 1253 41253

Geheimtext: IE SBGE TZIW EARNT

Einordnung:

- Kryptoanalyse einer Permutationschiffre funktioniert anders als bei einer Substitutionschiffre:
Häufigkeitsanalyse führt von Buchstaben bringt nichts.
- Potentiell hohe Sicherheit möglich:
Umstellen der Buchstaben bringt oft mehr Sicherheit als das Ersetzen.
- Ansatz: Buchstabenkombinationen:
z.B. EN, CH, UND, usw. liefern Anhaltspunkte.

Nachteile:

- Known-Plaintext-Attacke sehr einfach.
Nur die ersten Buchstaben einer Nachricht reichen aus (bis zur Schlüssellänge)
- Im Geheimtext lässt sich ohne Entschlüsselung z.B. die Sprache erkennen.

[KRY]



Substitutions-Permutations-Netzwerk (SPN)

Herausforderung:

- Mehrere Substitutionen hintereinander,
 - mehrere Permutationen hintereinander
- erhöhen die Sicherheit nicht.

Ansatz:

Beides im Wechsel angewandt:

- zuerst substituieren,
- dann permutieren,
- wieder substituieren
- usw.

erhöht die Sicherheit.

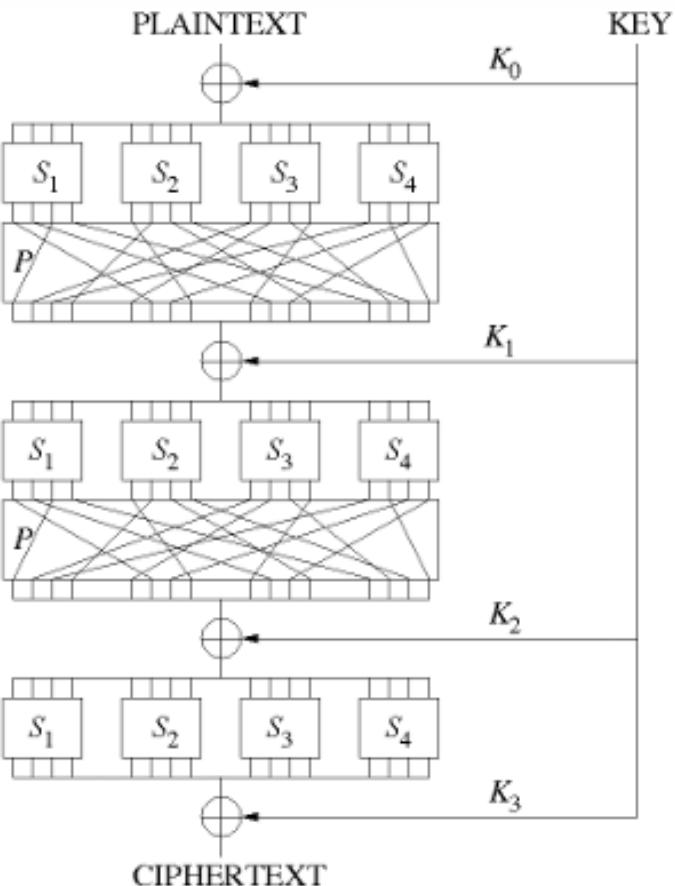
Diese Prinzip nennt sich SP-Netzwerk.

⇒ Grundlage für alle modernen
Verschlüsselungsverfahren

Prinzipieller Aufbau eines modernen
Verschlüsselungsverfahrens:

1. Runde

Substitution
Permutation



<https://de.wikipedia.org/wiki/Substitutions-Permutations-Netzwerk>

[Wikipedia]



Designkriterium Sicherheit

Es gibt zwei Möglichkeiten, ein Verschlüsselungsverfahren zu entwickeln:

1. Security by Intricacy (Komplexität):

- Möglichst komplizierter Entwurf
 - Hoffnung dass durch komplizierten Entwurf keine Schwachstellen entstehen
 - Hoffnung das Mallory Schwachstellen wg. Komplexität nicht findet
- ⇒ Findet keine relevante praktische Anwendung



Funktioniert in
der Praxis nicht

2. Durchdachtes, einfaches Design:

Ziel: keine Schwachstellen

- ⇒ Gängige Strategie

Best Practice:

- Veröffentlichung aller Design-Details (eines neuen Verfahrens)
- Verfahren gilt nur dann als sicher:
wenn seine Veröffentlichung der Sicherheit nicht schadet

[KRY]



Designkriterium Schlüssellänge

Schlüssellänge	Anzahl der Schlüssel	Dauer einer vollständigen Schlüsselsuche
40 Bit	$1,1 \cdot 10^{12}$	1,3 Sekunden
56 Bit	$7,1 \cdot 10^{16}$	24 Stunden
64 Bit	$1,8 \cdot 10^{19}$	256 Tage
80 Bit	$1,2 \cdot 10^{24}$	45.965 Jahre
128 Bit	$3,4 \cdot 10^{38}$	$1,3 \cdot 10^{19}$ Jahre
192 Bit	$6,3 \cdot 10^{57}$	$2,4 \cdot 10^{38}$ Jahre
256 Bit	$1,2 \cdot 10^{77}$	$4,4 \cdot 10^{57}$ Jahre



Sicher
Zusätzlicher
Puffer
bzw.
Paranoia

[KRY]



Designkriterium Ressourcenverbrauch

Verschlüsselung in (vergleichsweise) leistungsschwacher Hardware

Anforderungen:

- kein umfangreicher Programmcode
- geringer Bedarf an Arbeitsspeicher
- geringer Energieverbrauch

=> Spezialisierung der Verschlüsselungsverfahren:

z.B. auf ressourcenschwache Hardwareumgebungen mit meist nur kurze Nachrichten die es zu verschlüsseln gilt.

32C3: Verschlüsselung gängiger RFID-Schließanlagen geknackt

<http://www.heise.de/newsticker/meldung/32C3-Verschlüsselung-gängiger-RFID-Schliessanlagen-geknackt-3056646.html>

[KRY]



Designkriterium Verschlüsselungsgeschwindigkeit

Bsp:

- Verschlüsselung ganzer Festplatten
- Telefongespräch in Echtzeit verschlüsseln
- viele Verschlüsselungsverfahren laufen auf schwacher Hardware z.B. einer Smartcard

⇒ Die Dauer (Geschwindigkeit) der Ver- und Entschlüsselung ist wichtig für die praktische Anwendbarkeit

Ziele:

Ein Verschlüsselungsverfahren ist im Idealfall

- auf allen gängigen Plattformen performant zu implementieren
- erzielt sowohl in **Hardware** als auch in **Software** hohe Geschwindigkeiten.

[KRY]



Designkriterium Einfachheit

- Verschlüsselungsverfahren sind typischerweise nicht aus hochkomplexen mathematischen Funktionen zusammengesetzt
- Bsp DES: setzt sich aus drei der einfachsten Bit-Operationen zusammen:
XOR, Substitution, Permutation.
- Typischerweise werden die folgenden einfachen Operationen angewandt:

Zeichen	Name	Beispiel
\oplus	exklusives Oder	$1110 \oplus 1011 = 0101$
$+$	Addition	$1110 + 1011 = 1001$
$-$	Subtraktion	$1110 - 1011 = 0011$
$<<$	Linksverschiebung	$1110 << 2 = 1000$
$<<<$	Linksrotation	$1110 <<< 2 = 1011$
$>>$	Rechtsverschiebung	$1110 >> 2 = 0011$
$>>>$	Rechtsrotation	$1110 >>> 2 = 1011$
\vee	Oder	$1110 \vee 1011 = 1111$
\wedge	Und	$1110 \wedge 1011 = 1010$
\parallel	Konkatenation	$1110 \parallel 1011 = 11101011$

[KRY]



Advanced Encryption Standard (AES)

- momentan bedeutendste symmetrische Verschlüsselungsverfahren
- Ging als Sieger aus dem DES-Nachfolge-Wettbewerb hervor
- Seit November 2001 in den USA offiziell standardisiert,
wird dort seitdem für die Verschlüsselung staatlicher Dokumente verwendet.
- 2003: AES wird verwandt für die Verschlüsselung von Daten der höchsten
Geheimhaltungsstufe (Top Secret)
⇒ Zum ersten Mal ist damit ein Verschlüsselungsverfahren, das für derartige
Informationen eingesetzt wird, öffentlich bekannt.
- Zahlreiche staatliche Institutionen in anderen Ländern nutzen AES (auch BRD)
- Das Verfahren ist nicht patentiert und damit frei verwendbar.

https://de.wikipedia.org/wiki/Advanced_Encryption_Standard



Advanced Encryption Standard (AES) – Sicherheit

- Hohe Verschlüsselungsgeschwindigkeit
- Schlüssellänge von mindestens 128 Bit: vollständige Schlüsselsuche ist aussichtslos
- Über ein Jahrzehnt nach Einführung/ Veröffentlichung AES:
nach wie vor **keine Schwäche bekannt**,
die auch nur annähernd eine praktische Bedeutung hat!
- AES gilt nach wie vor als **sehr sicher**

https://de.wikipedia.org/wiki/Advanced_Encryption_Standard



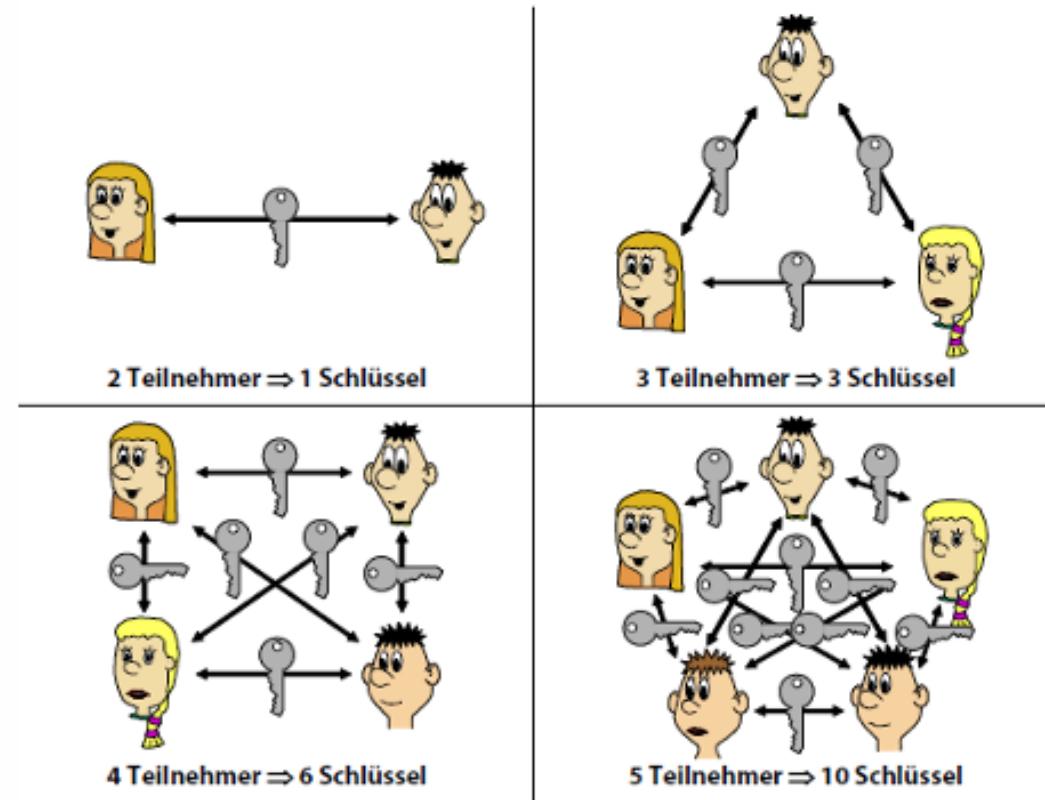
Motivation für Asymmetrische Verschlüsselung (II)

Benötigte Schlüssel:

- N: Teilnehmer
- Anzahl keys = $\frac{N}{2} \times (N - 1)$

Bsp:

- ⇒ Bei 5 Teilnehmern müssen bereits 10 Schlüssel ausgetauscht werden
- ⇒ Bei 100 Teilnehmern müssen mehrere tausend Schlüssel getauscht werden
- ⇒ Das Schlüsseltauschproblem ist **praktisch gesehen ein MASSIVES Problem**



[KRY]



Motivation für Asymmetrische Verschlüsselung (III)

Möglichkeiten des Schlüsselaustausches:

1. „Out of Band“ Schlüsselaustausch

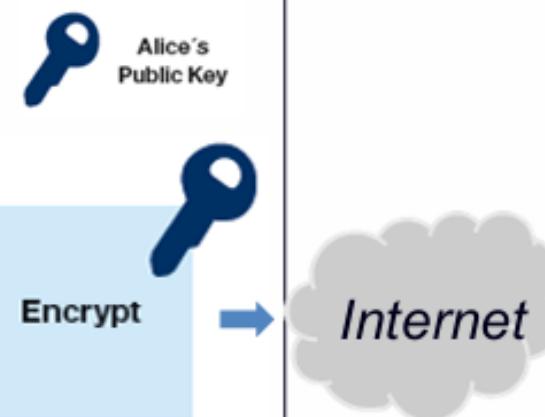
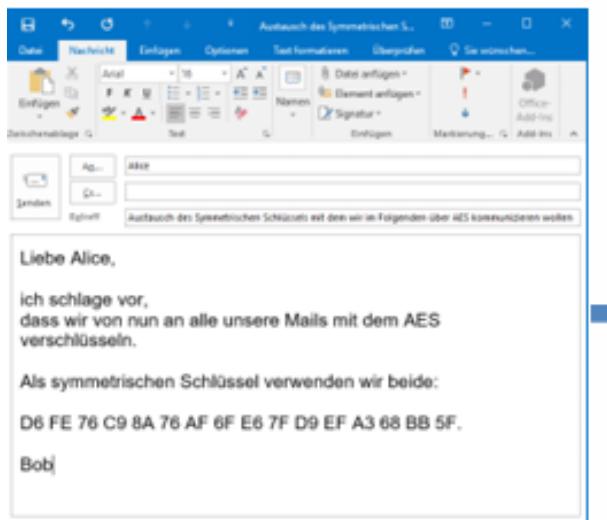
(also z.B. ein persönliches Treffen)

bei mehreren Tausend Schlüsseln die bereits in kleinen Firmen zu tauschen sind
=> nicht machbar

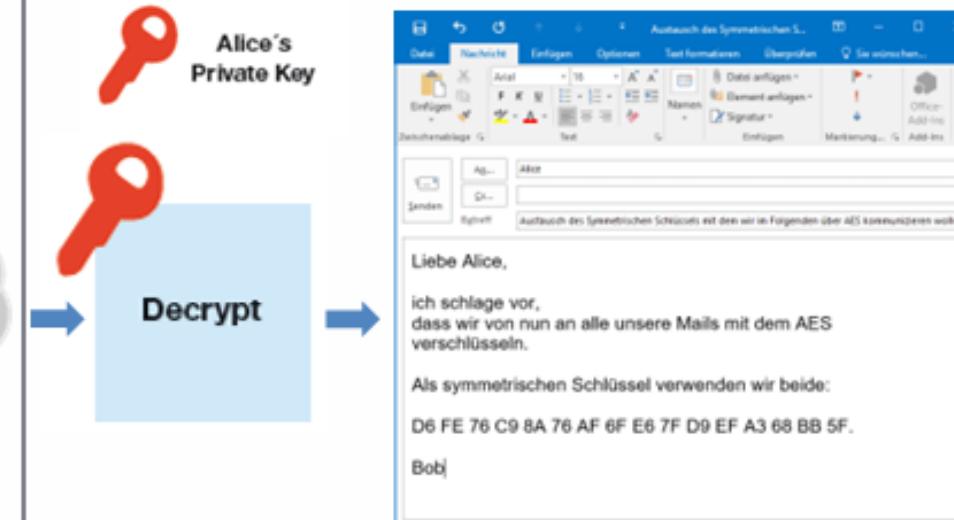
2. Asymmetrische Kryptografie

- Bob teilt Alice per mail den symmetrischen Schlüssel mit.
- Diese mail wird asymmetrisch verschlüsselt mit Alice's Public Key,
- und asymmetrisch entschlüsselt mit Alice's Private Key

Bob's Laptop



Alice's Laptop



Asymmetrische Verschlüsselung - Bsp

Der User Alice generiert sich einen Public Key und einen Private Key.

1. Ihren Public Key veröffentlicht Alice
2. Ihren Private Key hält sie geheim

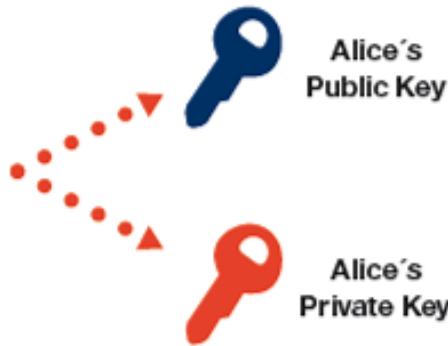
ALICE

52ED879E
70F71D92

Big Random
Number



KeyGen
Function



Öffentlich:

Diesen nutzt jeder Kommunikationspartner von Alice.

Geheim:

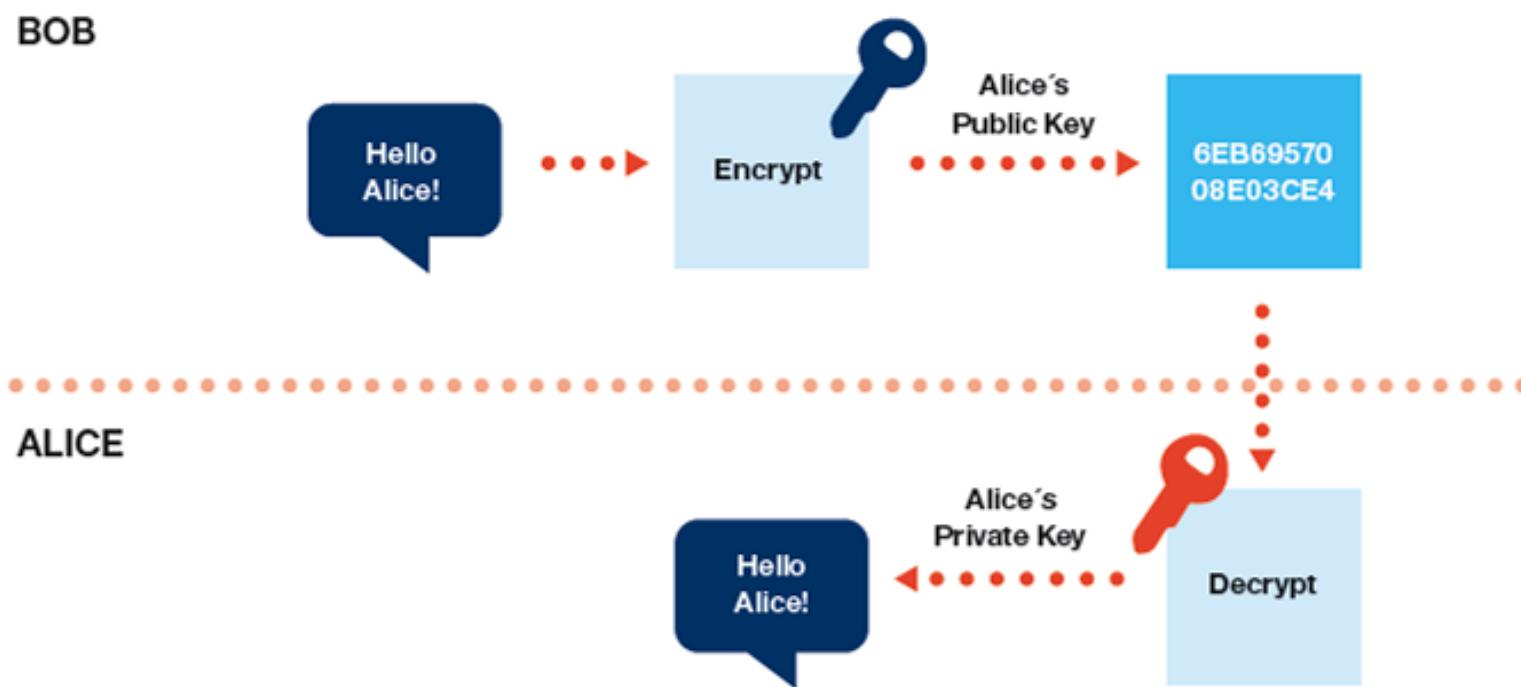
Nur Alice hat den private key



Asymmetrische Verschlüsselung - Bsp

Der User Bob möchte Alice eine verschlüsselte Nachricht (Hello Alice!) schicken:

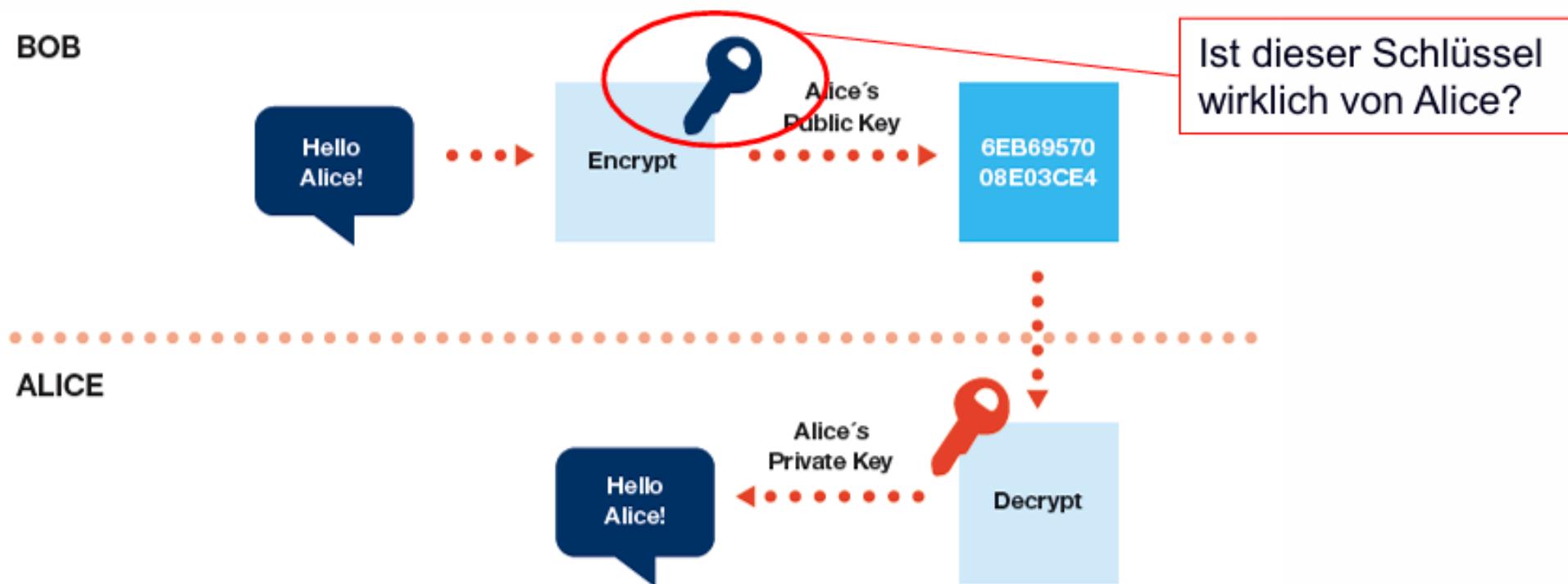
1. Bob nimmt Alice' Public Key und verschlüsselt die Nachricht.
2. Jetzt kann nur noch Alice die Nachricht entschlüsseln, mit ihrem Private Key



Asymmetrische Verschlüsselung - Bsp

Kritische Situation:

1. Bei der Übergabe des Public Keys muss sichergestellt werden, **dass er auch tatsächlich von dem Absender stammt**, von dem er erwartet wird. #
2. Lösung: Übergabe des Public Keys über eine Beglaubigungsstelle (CA) die Zusammengehörigkeit von Person und Public Key bestätigen.



[IBM]



Abschließende Betrachtung des Beispiels: Welche Ziele der Kryptographie wurden erreicht?

1. Vertraulichkeit/Zugriffsschutz:
Bedingt – nur Alice kann die Nachricht lesen.
Einschränkung: Wenn Bob den tatsächlichen public key von Alice genommen hat, und ein Angreifer ihm keinen anderen public key untergeschoben hat.
2. Integrität/Änderungsschutz:
Nein – ein Angreifer könnte die Daten von Bob gestohlen haben vor der Verschlüsselung.
Diese verändert haben, und dann verschlüsselt an Alice geschickt haben.
3. Authentizität/Fälschungsschutz:
Nein – der Sender ist nicht authentifiziert im Beispiel
4. Verbindlichkeit/Nichtabstreitbarkeit:
Nein – der Sender ist nicht authentifiziert im Beispiel

⇒ Das einfache Beispiel genügt in keiner Weise den Anforderungen/ Zielen



Moderne Kryptografie - Asymmetrische Verschlüsselung – RSA (I)

Berechnung des Public-Key:

1. Finde 2 (sehr große) Primzahlen: p und q

- Sehr groß bedeutet eigentlich "unvorstellbar groß":
dh. Zahlen mit "mehreren hundert Stellen":
Größenordnung z.B. 10^{500}
zum Vergleich: unser Universum ist ungefähr 10^{18} Sekunden alt.
=> Diese Primzahlen sind unvorstellbar viel größer – als das Universum alt ist...

2. Bilde daraus das Produkt: $N = p \times q$

Empfehlung BSI:

N : unsigned int mit ~3.000 bit !

3. Berechne: $\varphi(N) = (p - 1) \times (q - 1)$

Bemerkung:

$\varphi(N)$: ebenfalls unsigned int mit ~3.000 bit !

4. Wähle eine zu $\varphi(N)$ teilerfremde Zahl e :

$$\text{ggT}(e, \varphi(N)) = 1$$

Empfehlung BSI:

$$(2^{16} + 1) < e < (2^{256} - 1)$$

Background:

Eulersche Funktion $\varphi(N)$:

https://de.wikipedia.org/wiki/Eulersche_Phi-Funktion

- Allgemein:

$\varphi(N)$ gibt für jede natürliche Zahl N an, wie viele zu N teilerfremde natürliche Zahlen es gibt, die nicht größer als N sind.

Bsp: $N=6$:

Die Zahlen 1 und 5 sind teilerfremd

$$=> \varphi(6) = 2$$

- Spezialfall: $N = \text{Primzahl1} \times \text{Primzahl2}$

dann gilt:

$$\varphi(N) = (\text{Primzahl1} - 1) \times (\text{Primzahl2} - 1)$$

$$\text{bzw: } \varphi(N) = (p - 1) \times (q - 1)$$

Der Public-Key ist damit definiert: (e, N)

Auf keinen Fall veröffentlichen:

$p, q, \varphi(N)$ UNBEDINGT privat halten!



Moderne Kryptografie - Asymmetrische Verschlüsselung – RSA (II)

Berechnung des Private-Key:

5. Berechne d mit: $e \times d \equiv 1 \text{ mod } \varphi(N)$
bzw.: $e \times d + k \times \varphi(N) = 1$
mit dem erweiterten euklidischen Algorithmus

Background:

Erweiterter Euklidischer Algorithmus:

https://de.wikipedia.org/wiki/Erweiterter_euklidischer_Algorithmus

https://de.wikipedia.org/wiki/Lineare_diophantische_Gleichung

Der private-Key ist damit definiert: (d, N)

Auf keinen Fall veröffentlichen:

$p, q, \varphi(N)$, und d : UNBEDINGT privat halten!

Es wird nur der Public-Key: (e, N) veröffentlicht!



Moderne Kryptografie - Asymmetrische Verschlüsselung – RSA (III)

Verschlüsselung:

gegeben:

- der Public-Key (e, N)
- der Klartext K : z.B. „Hello Alice“

1. Der Klartext wird z.B. in 128 Bit-Blöcke „zerteilt“: dh. z.B. 16 Buchstaben.
2. Dann wird der 128 Bit-Block als natürliche Zahl interpretiert: K
 K ist eine große Zahl: maximal: 2^{128}
3. Anschließend wird der Geheimtext G berechnet:

$$G = K^e \bmod N$$

Die Verschlüsselung ist:

- Theoretisch sehr einfach
- Praktisch ein sehr großer Aufwand im Computer:
 K und e sind sehr große Zahlen => K^e ist eine umfangreiche Rechenaufgabe für einen Computer
z.B. Dauer Encryption: ca. 0,05 ms

Background:

Modulo-Rechnung

Annahme:

Dauer Encryption ca. 0,05ms

Gesucht:

1. Wie lange dauert es ein ppt mit ca. 5 Mbyte zu encrypten?

$$t = 5 \times 10^6 \times 8 \times 0,05 \times \frac{10^{-3}}{128} \text{ sec} = \\ = \text{ca. 15 sec}$$

2. Wie lange dauert es eine ganze Harddisk voll mit Bildern: ca. 1 Tbyte zu encrypten?

$$t = 1 \times 10^{12} \times 8 \times 0,05 \times \frac{10^{-3}}{128} \text{ sec} = \\ = \text{ca. 312.000 sec}$$

Frage:
Warum nicht unbedingt praktikabel für große Datenmengen ?



Moderne Kryptografie - Asymmetrische Verschlüsselung – RSA (IV)

Entschlüsselung:

gegeben:

- der Private-Key (d, N)
- der Geheimtext G

Mathematischer Background:

Modulo-Rechnung – sonst nichts.

1. Berechnung des Klartext K :

$$K = G^d \bmod N$$

2. K ist dann wieder die natürliche Zahl:
als 128 Bit-Block dargestellt.
3. Dieser 128 Bit-Block entspricht dem
Klartext vor der Verschlüsselung

Auch die Entschlüsselung ist:

- Theoretisch sehr einfach
- Praktisch ein sehr großer Aufwand im Computer.
(analog zur Verschlüsselung)



Moderne Kryptografie - Asymmetrische Verschlüsselung – RSA (V)

Begrenzung der Verschlüsselung:

$$G = K^e \bmod N$$

1) Rechen-Zeit:

1) Annahme:

Dauer Encryption ca. 0,05ms

2) Gesucht:

1) Wie lange dauert es ein ppt mit ca. 5 Mbyte zu encrypten?

~ 15 sec

2) Wie lange dauert es eine ganze Harddisk voll mit Bildern: ca. 1 Tbyte zu encrypten?

~ 300.000 sec / ca. 80 Stunden

2) Speicherplatz für den Geheimtext:

1) Annahme:

Klartext K wird in 128 bit Blöcke aufgeteilt

2) Geheimtext G ist dann: $0 < G < N$

Annahme: **N ist eine unvorstellbar große Zahl: Bsp 2^{2048} (im Dezimal-System: mehrere hundert Stellen)**

bzw. 2048 bit

3) Speicherbedarf für den Geheimtext:

Jeder Klartext-Block mit 128 bit wird auf einen 2048 bit Geheimtext-Block abgebildet.

=> dh. z.B. 16 mal soviel Speicher.

Bsp: ein Powerpoint mit 3 Mbyte im Klartext, benötigte dann 48 Mbyte an Speicherplatz im Geheimtext...



Moderne Kryptografie - Asymmetrische Verschlüsselung – RSA (VI)

Verschlüsselung mit dem Public Key: **Verschlüsselung mit dem Private Key:**

1. Verschlüsselung

$$G = K^e \text{ mod } N$$

2. Entschlüsselung

$$K = G^d \text{ mod } N$$

3. Zusammenhang:

$$K = (G)^d \text{ mod } N$$

1. Verschlüsselung: Geheimtext

$$G' = K^d \text{ mod } N$$

2. Entschlüsselung

$$K = G'^e \text{ mod } N$$

3. Zusammenhang:

$$K = (G')^e \text{ mod } N$$

$$K = (K^e)^d \text{ mod } N$$

$$K = (K)^{(e \times d)} \text{ mod } N$$

$$K = (K)^{(d \times e)} \text{ mod } N$$

$$K = (K^d)^e \text{ mod } N$$

$$K = (K^d)^e \text{ mod } N$$

Prinzip:

Es kann verschlüsselt werden:

1. mit dem public key: e
2. als auch mit private key: d

Entschlüsselung dann mit dem jeweiligen anderen zugehörigen key!



Moderne Kryptografie - Asymmetrische Verschlüsselung– RSA (VII)

Lösung der 4 Sicherheitsziele mit doppelter Anwendung der Asymmetrischen Verschlüsselung:

Bob:

Public-Key-Bob: (e_{Bob}, N_{Bob})

Private-Key-Bob: (d_{Bob}, N_{Bob})

Alice:

Public-Key-Alice: (e_{Alice}, N_{Alice})

Private-Key-Bob (d_{Alice}, N_{Alice})

Bob möchte den Klartext: K : „Hallo Alice“ an Alice schicken UND alle Sicherheitsziele erfüllen.

1. Bob:

1. generiert den Klartext K

2. Bob verschlüsselt K mit public-Key-Alice:

$$G1 = K^{e_{Alice}} \text{ mod } N_{Alice}$$

3. Bob verschlüsselt $G1$ mit private-Key-Bob

$$G = G1^{d_{Bob}} \text{ mod } N_{Bob}$$

4. Bob versendet den Geheimtext G

2. Alice

1. empfängt Geheimtext G

2. Alice entschlüsselt G mit public-key-Bob

$$G1 = G^{e_{Bob}} \text{ mod } N_{Bob}$$

3. Alice entschlüsselt $G1$ mit private-key-Alice

$$K = G1^{d_{Alice}} \text{ mod } N_{Alice}$$

4. Alice hat den Klartext K



Moderne Kryptografie - Asymmetrische Verschlüsselung

Diffie-Hellman: Methode zum Schlüsselaustausch (I)

Ablauf

1. Öffentlich:
Alice und Bob einigen sich auf:
 1. Eine Primzahl p
Empfehlung BSI:
Länge von p : mindestens 3.000 bit!
 2. Eine natürliche Zahl g
mit $g < p$
Empfehlung BSI:
Länge von g : mindestens 250 bit!
 3. p und g werden in einem Klartext per mail
(unverschlüsselt) ausgetauscht – z.B. per mail
2. Geheim:
 1. Alice wählt eine natürliche Zahl x
mit $x < p$
Empfehlung BSI: Gleichverteilt!
 2. Bob wählt eine natürliche Zahl y
mit $y < p$
Empfehlung BSI: Gleichverteilt!
 3. x und y halten Alice und Bob jeder für sich privat.
3. Alice:
 1. berechnet: $a = g^x \text{ mod } p$
 2. sendet a an Bob unverschlüsselt!
4. Bob:
 1. berechnet: $b = g^y \text{ mod } p$
 2. sendet b an Alice unverschlüsselt!

Alice:

- hat folgende Zahlen:
 - p, g (nicht geheim)
 - x (privat) Private-Key-Alice
 - b (nicht geheim, von Bob empfangen) Public-Key-Alice
- berechnet:
 - $k1 = b^x \text{ mod } p$

Bob:

- hat folgende Zahlen:
 - p, g (nicht geheim)
 - y (privat) Private-Key-Bob
 - a (nicht geheim, von Alice empfangen) Public-Key-Bob
- berechnet:
 - $k2 = a^y \text{ mod } p$

Prinzip: Es gilt: $k1 = k2 = k$

Vorteil:

- Wenn k groß genug (z.B. 128bit) – kann Mallory k nicht berechnen!
- k kann somit als Schlüssel dienen.



Hybride Verschlüsselung

- Kombination der Vorteile des symmetrischen und des asymmetrischen Verfahrens
- Motivation:
 - Asymmetrische Verschlüsselung hat kein Problem mit Schlüsselaustausch
 - Ist aber deutlich aufwendiger als die Symmetrische (Zeit und Speicherplatz)
- Prinzip:
 - Die eigentlichen Daten werden symmetrisch verschlüsselt.
 - Hierzu wird für jede Datenübertragung ein eigener Schlüssel (Session Key) verwendet.
 - Die Session Keys werden durch asymmetrische Verschlüsselung übermittelt.
 - Da jetzt nur noch die Schlüssel – und nicht mehr die großen Datenpakete – asymmetrisch verschlüsselt werden, wirkt sich die nachteilige Komplexität des asymmetrischen Verfahrens kaum aus.
- Vorteil:
 - Geschwindigkeit der symmetrischen Verschlüsselung
 - Sicherheit der asymmetrischen Verschlüsselung beim Schlüsselaustausch



Danke, dass ihr auch Info-Posts einen Like gebt!

