

# RNVS: Typische Klausuraufgaben

Inklusive Lösung, wenn vorhanden. Nach Alphabet und Klausur bzw. Übungsjahr sortiert.

**Wichtig: Alle Angaben ohne Gewähr, besonders eigene Lösungen. Bei Unklarheiten lieber die Übungsleitung fragen!**

Quelle:

<https://gaf.fs.lmu.de/klausuren/Klausuren/%28Medien-%29Informatik/Bachelor/Rechnernetze%20und%20verteilte%20Systeme/>

Username: klausuren

Password: pennyisafreeloader

## Ältere Aufgaben

Diese sind hier ohne Bild, da es nicht so wahrscheinlich ist, dass sie drankommen:

### Vielfachzugriff

Siehe 2014 und 2015 Klausur

### Prüfsummen und Übertragungsfehler

siehe 2011 3 und 2014 Klausur IV.

### PPP

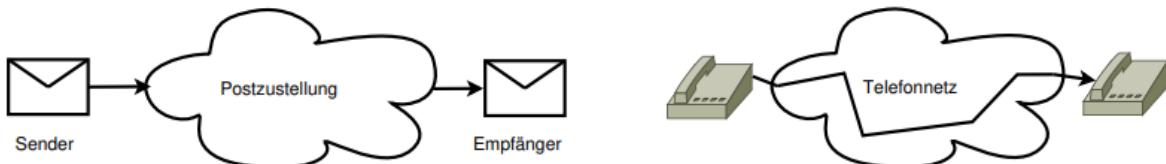
2009 Klausur und 2013 Klausur

## Allgemeine Fragen und Multiple Choice

### 2022 Übung

#### 3. Verbindungslose und verbindungsorientierte Kommunikation (H)

Ein Kriterium, um eine Kommunikationsbeziehung genauer zu charakterisieren, ist die Einteilung in verbindungslose und verbindungsorientierte Kommunikation. Beispiele aus dem Alltag wären z.B. die Briefpost und ein Telefongespräch.



1. Erläutern Sie kurz den konzeptionellen Unterschied zwischen verbindungsloser und verbindungsorientierter Kommunikation!
2. Nennen Sie je zwei Beispieldienste!
3. Unter welchen Bedingungen ist eine verbindungslose Kommunikation von Vorteil?

### Aufgabe 3: Verbindungslose und verbindungsorientierte Kommunikation

a)

Verbindungsorientiert	Verbindungslos
Zuerst Steuerpakete zum Verbindungsaufbau	Direkt Nutzdaten losschicken
Nutzdaten brauchen keine Zielinformationen	Jedes Paket mit Zieladresse versehen
Exklusive Verbindungsnutzung	Geteilte Ressourcen

b)

Verbindungsorientiert	Verbindungslos
Telefon, ISDN, Mobilfunk (ohne VoIP), Standleitung	IP, Post, Ethernet, Broadcasting

c)

- Bei sehr kurzen Kommunikationen (DNS, Ping etc.)
- Stark Schwankendes Datenaufkommen (Internetnutzung, Post)
- Kommunikation ohne inhaltlicher Vertraulichkeit und ohne Schaden bei Verlust (DNS [Vertraulichkeit inzwischen kritisch darum DoH])

## Adressierung & CIDR und Subnetting & Internet Protocol

### 2009 Klausur = 2014 Probeklausur

#### 5. Internet Protocol

- (a) Ursprünglich wurde der Adressraum für Internetadressen in Klassen aufgeteilt.
- Aus welchen zwei Teilen besteht demzufolge eine IPv4-Adresse?

**Lösung:**

Netz-ID, Host-ID

- Nennen Sie einen **Vorteil** und einen **Nachteil** der klassenbasierten Adressvergabe.

**Lösung:**

*Vorteil:* anhand Netz-ID/Präfix können schnell Routing-Entscheidungen getroffen werden; Adressraum leichter zu verwalten

*Nachteil:* großer Teil der Adressen bleibt unbenutzt

- Mit CIDR wurde ein flexibleres Schema für die Vergabe von Adressräumen benutzt. Worin besteht der Unterschied zur klassenbasierten Aufteilung des Adressraums?

**Lösung:**

Länge der Netz-ID ist variabel (nicht an Klassen gebunden)

- Wie lang (in Bits) darf eine Netz-ID für ein IPv4-basiertes Subnetz mit 58 Hosts höchstens sein?

**Lösung:**

26

- Wie lautet die Netzmaske für das Subnetz 192.168.218.0/28? Machen Sie Ihre Angabe in der Form r.s.p.q mit  $r, s, p, q \in \{0, \dots, 255\}$ , d.h. in dezimaler Schreibweise.

**Lösung:**

255.255.255.240

- (b) Nennen Sie einen Fall, in dem IPv4-Pakete fragmentiert werden müssen!

**Lösung:**

wenn Paketlänge > MTU auf Pfad

- (c) Im Internet kann mittels des Internet Control Message Protocol (ICMP) signalisiert werden, dass kein Weg zum Ziel eines IP-Paketes ermittelt werden kann (*destination unreachable*). Nennen Sie zwei weitere Meldungen, die mittels ICMP gesendet bzw. empfangen werden können!

**Lösung:**

Zum Beispiel:

- echo request
- time exceeded

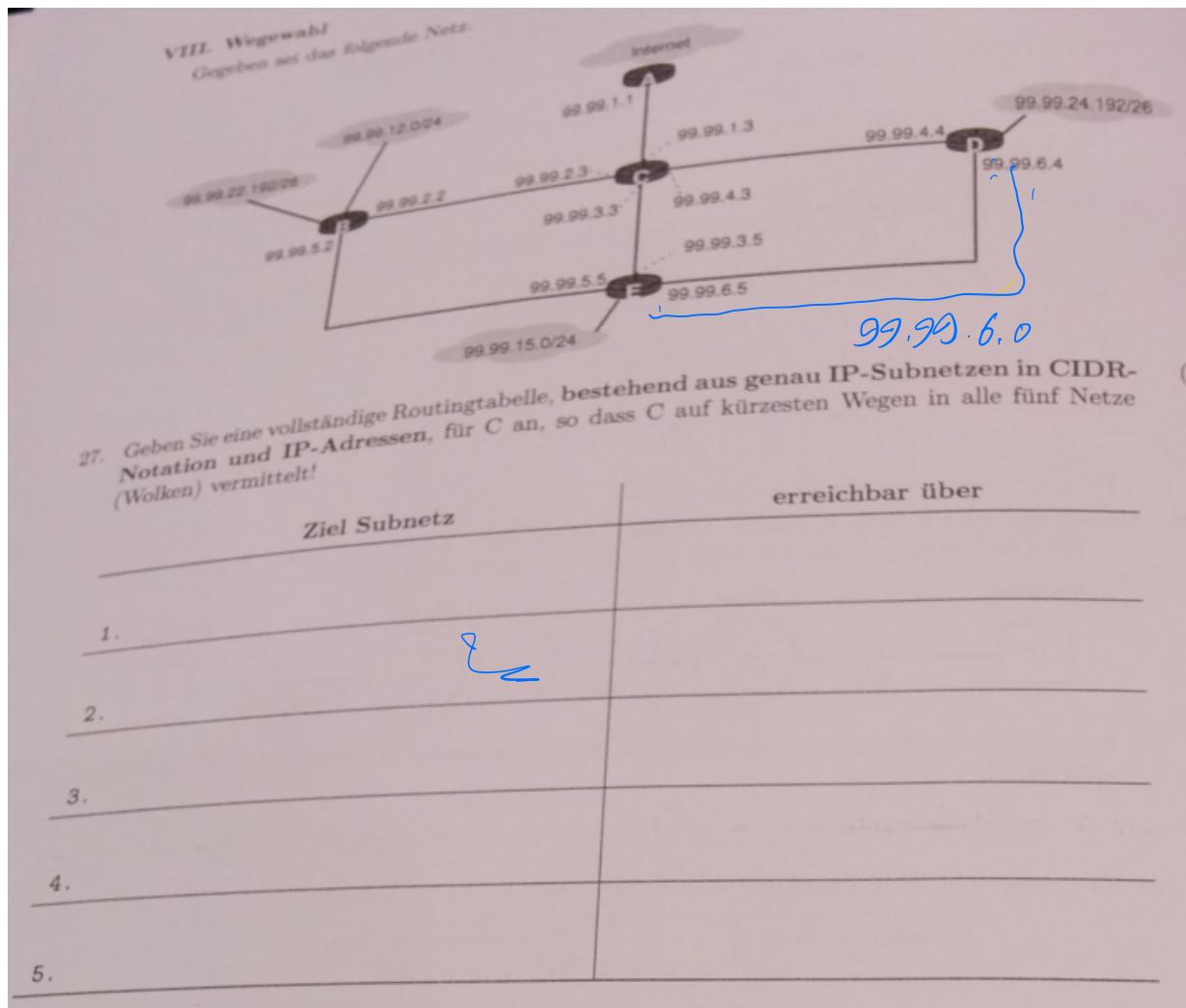
- (d) Zwischen autonomen Systemen werden andere Routing-Protokolle eingesetzt als innerhalb. Nennen Sie einen Grund dafür (mit kurzer Erklärung)!

**Lösung:**

Zum Beispiel:

- verschiedene Metriken/Entscheidungskriterien für Wegewahl

## 2013 Klausur = 2014 Klausur

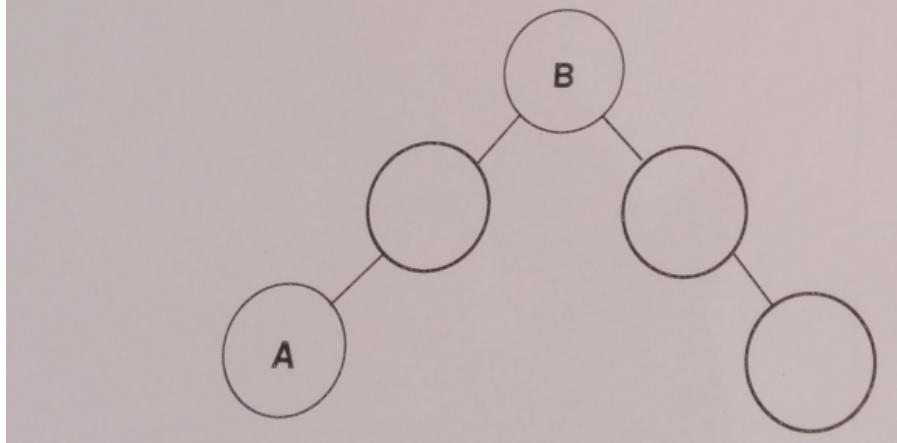


27. Ziel Subnetz | erreichbar über

1. 99.99.15.0/24 | 99.99.3.5
2. Internet | 99.99.1.1

3. 99.99.24.192/26 | 99.99.4.4
4. 99.99.22.192/26 | 99.99.2.2
5. 99.99.12.0/24 | 99.99.2.2

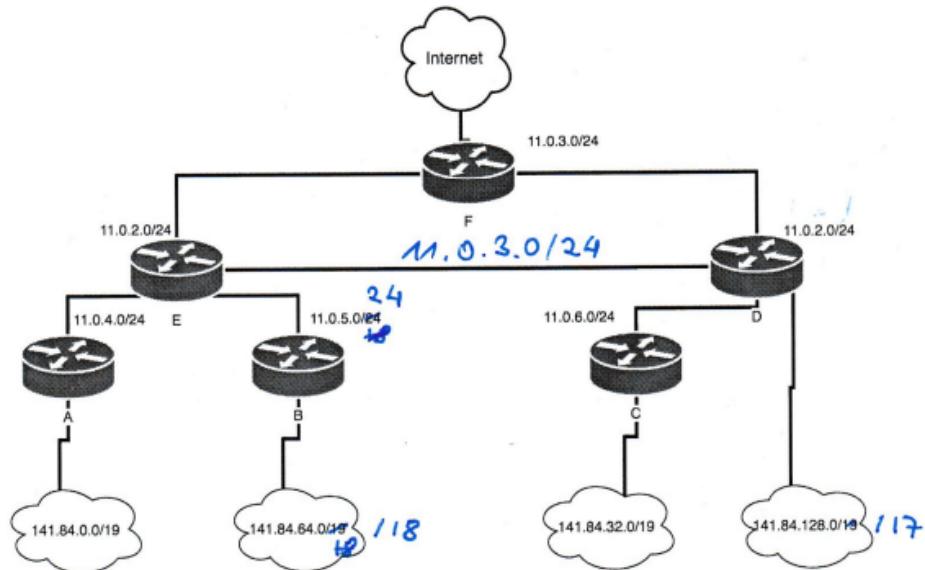
Ergänzen Sie die Skizze auf einen optimalen Quell-Senken-Baum für B!



# 2017 Klausur = 2018 Klausur

## 5 Adressierung in Rechnernetzen

Ein Unternehmen will ein strukturiertes IP-Netz aufbauen. Organisatorisch hat das Unternehmen mehrere Arbeitsgruppen, die völlig unabhängig voneinander arbeiten. Der Netzadministrator empfiehlt deshalb den Aufbau entsprechend vieler Subnetze innerhalb des Adressblocks 141.84.0.0/16.



16. Bestimmen Sie die Broadcast-Adresse des Subnetzes (141.84.32.0/19), das direkt an Router C angeschlossen ist.

141.84.63.255

17. Schreiben Sie Einträge der Routingtabelle für Router E, so dass E auf dem kürzesten Weg IP-Pakete aus dem Internet an alle Rechner im Adressbereich 141.84.0.0/16 korrekt weiterleitet! Wählen Sie für die Router passende IP-Adressen aus den angegebenen Subnetzen.

Ziel	Router
141.84.0.0/19	11.0.4.1 (Router A)
141.84.32.0/19	11.0.3.1 (Router B)
141.84.64.0/18	11.0.5.1 (Router B)
141.84.128.0/17	11.0.6.1 (Router D)

zu 16.

- Subnetzmaske aufstellen: Präfix 19, d.h. 19 vordere 1en: 1111 1111 . 1111 1111 . 1110 0000 . 0000 0000 = Subnetzmaske 255.255.224.0
- Hostanteil:** hier die letzten 13 Bits. Diese machen wir zu 1en: 141.84.32.0 = 1000 1101 . 0101 0100. 0010 0000 . 0000 0000 + 13 Bits = 1000 1101 . 0101 0100 . 0011 1111 . 1111 1111 = 141.84.63.255

18. Angenommen der Block 141.84.0.0/16 wird mit der Subnetzmaske 255.255.255.192 aufgeteilt. Wieviele Subnetze lassen sich damit maximal realisieren?

$2^{10} - 2$

19. Notieren Sie die IPv6 Adresse 1337:0000:0000:0000:1000:0000:0000:0001 maximal verkürzt, so dass keine kürzere vollständige Darstellung dieser Adresse in IPv6 existiert.

1337::1000:0:0:1

20. Ein Internetanbieter erhält das Subnetz 2001:CDE0:0000:0000:0000:0000:0000/27. Dieses wird vollständig in vier gleich große Teilbereiche geteilt.

- (a) Geben Sie die Länge der Netz-ID der entstehenden Teilnetze in Anzahl Bits an!

29

- (b) Schreiben Sie die vier entstehenden Subnetze in CIDR-Notation auf!

x [2001: CDE0:0000:0000:0000:0000:0000] /29

1. x { 2001: CDE0 :: /29

2. x 2001: CDE8 :: /29

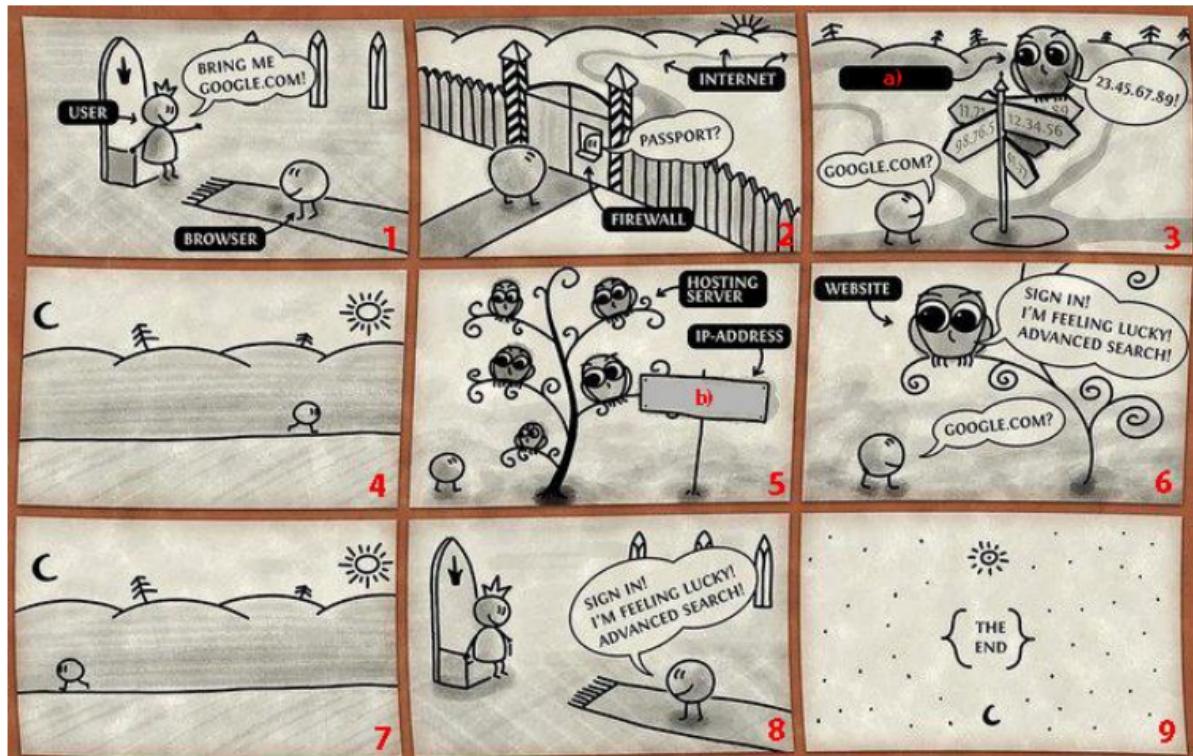
3. x 2001: CDFA :: /29

4. 2001: CDFA :: /29

# Comic: Das Internet

2015 Klausur = 2022 Klausur

## X. Comic: So funktioniert das Internet



Bearbeitet nach: [http://images.forwallpaper.com/files/thumbs/preview/111/1110896\\_how-internet-works\\_p.jpg](http://images.forwallpaper.com/files/thumbs/preview/111/1110896_how-internet-works_p.jpg)

23. Wie muss das Feld a) in Bild 3 beschriftet werden?
  24. Wie muss das Feld b) in Bild 6 beschriftet werden?
  25. Welches Protokoll sprechen Browser und Eule in Bild 3?
  26. Welches Protokoll sprechen Browser und Eule in Bild 5?
  27. Nennen Sie ein Bild, in dem zwei Instanzen der Anwendungsschicht miteinander kommunizieren.
- 
23. DNS-Server
  24. 23.45.67.89
  25. DNS
  26. TCP, Bild 6 HTTP
  27. 3, 6 (Warum nicht Bild 8? In Bild 8 gibt der Browser die Info an den User, welcher nicht Teil der Anwendungsschicht ist, zurück / weiter)

# Cyclic Redundancy Check (CRC)

2017 Klausur

## 8 Cyclic Redundancy Check

32. Gegeben sei das Generatorpolynom  $G = x^3 + 1$ .

- (a) Durch wie viele Bits wird  $G$  bei CRC repräsentiert?

4

- (b) Es soll die Nachricht 11 00 11 CRC-geschützt übertragen werden. Berechnen Sie die zu übertragende Bitfolge (inkl. CRC-Prüfsumme!) unter der Verwendung des Generatorpolynoms  $G$ .

Zu übertragende Bitfolge: 11 00 11 101

*4-1 padding*

$$\begin{array}{r} 110011\overbrace{000}^{\text{4-1 padding}} : 1001 = 110101 \\ \underline{1001} \\ 01011 \\ \underline{1001} \\ 00101 \\ \underline{00000} \\ 1010 \\ \underline{1001} \\ 001\cancel{0}10 \\ \underline{00000} \\ 1000 \\ \underline{1001} \\ 0101 \end{array}$$

Rest: 101

- (c) Nehmen Sie an, dass Sie die CRC-geschützte Bitfolge 10 01 10 01 empfangen haben. Zeigen Sie, dass die empfangene Bitfolge unter Verwendung des Generatorpolynoms  $G$  korrekt ist (inkl. Rechnung). Markieren Sie in Ihrer Rechnung die Stelle, an der der Empfänger die Korrektheit ablesen kann.

$$\begin{array}{r} 10011001 : 1001 = \dots \\ \underline{1001} \\ 00001001 \\ \underline{1001} \\ 0 \leftarrow \text{Rest } 0 \Rightarrow \text{Prüfsumme stimmt} \end{array}$$

## 2022 Übung

### 4. CRC (H)

- (a) Gegeben sei das Generatorpolynom  $G = x^3 + 1$ .
- Durch wie viele Bits wird  $G$  bei CRC repräsentiert?
  - Es soll die Nachricht 11 00 11 CRC-geschützt übertragen werden. Berechnen Sie die zu übertragende Bitfolge (inkl. CRC-Prüfsumme) unter Verwendung des Generatorpolynoms  $G$ .
  - Nehmen Sie an, dass Sie die CRC-geschützte Bitfolge 10 01 10 01 empfangen haben. Zeigen Sie, dass die empfangene Bitfolge unter Verwendung des Generatorpolynoms  $G$  korrekt ist (inkl. Rechnung). Markieren Sie in Ihrer Rechnung die Stelle, an der der Empfänger die Korrektheit ablesen kann.

i)  $\Rightarrow 4$  Bits, da der Grad von  $G$  3 ist

ii) Generatorpolynom  $G = x^3 + 1$

$$\begin{aligned} - & 1 * x^3 + 0 * x^2 + 0 * x^1 + 0 * x^0 = 1 * x^3 = 1000 \\ - & 1000 + 1 = 1001 \end{aligned}$$

$$\Rightarrow \text{Generatorpolynom} = 1001$$

$\Rightarrow$  CRC-Prüfsumme: n viele 0er, n abhängig von Grad von  $G$ : 3

Nachricht inkl. Prüfsumme: 11 00 11 000

Rechnung (XOR):

1 1 0 0 1 1 0 0 0

1 0 0 1

-----

0 1 0 1 1

1 0 0 1

-----

0 0 1 0 1 0

1 0 0 1

-----

0 0 1 1 0 0

1 0 0 1

-----

0 1 0 1

$\Rightarrow 101$ , finale Bitfolge: 11 00 11 101

**MERKE:**  
bei XOR  
1 1 = 0  
1 0 = 1  
0 1 = 1  
0 0 = 0

wir fangen ab da an wo es eine 1 gibt, Achtung wenn am Anfang und nach rechts weniger als 4 Ziffern da sind, dann muss von oben ein runtergezogen werden

## Domain Name System

### 2014 Klausur

Die Konsolenausgabe ist nicht gleich der 2017er Klausur.

43. Ein A Resource Record (RR-Typ) repräsentiert die Abbildung eines Hostnamen (RR-Name) auf eine IPv4-Adresse (RR-Wert):  $f_A(\text{Hostname}) : \text{Hostname} \rightarrow \text{IPv4-Adresse}$ . Notieren Sie im Folgenden die Abbildungen der angegebenen RR-Typen in der Form  $f_{RR-\text{Typ}}(\text{RR-Name}) : \text{RR-Name} \rightarrow \text{RR-Wert}$ , analog zum A Beispiel!

(a) NS: \_\_\_\_\_ (1)

(b) CNAME: \_\_\_\_\_ (1)

(c) MX: \_\_\_\_\_ (1)

44. Ist es möglich auf Basis der oben stehenden Informationen der Konsolenausgabe eine E-Mail an postmaster@ifi.lmu.de erfolgreich auszuliefern? Begründen Sie kurz. (2)

## 2017 Klausur = 2018 Klausur

### 4 Domain Name System (6 Punkte)

Gegeben sei folgende Konsolenausgabe einer DNS-Anfrage:

```
; global options: +cmd
.
      151595    IN    NS    d.root-servers.net.
.
      151595    IN    NS    b.root-servers.net.
.
      151595    IN    NS    a.root-servers.net.
.
      151595    IN    NS    k.root-servers.net.
;; Received 449 bytes from 85.214.7.22#53(85.214.7.22) in 296 ms

de.          172800    IN    NS    c.de.net.
de.          172800    IN    NS    z.nic.de.
de.          172800    IN    NS    a.nic.de.
de.          172800    IN    NS    s.de.net.
de.          172800    IN    NS    l.de.net.
de.          172800    IN    NS    f.nic.de.
;; Received 290 bytes from 128.8.10.80#53(d.root-servers.net) in 108 ms

lmu.de.      86400     IN    NS    dns1.lrz-muenchen.de.
lmu.de.      86400     IN    NS    dns2.lrz-muenchen.de.
lmu.de.      86400     IN    NS    dns3.lrz-muenchen.de.
;; Received 206 bytes from 208.48.81.43#53(c.de.net) in 185 ms

www.ifi.lmu.de 86400     IN    CNAME  salerno.tcs.ifi.lmu.de
salerno.tcs.ifi.lmu.de 86400     IN    A     141.84.94.49
tcs.ifi.lmu.de   86400     IN    NS    kokytos.rz.informatik.uni-muenchen.de
tcs.ifi.lmu.de   86400     IN    NS    acheron.informatik.uni-muenchen.de
;; Received 177 bytes from 129.187.5.2#53(dns3.lrz-muenchen.de) in 23 ms
```

14. (a) Wie viele Anfragen zur oben gezeigten DNS-Anfrage waren nötig, um den Hostnamen `www.ifi.lmu.de` aufzulösen?

4

---

- (b) Die URL `http://www.ifi.lmu.de/` soll in einem Web-Browser angezeigt werden. Geben sie die IPv4-Adresse des Rechners an, an den die HTTP-Anfrage gestellt wird!

141.84.94.49

---

- (c) Ist die DNS-Anfrage rekursiv oder iterativ? Begründen Sie Ihre Antwort!

Iterativ, sonst würde man nicht H DNS  
Requests sehen können.

---

15. Welches Transportprotokoll wird auf Schicht IV des ISO/OSI-Referenzmodells bei DNS ...

- (a) ... für Zonentransfers eingesetzt?

TCP

---

- (b) ... für DNS-Anfragen empfohlen?

UDP

---

# 2022 Übung

## 1. Interpretation einer DNS-Antwort (H)

Ein nützliches Diagnosewerkzeug für den DNS ist das Programm `dig` (1), das auf vielen Unix-Derivaten (z.B. GNU/Linux Installationen) vorhanden ist. Nachfolgend sehen Sie die aus einer Anfrage resultierenden Resource Records. Beziehen Sie sich beim Bearbeiten der Aufgabe auf die relevanten Zeilnummern!

```
bash$ dig +trace mail.nm.ifi.lmu.de

1 ; <>> DiG 9.2.3 <>> +trace mail.nm.ifi.lmu.de
2 ;; global options: printcmd
3 .          80298  IN      NS      d.root-servers.net.
4 .          80298  IN      NS      e.root-servers.net.
5 .          80298  IN      NS      f.root-servers.net.
6 .          80298  IN      NS      j.root-servers.net.
7 .          80298  IN      NS      g.root-servers.net.
8 .          80298  IN      NS      h.root-servers.net.
9 .          80298  IN      NS      b.root-servers.net.
10 .         80298  IN      NS      l.root-servers.net.
11 .         80298  IN      NS      i.root-servers.net.
12 .         80298  IN      NS      c.root-servers.net.
13 .         80298  IN      NS      m.root-servers.net.
14 .         80298  IN      NS      a.root-servers.net.
15 .         80298  IN      NS      k.root-servers.net.
16 ;; Received 500 bytes from 192.168.218.30#53(192.168.218.30) in 0 ms
17
18 de.          172800  IN      NS      C.DE.NET.
19 de.          172800  IN      NS      L.DE.NET.
20 de.          172800  IN      NS      F.NIC.de.
21 de.          172800  IN      NS      S.DE.NET.
22 de.          172800  IN      NS      A.NIC.de.
23 de.          172800  IN      NS      Z.NIC.de.
24 ;; Received 294 bytes from 128.8.10.90#53(d.root-servers.net) in 104 ms
25
26 lmu.de.       86400   IN      NS      dns3.lrz-muenchen.de.
27 lmu.de.       86400   IN      NS      dns1.lrz-muenchen.de.
28 lmu.de.       86400   IN      NS      dns2.lrz-muenchen.de.
29 ;; Received 210 bytes from 208.48.81.43#53(C.DE.NET) in 200 ms
30
31 mail.nm.ifi.lmu.de. 86400   IN      CNAME   pcheger0.nm.ifi.lmu.de.
32 pcheger0.nm.ifi.lmu.de. 86400   IN      A       141.84.218.30
33 nm.ifi.lmu.de.       86400   IN      NS      acheron.ifi.lmu.de.
34 nm.ifi.lmu.de.       86400   IN      NS      dns3.lrz-muenchen.de.
35 nm.ifi.lmu.de.       86400   IN      NS      dns1.nm.ifi.lmu.de.
36 nm.ifi.lmu.de.       86400   IN      NS      dns1.lrz-muenchen.de.
37 nm.ifi.lmu.de.       86400   IN      NS      dns2.lrz-muenchen.de.
38 nm.ifi.lmu.de.       86400   IN      NS      dns0.nm.ifi.lmu.de.
39 ;; Received 357 bytes from 129.187.5.2#53(dns3.lrz-muenchen.de) in 1 ms
```

(a) Zeichnen Sie eine Skizze, die den DNS-Verkehr zur Anfrage darstellt, mit mindestens:

- dem anfragenden Host
- dem für diesen Host zuständigen DNS-Server (lokaler DNS-Server)
- dem DNS-Server, der die richtige IP-Adresse für `mail.nm.ifi.lmu.de` liefert
- eventuellen weiteren DNS-Servern, die Teile der Antwort liefern.
- den Nachrichten, die ausgetauscht wurden.

Geben Sie bei jedem Host in Ihrer Skizze, falls vorhanden, IP-Adresse und Hostname an.

- (b) Ist die Anfrage rekursiv oder iterativ?
- (c) Die Ausgabe enthält eine Anfrage an einen der DNS-Root-Server. Wonach wird er gefragt?
- (d) Der gesuchte Rechnername `mail.nm.ifi.lmu.de` ist ein Alias.
  - i. Wie heisst die Maschine wirklich?
  - ii. Welche IP-Adresse hat sie?
- (e) Anhand der Ausgabe können weitere Aussagen bezüglich der DNS-Server gemacht werden.
  - i. Wer betreibt die DNS-Server, die für Anfragen über die Domäne `lmu.de` zuständig sind?
  - ii. Welche DNS-Server können Anfragen für die Domäne der gesuchten Maschine liefern?
  - iii. Wurde die gesuchte IP-Adresse von einem autoritativen Server geliefert?
- (f) Angenommen Sie haben als Administrator Zugriff auf den DNS-Cache der lokalen DNS-Server im LRZ. Gibt es für Sie damit eine Möglichkeit, die von Nutzern meist besuchten Web-Server im Internet ausfindig zu machen Fassen Sie sich kurz.

## Aufgabe 1: Interpretation einer DNS-Antwort

a)

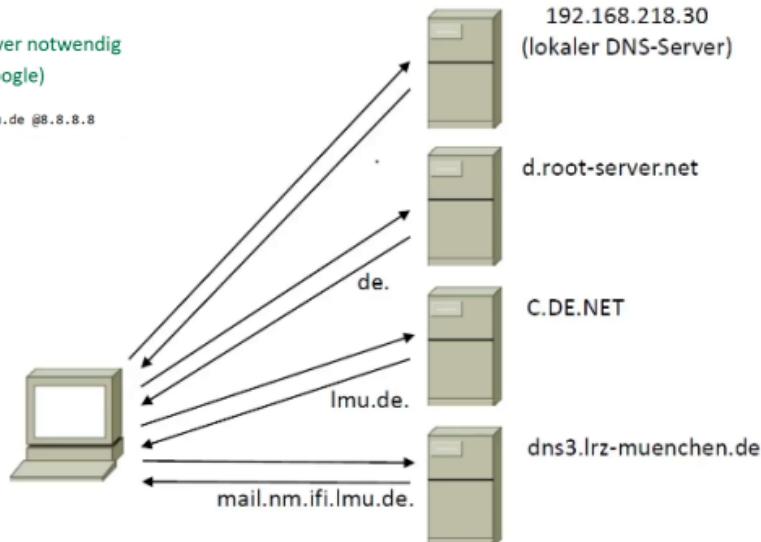
Anderer Befehl notwendig, um die Ausgabe vom Übungsblatt zu reproduzieren:

Zusätzlich +nodnssec notwendig

Bei manchen Systemen expliziter DNS-Server notwendig

Zum Beispiel @8.8.8.8 (DNS-Server von Google)

```
bash$ dig +trace +nodnssec mail.nm.ifi.lmu.de @8.8.8.8
```



b)

Interativ, nachdem wir alle Zwischenserver sehen können.

c)

Antwort des Root-Servers:

```
18     de.          172800 IN      NS      C.DE.NET.  
19     de.          172800 IN      NS      L.DE.NET.  
20     de.          172800 IN      NS      F.NIC.de.  
21     de.          172800 IN      NS      S.NIC.de.  
22     de.          172800 IN      NS      A.NIC.de.  
23     de.          172800 IN      NS      Z.NIC.de.  
24 ; Received 294 bytes from 128.8.10.90#53(d.root-servers.net) in 104 ms
```

Antwort ist aber nicht unbedingt identisch zur Anfrage

Anfrage: mail.nm.ifi.lmu.de

Root-Server kann nur auf de. antworten.

```
bash$ dig mail.nm.ifi.lmu.de @d.root-servers.net  
  
; <>> DIG 9.16.1-Ubuntu <>> mail.nm.ifi.lmu.de @d.root-servers.net  
; global options: +cmd  
; Got answer:  
; ->>HEADER<- opcode: QUERY, status: NOERROR, id: 5153  
; flags: qr rd; QUERY: 1, ANSWER: 0, AUTHORITY: 6, ADDITIONAL: 13  
; WARNING: recursion requested but not available  
  
; OPT PSEUDOSECTION:  
; EDNS: version: 0, flags:; udp: 1450  
; QUESTION SECTION:  
;mail.nm.ifi.lmu.de.      IN      A  
  
; AUTHORITY SECTION:  
de.          172800 IN      NS      a.nic.de.  
de.          172800 IN      NS      f.nic.de.  
de.          172800 IN      NS      l.de.net.  
de.          172800 IN      NS      n.de.net.  
de.          172800 IN      NS      s.de.net.  
de.          172800 IN      NS      z.nic.de.  
  
; ADDITIONAL SECTION:  
a.nic.de.    172800 IN      AAAA    2001:678:2::53  
f.nic.de.    172800 IN      AAAA    2a02:568:02::53  
l.de.net.    172800 IN      AAAA    2001:668:1f:11::105  
n.de.net.    172800 IN      AAAA    2001:67c:1011:1::53  
s.de.net.    172800 IN      AAAA    2003:8:14::53  
z.nic.de.    172800 IN      AAAA    2a02:568:fe02::de  
a.nic.de.    172800 IN      A      194.0.0.53  
f.nic.de.    172800 IN      A      81.91.164.5  
l.de.net.    172800 IN      A      77.67.63.105  
n.de.net.    172800 IN      A      194.146.107.6  
s.de.net.    172800 IN      A      195.243.137.26  
z.nic.de.    172800 IN      A      194.246.96.1
```

d)

i) pcheger0.nm.ifi.lmu.de

ii) 141.84.218.30

```
31 mail.nm.ifi.lmu.de. 86400 IN CNAME pcheger0.nm.ifi.lmu.de.  
32 pcheger0.nm.ifi.lmu.de. 86400 IN A 141.84.218.30
```

e)

i) Das LRZ (Leibniz-Rechenzentrum)

```
26 lmu.de. 86400 IN NS dns3.lrz-muenchen.de.  
27 lmu.de. 86400 IN NS dns1.lrz-muenchen.de.  
28 lmu.de. 86400 IN NS dns2.lrz-muenchen.de.
```

ii) Alle in der Liste.

```
31 mail.nm.ifi.lmu.de. 86400 IN CNAME pcheger0.nm.ifi.lmu.de.  
32 pcheger0.nm.ifi.lmu.de. 86400 IN A 141.84.218.30  
33 nm.ifi.lmu.de. 86400 IN NS acheron.ifi.lmu.de.  
34 nm.ifi.lmu.de. 86400 IN NS dns3.lrz-muenchen.de.  
35 nm.ifi.lmu.de. 86400 IN NS dns1.nm.ifi.lmu.de.  
36 nm.ifi.lmu.de. 86400 IN NS dns1.lrz-muenchen.de.  
37 nm.ifi.lmu.de. 86400 IN NS dns2.lrz-muenchen.de.  
38 nm.ifi.lmu.de. 86400 IN NS dns0.nm.ifi.lmu.de.  
39 ;; Received 357 bytes from 129.187.5.2#53(dns3.lrz-muenchen.de) in 1 ms
```

iii) Ja, dns3.lrz-muenchen.de ist autoritativ für mail.nm.ifi.lmu.de

```
31 mail.nm.ifi.lmu.de. 86400 IN CNAME pcheger0.nm.ifi.lmu.de.  
32 pcheger0.nm.ifi.lmu.de. 86400 IN A 141.84.218.30  
33 nm.ifi.lmu.de. 86400 IN NS acheron.ifi.lmu.de.  
34 nm.ifi.lmu.de. 86400 IN NS dns3.lrz-muenchen.de.  
35 nm.ifi.lmu.de. 86400 IN NS dns1.nm.ifi.lmu.de.  
36 nm.ifi.lmu.de. 86400 IN NS dns1.lrz-muenchen.de.  
37 nm.ifi.lmu.de. 86400 IN NS dns2.lrz-muenchen.de.  
38 nm.ifi.lmu.de. 86400 IN NS dns0.nm.ifi.lmu.de.  
39 ;; Received 357 bytes from 129.187.5.2#53(dns3.lrz-muenchen.de) in 1 ms
```

f)

Der Cache gibt keine Auskunft über Häufigkeit

Wiederholtes Abfragen des Caches

> Adressen, die immer wieder enthalten sind

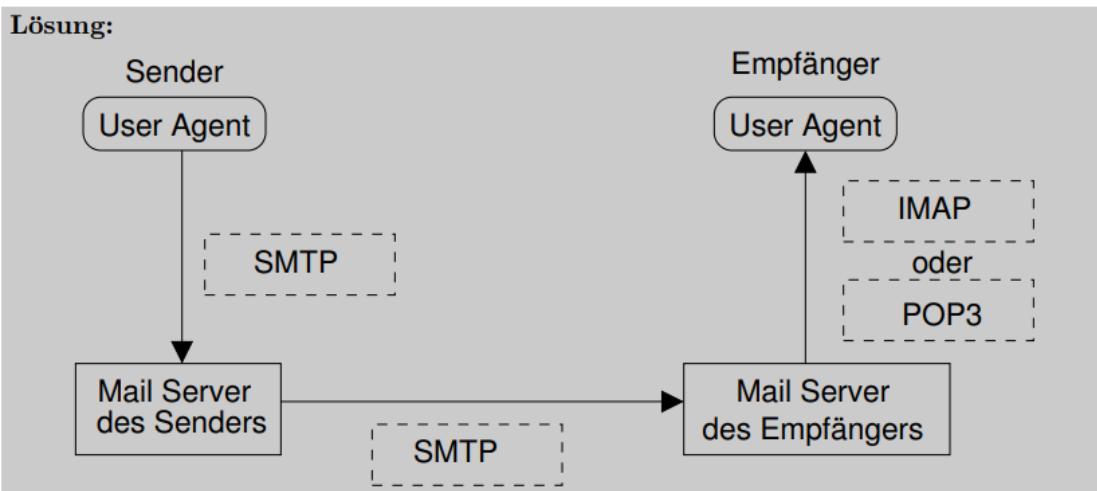
Immer noch keine Häufigkeit, aber Regelmäßigkeit

# E-Mail

## 2009 Klausur = 2014 Probeklausur

### 7. E-Mail

- (a) Beschriften Sie alle Pfeile in der Zeichnung mit den entsprechenden E-Mail Protokollen.



- (b) Internet E-Mail ist empfindlich gegen den Dienstgüteparameter ‘‘Datenverlust’’ des Transportnetzes. Nennen Sie zwei Dienstgüteparameter, gegen die E-Mail **unempfindlich** ist und begründen Sie.

**Lösung:**

Zum Beispiel:

- **Latenz**, da kein Echtzeitdienst
- **Jitter**, da kein Echtzeitdienst

## 2014 Klausur

X. Elektronisches Post System  
Gegeben sind die vollständigen Daten, die ein SMTP-Server während einer SMTP-Sitzung empfängt:

1 Hello nm.ifi.lmu.de  
 2 Mail From: <root@nm.ifi.lmu.de>  
 3 Rcpt To: <manager@nm.ifi.lmu.de>  
 4 Data  
 5 Reply-To: <operators@nm.ifi.lmu.de>  
 6 From: <admin@nm.ifi.lmu.de>  
 7 To: <manager@nm.ifi.lmu.de>  
 8 Subject: Was ist das  
 9  
 10 was das ist?  
 11  
 12 .  
 13 Quit

→ Diese  zum Markieren von Zeilen benutzen!

34. Markieren Sie genau alle Zeilen, die SMTP-Protokollsteuerinformationen enthalten!  
*Hinweis: Es zählt genau die Markierung in den vorgegebenen Bereichen!*

35. An wen soll ggf. eine Antwort adressiert werden?

---

36. Wie lautet die Kennung des Empfängers?

---

37. Nennen Sie den Unterschied in der E-Mailverwaltung, der mit dem Einsatz von IMAP statt POP3 einher geht!

---

---

38. Welches Protokoll verwendet ein User Agent (UA) zum Versenden von E-Mails?

---

39. Welche Rolle in einem Message Handling System (MHS) übernimmt Webmail?

---

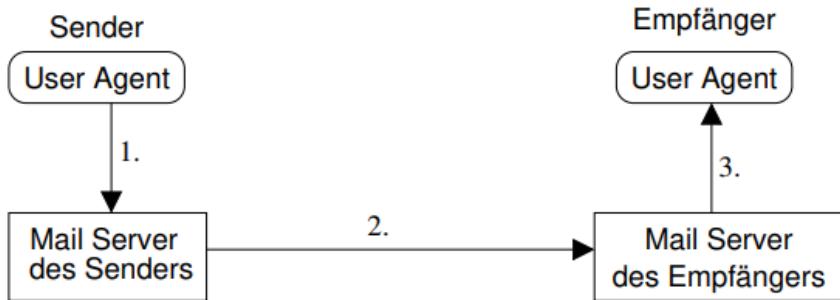
40. Wie nennt man bei E-Mail den Mechanismus, mit dem z.B. auch das Versenden von nicht ASCII Zeichen ermöglicht wird?

---

Wird E-Mail durch den Einsatz von TCP unempfindlich gegenüber dem Dienstgütemerkmal "Datenverlust"? Begründen Sie Ihre Antwort!

## 2022 Übung

### 4. Email (H)



- Die Abbildung skizziert den Weg einer Email vom Verfasser zum Adressaten. Welche Protokolle der Anwendungsschicht können auf den drei eingezeichneten Übertragungswegen eingesetzt werden?
- Nehmen Sie nun an, dass der Sender mit einem webbasierten E-Mail Account (bspw. GMail oder GMX) eine E-Mail an den Empfänger verschickt. Welche (zusätzlichen) Protokolle im Vergleich zu Teilaufgabe (a) sind involviert?
- Welche dargestellten Systeme sind Teil des *Message Transfer Systems*?
- Internet E-Mail ist empfindlich gegen den Dienstgüteparameter "Datenverlust" des Transportnetzes. Gibt es allgemeine Dienstgüteparameter, gegen die E-Mail unempfindlich ist? Begründen Sie Ihre Antwort!

#### Aufgabe 4: Email

1. SMTP  
2. SMTP  
3. IMAP und POP3
1. HTTP bzw. bei verschlüsselter Verbindung HTTP und TLS (SSL)
- c)  
Nur die beiden Mail Server  
User Agents bilden den Abschluss des Message Handling System.  
Das Transfer System vermittelt die Nachricht.
- d)  
Latenz: E-Mail ist kein Echtzeitdienst. (Verzögerung)  
Übertragungsrate: Langsame Übertragung verschlechtert den Dienst nicht.

# Ethernet, CSMA

## 2009 Klausur = 2014 Probeklausur

### 4. Ethernet, CSMA

Gegeben sei ein Ethernet mit einer Übertragungsrate von 1 GBit/s, einer Leitungslänge von 1000 m und einer Signalgeschwindigkeit von  $2 \cdot 10^8 \text{ m/s}$ . Berechnen Sie die minimale Rahmengröße, bei der CSMA/CD als Vielfachzugriffsverfahren noch einsetzbar wäre. Geben Sie das Ergebnis in Bytes (Oktetten), sowie den Rechenweg an!

*Hinweis:* 1 GBit =  $10^9$  Bits

#### Lösung:

RTD des Signals:

$$T_{rtt\text{-}signal} = \frac{1\text{km}}{200.000\text{km/s}} \times 2 = 10^{-5}\text{s}$$

Minimale Framegröße beträgt:

$$S_{frame\text{-}size} = 10^{-5}\text{s} \times 10^9 \text{ bit/s} = 10^4 \text{ bit} = 1250 \text{ byte}$$

## 2015 Klausur

### V. Ethernet, CSMA

11. Gegeben sei ein Ethernet mit einer Übertragungsrate von 10 GBit/s, einer Leitungslänge von 2000m und einer Signalgeschwindigkeit von  $2 \cdot 10^8 \text{ m/s}$ . Berechnen Sie die minimale Rahmengröße, bei der CSMA/CD als Vielfachzugriffsverfahren noch einsetzbar wäre. Geben Sie das Ergebnis in Bytes (Oktetten), sowie den Rechenweg an!

*Hinweis:* 1 GBit =  $10^9$  Bits

## 2018 Klausur = 2019 Klausur

### VIII. CSMA und Ethernets

22. Erläutern Sie kurz den Unterschied zwischen CSMA/CD und Slotted Aloha.

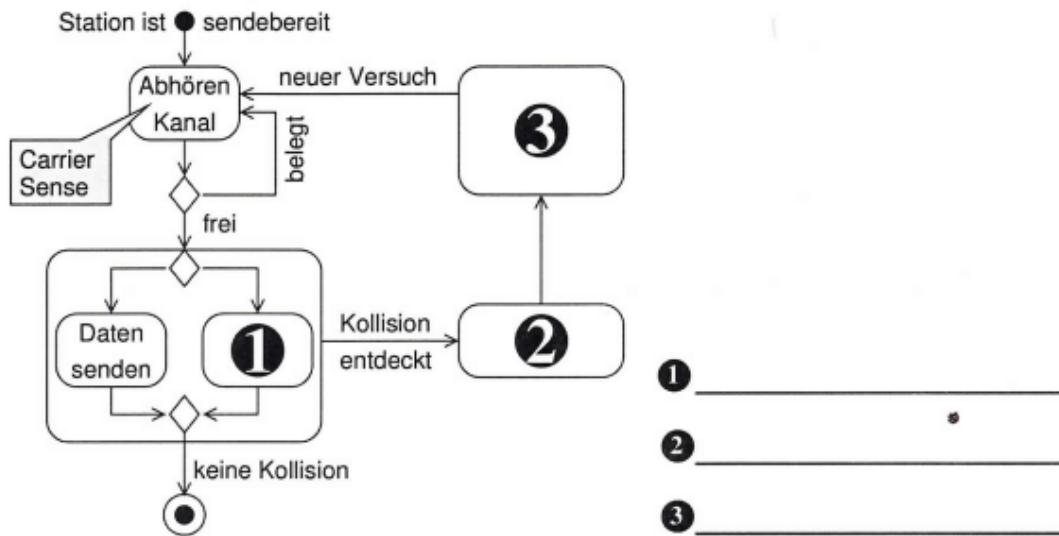
(2)

---

---

23. Vervollständigen Sie das CSMA/CD Ablaufdiagramm.

(3)



zu 22.

### Slotted Aloha: Protokollablauf

Sei  $p$  eine Wahrscheinlichkeit,  $p \in [0,1]$

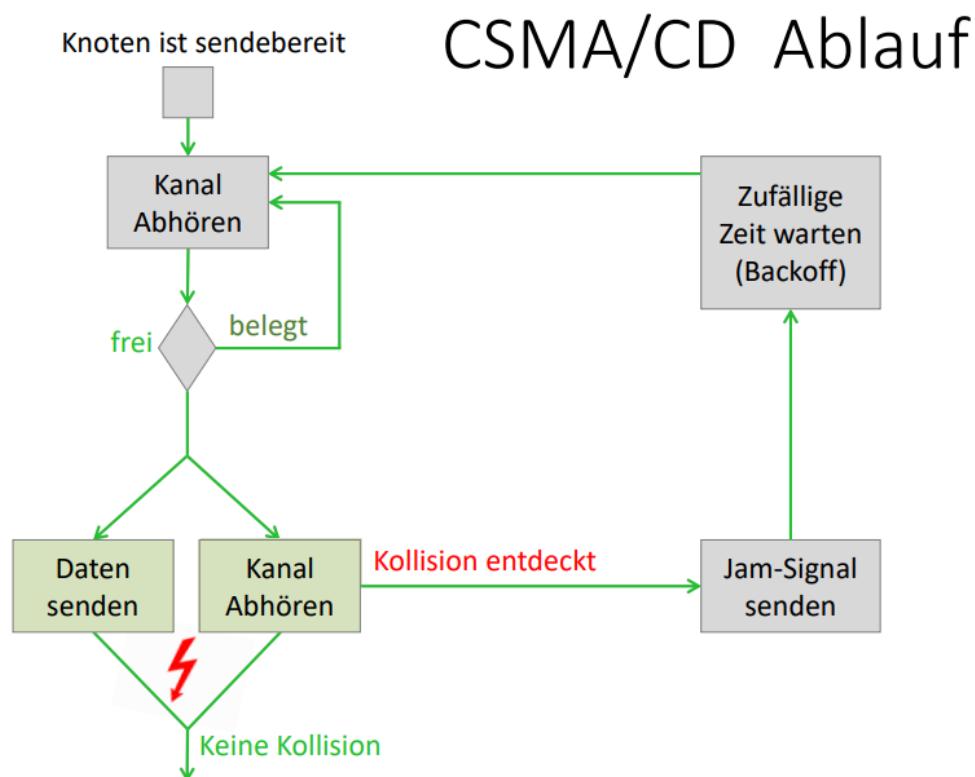
- Tritt keine Kollision auf, ist der Rahmen erfolgreich übertragen.
- Bei Auftreten einer Kollision: Erkennen vor Ende des *aktuellen* Sende-Intervalls. Erneute Übertragung im *nächsten* Intervall mit Wahrscheinlichkeit  $p$ .  
→ Nicht-Übertragung sowie erneutes Versuchen in späteren Zeitintervallen daher mit Wahrscheinlichkeit  $(1 - p)$ .

zu CSMA/CD

- Grundsätzliche Regeln (im Gegensatz zu Aloha)

- **Carrier Sensing:** Bevor die Übertragung eines Rahmens initiiert wird, Kanal abhören. Falls Kanal „belegt“, warten bis Kanal frei ist.
- Kollisionserkennung (**Collision Detection**): Tritt während der Übertragung eine Kollision auf, Übertragung abbrechen und zufällige Zeit warten bevor ein erneuter Versuch gestartet wird.

zu 23.



24. Sie wollen einen Rahmen mit einer Gesamtlänge von 512 Byte übertragen. Die effektive Übertragungsrate beträgt  $10^7 \text{ bits/sec}$ . Die Ausbreitungsgeschwindigkeit des Signals im Medium beträgt  $2 * 10^8 \text{ m/sec}$ .

- (a) Wie groß ist die maximale Leitungslänge, auf der Sie eine Kollision unter Verwendung von CSMA/CD als Medienzugriffsverfahren zuverlässig erkennen können (inkl. Rechnung)? (3)

Gegeben sind:

$$\text{Rahmen Gesamtlänge } g = 512 \text{ byte} = 8 * 2^9 \text{ bit} = 2^3 * 2^9 \text{ bit}$$

$$\text{Effektive Übertragungsrate } r = 10^7 \text{ bit/s}$$

$$\text{Ausbreitungsgeschwindigkeit des Signals } s = 2 * 10^8 \text{ m/s}$$

Gesucht ist maximale Leitungslänge  $l$ :

$$g = t * r \Rightarrow t = \frac{g}{r} = \frac{2^3 * 2^9 \text{ bit}}{10^7 \text{ bit/s}} = \frac{2^3 * 2^9}{5^7 * 2^7} \text{ s} = \frac{2^3 * 2^2}{5^7} \text{ s} = \frac{2^5}{5^7} \text{ s}$$

$$t = \frac{2l}{s} \Rightarrow l = \frac{ts}{2} = \frac{\frac{2^5}{5^7} \text{ s} * 2 * 10^8 \text{ m/s}}{2} = \frac{2^5}{5^7} \text{ s} * 10^8 \text{ m/s} = \frac{2^5 * 10^8}{5^7} \text{ m} = \frac{2^5 * 2^8 * 5^8}{5^7} \text{ m} = 2^{13} * 5 \text{ m} \\ = 8.192 * 5 \text{ m} = 40.960 \text{ m}$$

- (b) Wie verändert sich die maximal mögliche Leitungslänge zur Kollisionserkennung, wenn die effektive Übertragungsrate verringert wird? (1)

Wird die Übertragungsrate verringert, so erhöht sich die maximal mögliche Leitungslänge.

## Fehlererkennung bei UDP

### 2019 Klausur

#### IX. Fehlererkennung bei UDP

Ein typischer Fehler bei der Übertragung von Daten ist die Verfälschung, bei der Stellen des Bitstroms invertiert werden. In dieser Aufgabe soll die Nachricht RNVS übertragen werden.

Nachfolgende Tabelle beinhaltet die zu den Buchstaben entsprechende ASCII Codierung:

Buchstabe	ASCII
R	1010010
N	1001110
V	1010110
S	1000011

25. Die Nachricht soll über das UDP Transportprotokol, das die 16-bit Internet Checksumme verwendet, versandt werden.

- (a) Teilen Sie die Nachricht dazu in zwei 16-bit Segmente auf und berechnen Sie anschließend die Internet-Checksumme nach Vorschrift. *Hinweis:* Führende Nullen sind notwendig, damit 8 Bit pro Buchstaben zur Codierung verwendet werden. (3)

Checksumme: \_\_\_\_\_

Rechnung: \_\_\_\_\_

$$R+N = 10100000$$

$$\rightarrow \text{Einerkomplement: } 01011111$$

$$V+S = 10011001$$

-> Einerkomplement: 01100110

Checksumme: 01011111 01100110

- (b) Welche Schritte muss der Empfänger ausführen, um die Nachricht als korrekt zu verifizieren? (2)

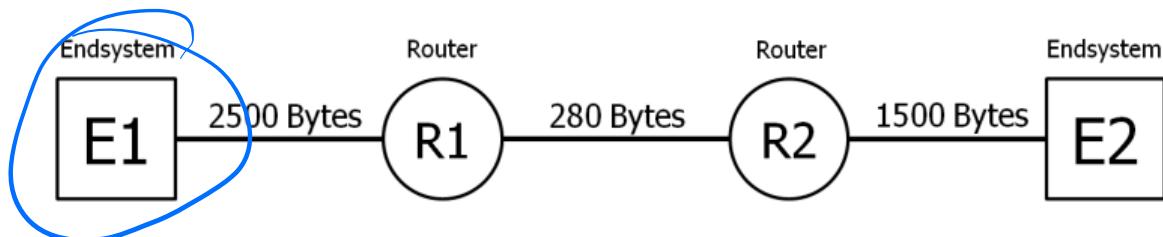
Die 16-bit Segmente und Checksumme summieren

Wenn Ergebnis nur 1en enthält -> korrekt

## Fragmentierung

### 2015 Klausur

8. Betrachten Sie ein Netz bestehend aus den Endsystemen E1 und E2, sowie den Routern R1 und R2. Die Leitungen sind mit ihren maximum transfer units beschriftet.



- a. Nennen Sie die Komponente(n), die ein 1000 Byte langes IPv6-Paket von E1 an E2 fragmentieren!
- b. E1 möchte ein IPv6 Paket mit 600 Bytes Nutzdaten an E2 senden.

Header (40 Byte)	Nutzdaten (600 Byte)
------------------	----------------------

Zeichnen Sie die Fragmente so wie E2 sie empfängt. Geben Sie dabei Kopf- und Nutzdatenlänge analog zur Darstellung des ursprünglichen Pakets an! Hinweis: Ein IPv6 Header inklusive Fragmentation-Header ist 48 Byte lang.

Fragmentnummer	Fragment
1	M=1   F0=0   232
2	M=1   F0=29   232
3	M=0   F0=58   736

HEADER                            NUTZDATEN  
48 Byte

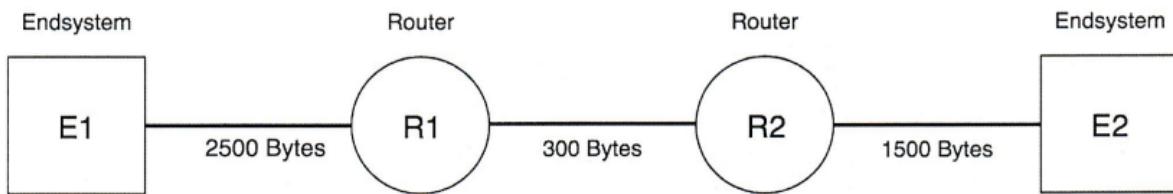
8.

- a. Bei IPv6 darf nur der Absender das Paket fragmentieren, daher E1.

b. IPv6 Header: 40 Byte, Fragmentation Header: 8 Byte. (Bei IPv4 ist der Fragmentation Header mit eingebaut) -> Headerlänge 48 Byte

2017 Klausur = 2018 Klausur

## 6 Fragmentierung



Betrachten Sie ein Netz bestehend aus zwei Endsystemen und zwei Routern, die mittels dreier Kanäle A,B,C verbunden sind mit Maximum Transfer Units (MTU, in Bytes) wie folgt:

Kanal A: 2500 Bytes    Kanal B: 300 Bytes    Kanal C: 1500 Bytes

Das Netz ist so konfiguriert, dass Fragmentierung in der Vermittlungsschicht durchgeführt wird. E1 schickt ein IPv4-Paket, das an E2 adressiert ist. (Hinweis: Byte = Oktett)

21. Nennen Sie die Komponenten, die das IP-Paket fragmentieren, wenn die Gesamtlänge

(a) 200 Byte beträgt.

Keine - (Nein )

(b) 1000 Byte beträgt.

R1

(c) 2000 Byte beträgt.

R1    ~~2000~~

22. Angenommen, R1 soll folgendes IPv4-Paket an R2 weiterleiten:

Header(20 Byte)	Nutzdaten(600 Byte)
-----------------	---------------------

(a) Zeichnen Sie die Fragmente unter Angabe von Kopf- und Nutzdatenlänge (wie in der Aufgabenstellung), in der Reihenfolge, in der sie von R1 versendet werden!

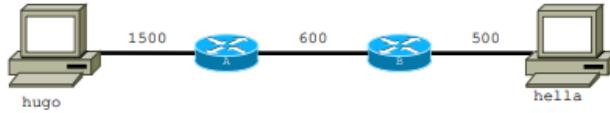
Fragmentnummer	Header-Länge	Nutzdatenlänge
1	20	280 ← muss teildar & sein
2	20	280
3	20	40

(b) Wie erkennt R2 beim ersten Fragment, dass es sich um ein Fragment handelt (und nicht um ein „vollständiges“ IPv4-Paket)?

gesetztes „more“

# 2022 Übung

## 2. Fragmentierung (H)



Der Rechner **hugo** möchte Daten an den Rechner **hella** übertragen. Die Abbildung zeigt die beiden Rechner und dazwischen befindliche Router, sowie Leitungen, die mit ihrer MTU beschriftet sind. Die MTU gibt die maximale Größe von IP-Paketen an, die auf dieser Leitung übertragen werden kann. Ein verbindungsloses Protokoll der OSI-Schicht 4 ohne Bestätigungen (z.B. UDP) übergibt Segmente an Schicht 3. Die Daten sollen per IPv4 an **hella** vermittelt werden. Insgesamt werden 8000 Bytes an IPv4-Nutzdaten an **hella** übertragen. Auf Schicht 2 wird Ethernet mit 1Gbps Übertragungsrate eingesetzt.

Beachten Sie, dass IPv4 nur an den Grenzen von 8 Byte Blöcken fragmentieren kann. Der Offset gibt an wie viele dieser 8 Byte Blöcke vor dem aktuellen Fragment fehlen.

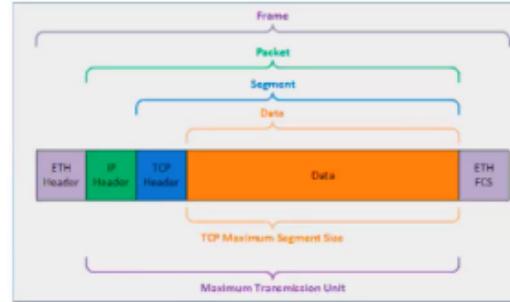
Die Ausbreitungsgeschwindigkeit von Signalen in den Leitungen ist  $2 \cdot 10^8 \frac{m}{s}$ . Vernachlässigen Sie die Verarbeitungsverzögerungen und den Overhead von Schicht 2.

- (a) Bestimmen Sie die größtmögliche Segmentlänge, die **hugo** mit einer IPv4-Nachricht versenden kann!
- (b) Bei der Vermittlung der Daten zu **hella** tritt Fragmentierung auf. Wieviele IPv4-Fragmente empfängt **hella** mindestens, bis 5000 Bytes Nutzdaten empfangen wurden? *Hinweis:* **hugo** verschickt pro Rahmen maximal viele Nutzdaten.
- (c) Erstellen Sie eine Tabelle die in chronologischer Reihenfolge, die Länge jedes IPv4-Pakets in Bytes, gesetzte Header-Flags und das Fragment Offset der von **hella** empfangenen IPv4-Nachrichten zeigt!
- (d) Berechnen Sie die Zeit, die ein 1000 Bytes langes Segment von **hugo** nach **hella** benötigt, auf die Nanosekunde genau! *Hinweis:*  $1\text{Gbps} = 10^9 \frac{\text{Bit}}{\text{s}}$ .
- (e) **hugo** beginnt nun mit der Übertragung der 8000 Bytes an **hella**. Berechnen Sie die Zeit, die zwischen dem Versenden des ersten Rahmens durch **hugo** und dem vollständigen Versenden des letzten Rahmens durch Router A verstreicht, auf die Nanosekunde genau!
- (f) Nun wird an Stelle von IPv4 das neuere IPv6 eingesetzt und der Versuch, bei dem **hugo** 8000 Bytes Nutzdaten an **hella** überträgt wiederholt. Die Verarbeitungsverzögerung verändert sich durch den Austausch des Schicht 3 Protokolls nicht.
  - i. Informieren Sie sich zu nächst in RFC 2463 (<http://www.faqs.org/rfcs/rfc2463.html>) über ICMPv6 Fehlernachrichten und die „Packet too big“ Fehlernachricht. Wieviele „Packet too big“ Nachrichten wird **hugo** empfangen und wie wird der Rechner darauf reagieren? Begründen Sie Ihre Antwort!
  - ii. Erstellen Sie analog zu Teilaufgabe c eine Tabelle, für den Fall, dass für die Übermittlung IPv6 zum Einsatz kommt!

## Aufgabe 2: Fragmentierung

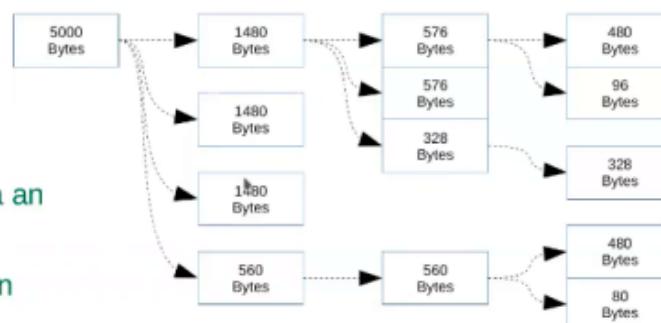
a)

- IPv4 Header = 20 Bytes
- TCP Header = min. 20 Bytes
- Maximum Transmission Unit (MTU):
  - Wieviele Bytes können in einem Ethernet Rahmen übertragen werden?
- MTU zwischen hugo und Router A beträgt 1500
  - **TCP Segment ist max. 1480 Bytes lang**
  - Aber:** MSS = max. 1460 Bytes



b)

- 5000 Bytes Nutzdaten von hugo
  - 3x 1480 Bytes + 560 Bytes (Rest)
- MTU zwischen Router A und Router B beträgt 600
  - max. 580 Bytes Nutzdaten in einem IPv4 Paket
- **Aber!** IPv4/IPv6 fragmentiert an 64Bit (8Byte) Grenzen
  - max. 580 Bytes – 580%8 Bytes = 576 Bytes Nutzdaten pro Paket
- MTU zwischen Router A und Router B beträgt 500
  - max. 480 Bytes Nutzdaten pro Paket (480%8 = 0)
- 3x 1480 Bytes kommen in je 3 Fragmenten bei B an
- 560 Bytes kommen in 1 Fragment bei B an
  - Bei hella:
- 3x 1480 Bytes kommen in je 5 Fragmenten bei hella an
- 560 Bytes kommen in 2 Fragmenten bei hella an
  - **Insgesamt 17 Fragmente**



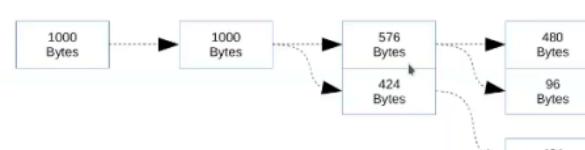
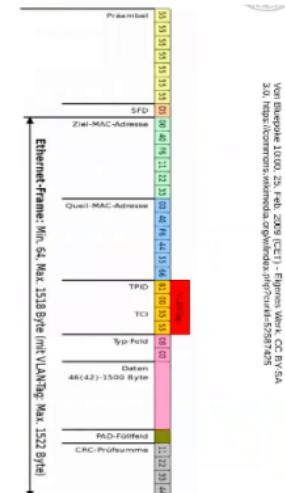
c)

- IPv4-Header (+20 Bytes) beachten
- Fragment Offset angegeben in 64Bit (8Byte) Blöcken

Länge	Flags	Offset
500	MORE	0
116	MORE	60
500	MORE	72
116	MORE	132
348	MORE	144
500	MORE	185
116	MORE	245
500	MORE	257
116	MORE	317
348	MORE	329
500	MORE	370
116	MORE	430
500	MORE	442
116	MORE	502
348	MORE	514
500	MORE	555
100		615

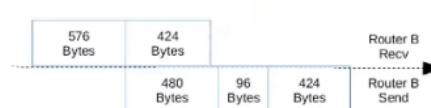
d)

- Ethernet Steuerdaten:
  - 12 Bytes MAC DST/SRC
  - 2 Bytes Ethertype
  - 4 Bytes CRC-Cheksum
  - 7 Bytes Präambel
  - 1 Byte SFD
- IPv4 Header:
  - 20 Bytes
- Insgesamt 46 Bytes Overhead pro Fragment
- 46 Bytes Overhead pro Fragment
- Router B sendet bereits nach Empfang des 1. Fragments



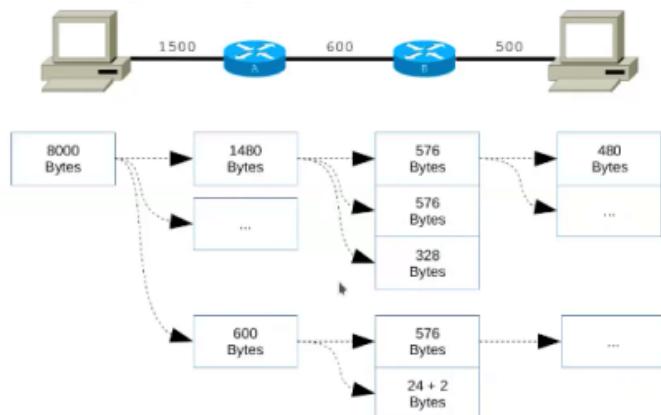
$$\begin{aligned} d_{\text{prop}} &= 100 \text{ m} / (2 \cdot 10^8 \frac{\text{m}}{\text{s}}) &= 0.5 \mu\text{s} \\ d_{\text{trans}, N} &= (N + 46) \cdot 8 \text{ Bit} / 1 \text{ Gbps} \end{aligned}$$

$$\begin{aligned} t_A &= d_{\text{prop}} + d_{\text{trans}, 1000} &= 8.868 \mu\text{s} \\ t_B &= t_A + d_{\text{prop}} + d_{\text{trans}, 576} &= 14.344 \mu\text{s} \\ t_{\text{hella}} &= t_B + d_{\text{prop}} + d_{\text{trans}, 480} + d_{\text{trans}, 96} + d_{\text{trans}, 424} &= 23.948 \mu\text{s} \end{aligned}$$



e)

- 8000 Bytes kommen an als:
  - $5 * 1480$  Byte
  - $1 * 600$  Byte
- 1480 Bytes gesendet als:
  - $2 * 576$  Byte
  - $1 * 328$  Byte
- 600 Bytes gesendet als:
  - $1 * 576$  Byte
  - $1 * 26$  Byte (2 Byte Padding da min. 46 Byte Nutzdaten pro Rahmen)



- Router A sendet bereits nach Empfang des 1. Fragments bei:

$$t_1 = d_{\text{prop}} + d_{\text{trans} \ 1480} = 12.708 \mu\text{s}$$

- Router A sendet:

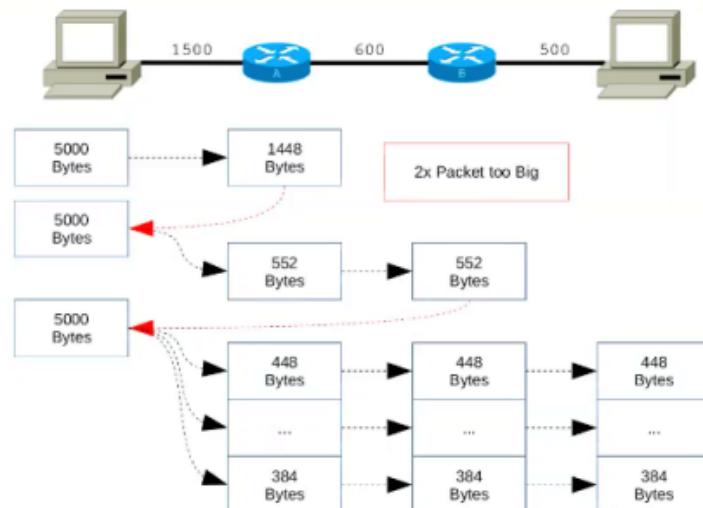
- 11 Fragmente mit 576 Bytes
- 5 Fragmente mit 328 Bytes
- 1 Fragment mit 26 Bytes

- Router A hat das letzte Fragment gesendet nach:

$$t = t_1 + 11 \cdot d_{\text{trans} \ 576} + 5 \cdot d_{\text{trans} \ 328} + d_{\text{trans} \ 26} = 12.708 \mu\text{s} + 70.272 \mu\text{s} = 82.980 \mu\text{s}$$

f)

Länge	Flags	Offset
496	MORE	0
496	MORE	56
496	MORE	112
496	MORE	168
496	MORE	224
496	MORE	280
496	MORE	336
496	MORE	392
496	MORE	448
496	MORE	504
496	MORE	560
496	MORE	616
496	MORE	672
496	MORE	728
496	MORE	784
496	MORE	840
496	MORE	896
432		952



# ISO OSI-Schichtenmodell & Schnittbildung

## 2009 Klausur = 2014 Probeklausur

### 1. ISO OSI-Schichtenmodell

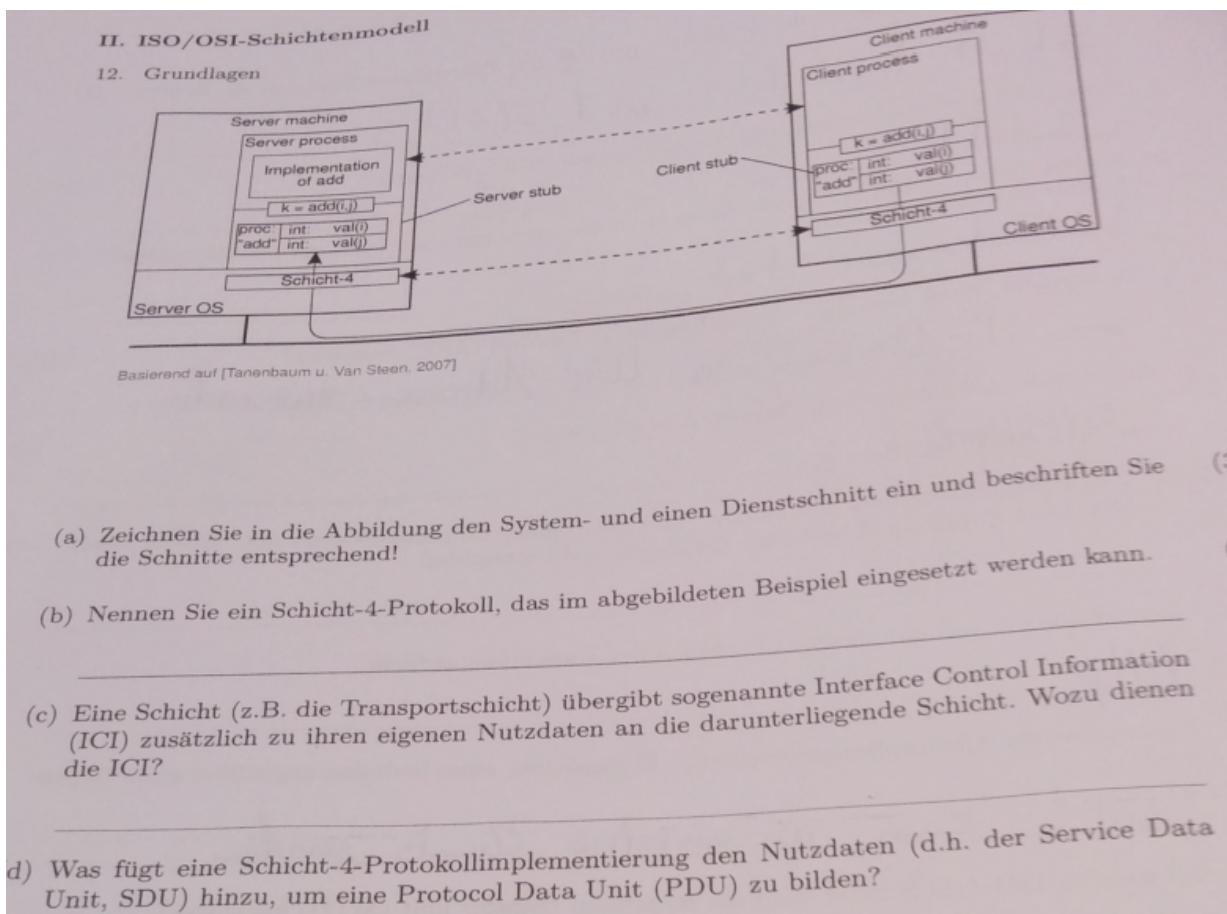
Ergänzen Sie die Namen der Schichten im ISO OSI-Schichtenmodell in Deutsch **und** Englisch und geben Sie je Schicht eine charakteristische Aufgabe an!

#### Lösung:

Hinweis: Nachfolgend sind z. T. mehr als eine charakteristische Aufgabe je Schicht aufgeführt. Entsprechend der Aufgabenstellung wäre lediglich eine charakteristische Aufgabe notwendig, um diese Aufgabe vollumfänglich zu erfüllen.

Schicht	Name	Charakteristische Aufgabe
7	Anwendungsschicht, Application Layer	Allgemein verwendbare Dienste werden standardisiert und als Dienste und Protokolle spezifiziert
6	Darstellungsschicht, Presentation Layer	Datenmodellierung in Objekten, Aushandeln der konkreten Transfersyntax, Abbilden lokale konkrete Syntax (z.B. Basic Encoding Rules, BER)
5	Kommunikationssteuerungsschicht, Session Layer	Benutzeridentifizierung, Dialogführung und Synchronisation innerhalb einer Sitzung(Session)
4	Transportschicht, Transport Layer	Verbindung zwischen zwei Prozessen, Netzunabhängiger Transport von Nachrichten zwischen zwei Endsystemen
3	Vermittlungsschicht, Network Layer	Wegewahl und Vermittlung
2	Sicherungsschicht, Data Link Layer	Zusammenfassung von Bits zu Blöcken/Frames, Fehlererkennung, ggf. Fehlerkorrektur, Medium Access Control
1	Bitübertragungsschicht, Physical Layer	Darstellung von Daten auf Medium, Transparente Übertragung von Bits

# 2014 Klausur



- (b) TCP
- (c) "Die Steuerinformationen [ICI] dienen dazu, der darunter liegenden Schicht Steuerinformationen zu übermitteln, die die Behandlung der zugehörigen Service Data Unit durch die diensterbringende Schicht näher beschreibt. Sie werden in der dienstnutzenden Schicht erzeugt und in der diensterbringenden Schicht terminiert. Es handelt sich um eine rein vertikale Kommunikation, die nur Relevanz zwischen zwei Schichten hat." - Wikipedia
- (d) PCI, da PCI + SDU = PDU  
analog: ICI + IDU

13. Einordnung von Protokollen

Geben Sie zu folgenden Protokollen den **Namen** der Schicht aus dem ISO/OSI-Schichtenmodell an, in dem das jeweilige Protokoll anzusiedeln ist.

<i>Protokoll</i>	<i>Abk.</i>	<i>Schichtname</i>
Hypertext Transfer Protocol	HTTP	
Address Resolution Protocol	ARP	
Internet Control Message Protocol	ICMP	
Open Shortest Path First	OSPF	
Internet Message Access Protocol	IMAP	

HTTP(S), IMAP, POP3, SMTP, DNS, FTP : Anwendung (7)

SSL/TLS : Darstellung (6)

TCP, UDP: Transport (4)

ICMP, OSPF, IP : Vermittlung (3)

ARP, PPP : Sicherung (2)

## 2017 Klausur = 2018 Klausur

### 2 ISO OSI-Schichtenmodell (6 Punkte)

11. Wie entsteht eine Protocol Data Unit (PDU) aus einer Service Data Unit (SDU)?

Hinzufügen von PCI

12. Eine Protokollinstanz der Schicht N tauscht PDUs mit ihrer Peer-Entity aus.

- (a) Auf welcher Schicht befindet sich die Peer-Entity?

Schicht N

- (b) An welche Schicht wird die PDU aus Schicht N übergeben

Schicht N-1

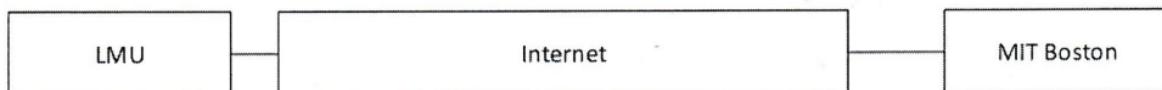
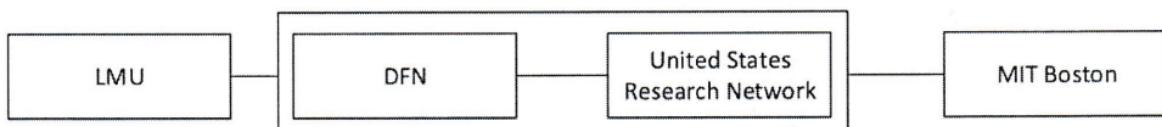
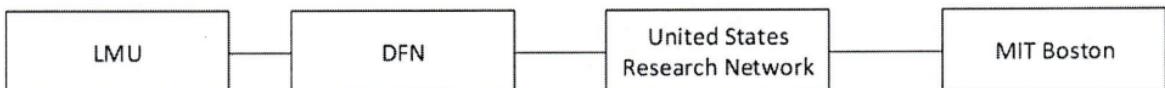
13. Geben Sie zu folgenden Protokollen den Namen der Schicht aus dem ISO/OSI-Schichtenmodell an, in dem das jeweilige Protokoll anzusiedeln ist.

Protokoll	Abk.	Schichtname
Address Resolution Protocol	ARP	<u>Sicherungsschicht</u>
Open Shortest Path First	OSPF	<u>Vermittlungsschicht</u>
Internet Control Message Protocol	ICMP	<u>Vermittlungsschicht</u>
Domain Name System	DNS	<u>Anwendungsschicht</u>

### 3 Schnittbildung (4 Punkte)

Ordnen Sie jeder der folgenden zwei Abbildung einen der Begriffe *Dienstschnitt*, *Protokollschnitt* oder *System-schnitt* zu. Begründen Sie Ihre Wahl kurz.

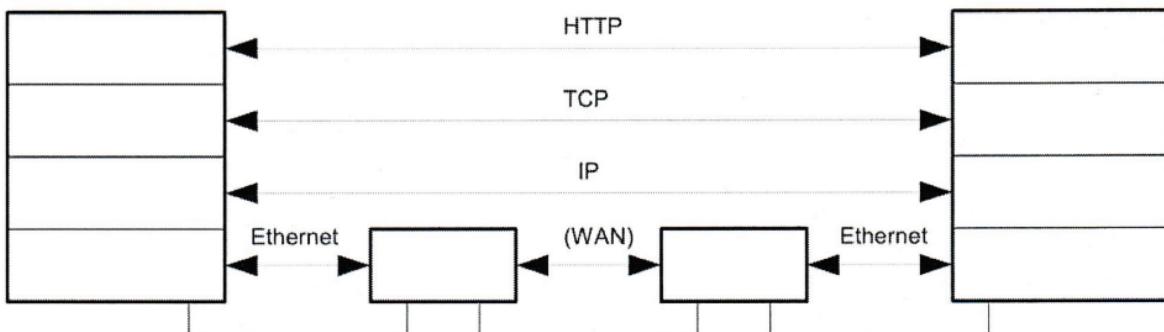
(a) Schnitt



Diese Abbildung zeigt den System-schnitt. Begründen Sie Ihre Antwort:

- Tatsächliche physischen Grenzen der Medien
- End- und Transitsysteme sind erkennbar

(b) Schnitt



Diese Abbildung zeigt den Protokollschnitt. Begründen Sie Ihre Antwort:

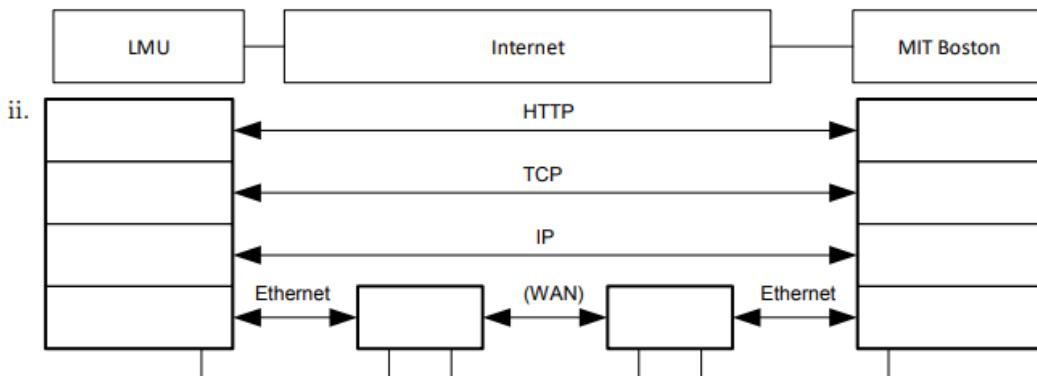
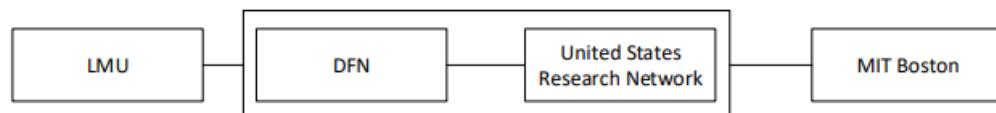
- Kommunikation zw. Peer-Entities

## 2022 Übung

### Bestandteile des Schichtenmodells (H)

Der Kommunikationsaustausch zwischen den Schichten in Rechnernetzen erfolgt über verschiedene Daten-Einheiten.

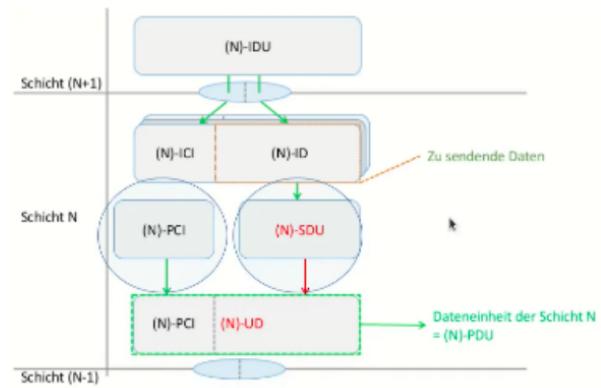
- (a) Wie entsteht eine Protocol Data Unit (PDU) aus einer Service Data Unit (SDU)?
- (b) Eine Protokollinstanz auf Schicht  $N$  tauscht PDUs mit ihrer Peer-Entity aus.
  - i. Auf welcher Schicht befindet sich die Peer-Entity?
  - ii. Wie ist der Zusammenhang zwischen PDU und SDU auf den Schichten  $N$  und  $N - 1$ ?
  - iii. Beschreiben Sie den Weg einer Nutzlast jeweils vertikal durch die Schichten in eigenen Worten.
- (c) Schnittbildung: Ordnen Sie jeder der folgenden zwei Abbildungen einen der Begriffe Dienstschnitt, Protokollschnitt oder Systemschnitt zu. Begründen Sie Ihre Wahl kurz.



## Aufgabe 2: Bestandteile des Schichtenmodells

a)

$$N\text{-PDU} = N\text{-PCI} + N\text{-SDU}$$



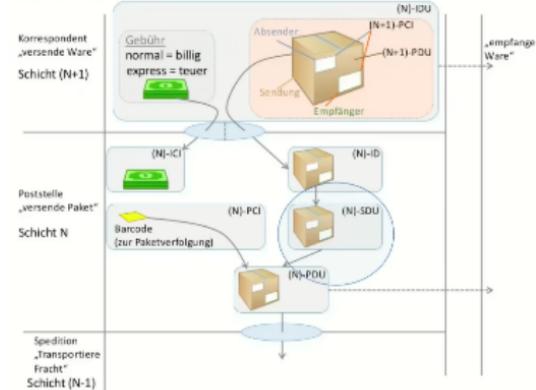
b)

i) Schicht N

$$\begin{aligned} \text{i)} \quad & 1.(N-1)\text{-SDU} = (N)\text{-PDU} \\ & 2.(N-1)\text{-PDU} = (N-1)\text{-PCI} + (N-1)\text{-SDU} \end{aligned}$$

Mit Wissen von 1. :

$$2.(N-1)\text{-PDU} = (N-1)\text{-PCI} + N\text{-PDU}$$



iii) Nutzlast N-UD + Kontrolldaten N-PCI = N-PDU

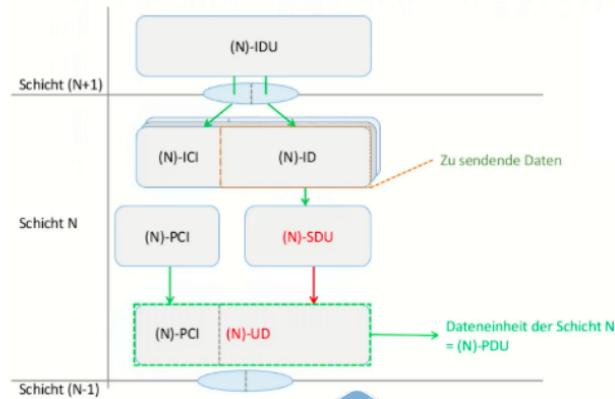
$$N\text{-PDU} + (N-1)\text{-ICI} \text{ an Schicht (N-1)}$$

(N-1)-ICI steuert Schicht (N-1)

$$\text{Nutzdaten } (N-1)\text{-SDU} = (N-1)\text{-UD}$$

$$(N-1)\text{-UD} + (N-1)\text{-PCI} = (N-1)\text{-PDU}$$

Wichtig: Jede Schicht erweitert die zu versendenden Daten um eigene Steuerdaten



# IP and Routing (Ethernet Topologie)

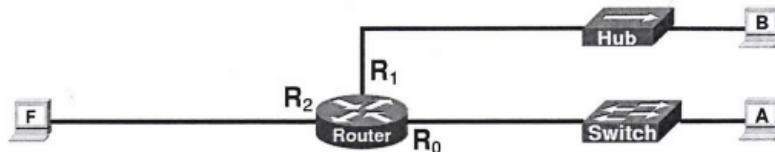
2018 Klausur = 2019 Klausur

2018 hatte noch folgende Aufgabe, über der Tabelle:

- (a) Wie viele Kollisionsdomänen zeigt die Abbildung?

## VI. IP und Routing

15. Gegeben sei die abgebildete Ethernet-Topologie.



- (a) Vergeben Sie für die Rechner A, B, F und für die drei Schnittstellen des Routers je eine IPv6-Adresse, so dass alle Rechner in unterschiedlichen Subnetzen liegen und jeder Rechner den Router erreichen kann. (3)

Benutzen Sie hierfür die Subnetze fd00::a:0/112 fd00::b:0/112 sowie fd00::f:0/112.

Rechner	IP-Adresse	Schnittstelle	IP-Adresse
A	fd00::a:1	R <sub>0</sub>	fd00::a:2
B	fd00::b:1	R <sub>1</sub>	fd00::b:2
F	fd00::f:1	R <sub>2</sub>	fd00::f:2

- (b) Schreiben Sie eine default-Route für die Routingtabelle von Rechner F, so dass IP-Nachrichten an die Rechner A und B korrekt weitergeleitet werden! (2)

Ziel Subnetz: \_\_\_\_\_

erreichbar über: \_\_\_\_\_

16. Angenommen der Block 172.16.0.0/12 wird mit der Subnetzmaske 255.255.255.128 aufgeteilt. (1)  
Wieviele Subnetze lassen sich damit maximal realisieren?

zu 15.

- (b) Ziel Subnetz: ::/0, erreichbar über: fd00::f:2

zu 16.

1. Die vorderen **12 Bit** sind Netz-Anteil, da 172.16.0.0/12

2. Subnetzmaske in binär umwandeln:  $255.255.255.128 = 1111\ 1111.1111\ 1111.1111\ 1111.1000\ 0000$   
→ vordere 25 Bit sind 1en →  $25 - 12 = 13$  Bit Subnetz-Anteil
  3.  $2^{13} - 2 = 8192 - 2 = 8190$  mögliche Subnetze. 2 wird abgezogen, weil die Adressen, in denen der Subnetz-Anteil nur 0en sowie nur 1en sind, speziell sind und nicht für Subnetze genutzt werden.
17. Notieren Sie die IPv6 Adresse **1337:0000:0000:0000:1000:0000:0000:0001** maximal verkürzt, (1) so dass keine kürzere vollständige Darstellung dieser Adresse in IPv6 existiert.
- 
18. Argumentieren Sie kurz, warum Router F ein Paket mit Zieladresse 10.0.0.8 nicht ins Internet (2) weiterleiten sollte.

zu 17.

1337::1000:0:0:1

zu 18.

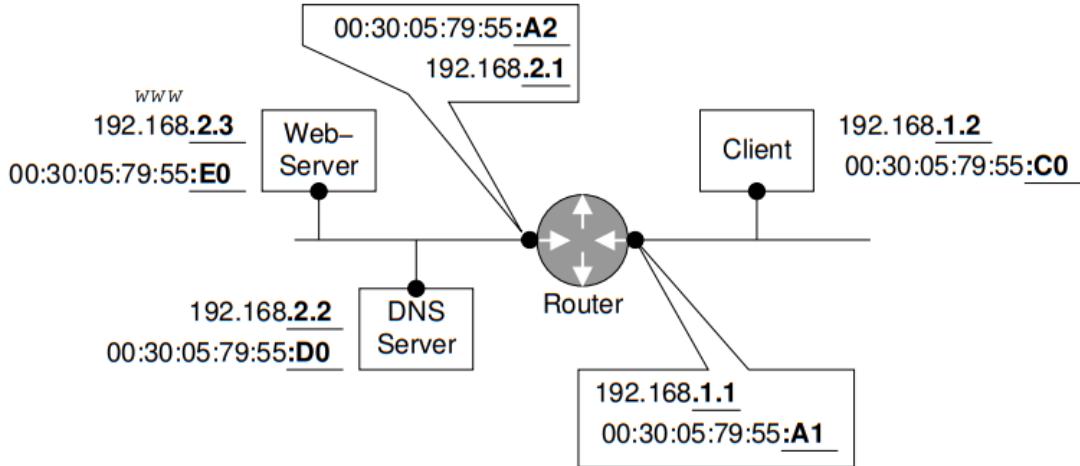
10.0.0.8 ist eine private Adresse. Sie ist nicht eindeutig und könnte mit anderen Geräten in privaten Netzwerken kollidieren, wenn sie ins Internet geleitet wird. Daher werten Router Pakete mit privaten Zieladressen als nicht ins Internet weiterleitbar.

# Kommunikationsablauf

## 2014 Probeklausur = 2022 Klausur

Hinweise:

- Benutzen Sie beim **Eintragen** in die Tabelle von...
  - MAC-Adressen nur das **letzte Byte**,
  - IP-Adressen nur die **letzten 2 Byte**.
  - Broadcasts die Angabe **B-Cast**.
  - Die erste Tabellenzeile ist als Beispiel vorgegeben.
- Der **Client** kennt:
  - die IP-Adresse seines lokalen DNS-Servers,
  - die URL des abzufragenden Web-Objekts und
  - eine Default-Route über .1.1 .
- Der **DNS-Server** ist **autoritativ** für alle Teilnehmer in der Abbildung.
- **Caches** (ARP, DNS, ...):
  - es existieren keine aktuellen Cache-Werte.
  - empfangene/aufgelöste Adressen werden aggressiv zwischengespeichert, und müssen nicht wieder angefragt werden.
- Eine Nachricht einer Schicht N passt immer in eine PDU der Schicht N-1.
- Vernachlässigen Sie Übertragungsfehler, Verluste oder verworfene Nachrichten!



Pkt	MAC-Adr		IP-Adr		Port		TCP	
	von	zu	von	zu	von	zu	Flags	Payload/Erklärung
1	:C0	B-Cast	-	-	-	-	-	ARP: wer hat .1.1?
2			-	-	-	-	-	ARP: ich habe .1.1!
3					12345		-	DNS Query: www?
4								ARP: wer hat .2.2?
5								ARP: ich habe .2.2!
6								
7					12345		-	
8							-	
9					4711			Verbindungsaufbauwunsch
10		B-Cast						
11	:E0							
12					4711			
13						4711		
14								Bestätigung Aufbauwunsch
15							ACK	
16					4711			

Vervollständigen Sie in der folgenden Tabelle die Kommunikation aller Teilnehmer auf den OSI-Schichten 2, 3 und 4, bis eine TCP-Verbindung zwischen Client und Webserver vollständig aufgebaut ist (heißt: alle relevanten Pakete wurden zugestellt).

### Lösung:

Bewertung: 1 Punkt pro richtige Zeile. 1/2 Punkt pro sinnvolle Zeile mit kleinem Fehler. Die Gesamtpunktezahl für die Aufgabe wird zugunsten Studi auf nächste ganze Punktezahl gerundet.

Pkt	MAC-Adr		IP-Adr		Port		TCP Flags	Payload/Erklärung
	von	zu	von	zu	von	zu		
1	:C0	B-Cast	-	-	-	-	-	ARP: wer hat .1.1?
2	:A1	:C0	-	-	-	-	-	ARP: ich habe .1.1!
3	:C0	:A1	.1.2	.2.2	X	dns(53)	-	DNS Query: www?
4	:A2	B-Cast	-	-	-	-	-	ARP: wer hat .2.2?
5	:D0	:A2	-	-	-	-	-	ARP: ich habe .2.2!
6	:A2	:D0	.1.2	.2.2	12345	dns(53)	-	DNS Query: www?
7	:D0	:A2	.2.2	.1.2	dns(53)	12345	-	DNS Response: .2.3
8	:A1	:C0	.2.2	.1.2	dns(53)	12345	-	DNS Response: .2.3
9	:C0	:A1	.1.2	.2.3	4711	www(80)	SYN	Conn-Req
10	:A2	B-Cast	-	-	-	-	-	ARP: wer hat .2.3?
11	:E0	:A2	-	-	-	-	-	ARP: ich habe .2.3!
12	:A2	:E0	.1.2	.2.3	4711	www(80)	SYN	Conn-Req
13	:E0	:A2	.2.3	.1.2	www(80)	4711	SYN,ACK	Conn-Req-Ack
14	:A1	:C0	.2.3	.1.2	www(80)	4711	SYN,ACK	Conn-Req-Ack
15	:C0	:A1	.1.2	.2.3	4711	www(80)	ACK	Conn-Est
16	:A2	:E0	.1.2	.2.3	4711	www(80)	ACK	Conn-Est

Warum dns(53) und www(80)?

DNS-Anfragen normalerweise per UDP Port 53

-> dns(53)

HTTP ist TCP Port 80 zugewiesen

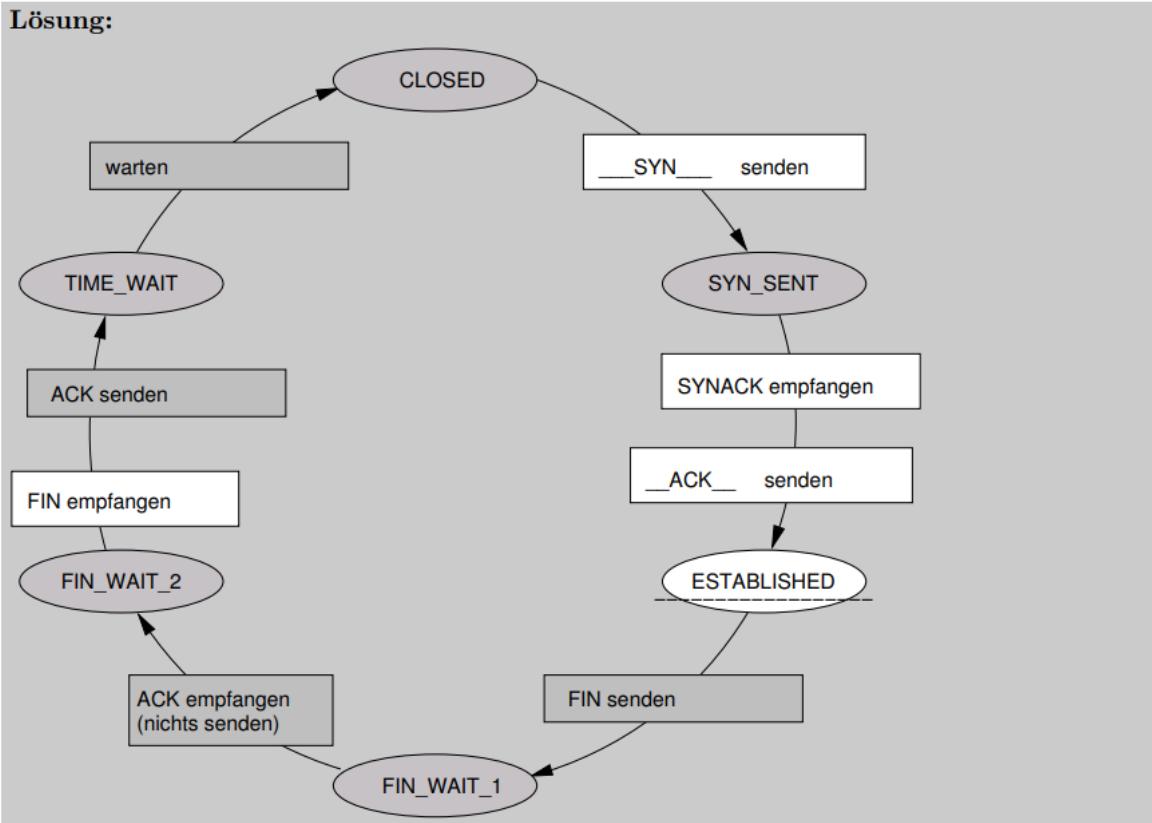
-> www(80)

# Transmission Control Protocol (TCP)

2009 Klausur = 2014 Probeklausur

## 6. Transmission Control Protocol

Das Diagramm zeigt die Zustände und Zustandsübergänge in einem TCP-basierten Client. Ergänzen Sie den Text in den weißen Flächen!

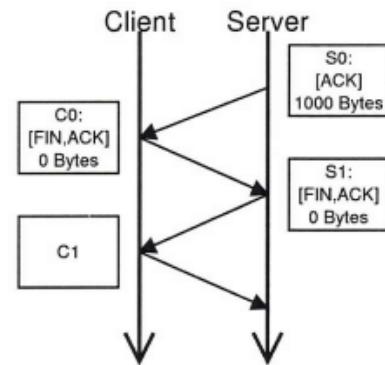


2014 Klausur = 2019 Klausur

## 21. Flusssteuerung

Zwei Rechner führen einen Anfrage-Anwort-Dialog aus, der über eine TCP-Verbindung zwischen einem Client- und einem Serverprozess transportiert wird. Die folgende Abbildung zeigt das Ende der Kommunikation. Es gelten folgende Randbedingungen:

- Es treten keinerlei Fehler auf.
- Es gibt keine ausstehenden Quittungen.



- (a) Was ist ausschlaggebend für das Fortzählen von TCP-Sequenznummern? (1)
- 
- (b) Welche gesetzten Flags im Header eines TCP-Segments werden immer durch ein ACK quittiert? (1)
- 
- (c) Welche Flags im TCP-Header sind in dem Segment C1 gesetzt? (1)
- 
- (d) Der Client begann den Verbindungsauftbau. Wieviele Segmente ohne Nutzdaten hat der Client im Verlauf der gesamten Kommunikation **mindestens** an den Server geschickt? (1)
- (a) Die Nutzdaten Größe in Bytes und die Seq der vorherigen Nachricht werden addiert  
 (b) SYN, FIN  
 (c) ACK  
 (d) Mindestens 4
- (e) Die Tabelle beinhaltet Einträge für die Sequenz- und Quittungsnummern für die oben in der Abbildung dargestellten TCP-Segmente in der Reihenfolge, in der sie gesendet werden. Vervollständigen Sie die Tabelle, indem Sie die fehlenden Sequenz- und Quittungsnummern für C0 und S1 eintragen.

	Seq-nummer	Ack-Nummer
S0	1051	6000
C0		
S1		

- (e) C0: Seq = 6000, ACK Nr = 2051 (1000 Byte Nutzdaten)

S1: Seq = 2051, ACK Nr = 6001 (0 Byte Nutzdaten, aber Phantom Byte wird zu 2051 addiert, da FIN Flag in S1 gesetzt)

(nicht mehr nötig, aber fürs Verständnis: C1: Seq = 6001, ACK Nr = 2052 (0 Byte Nutzdaten, aber Phantom Byte wird zu 2051 addiert, da FIN Flag in S1 gesetzt))

3. Die Tabelle beinhaltet Einträge für die Sequenz- und Quittungsnummern für die oben in der Abbildung dargestellten TCP-Segmente in der Reihenfolge, in der sie gesendet werden. Vervollständigen Sie die Tabelle, indem Sie die fehlenden Sequenz- und Quittungsnummern für C0 und S1 eintragen!

	Seq-nummer	Ack-Nummer
S0	8000	2049
C0		
S1		

In der 2014 Klausur waren die Aufgaben identisch, bis auf die gegebenen Seq und ACK Nr. (siehe oben):

C0: Seq = 2049, ACK Nr = 9000 (1000 Byte Nutzdaten)

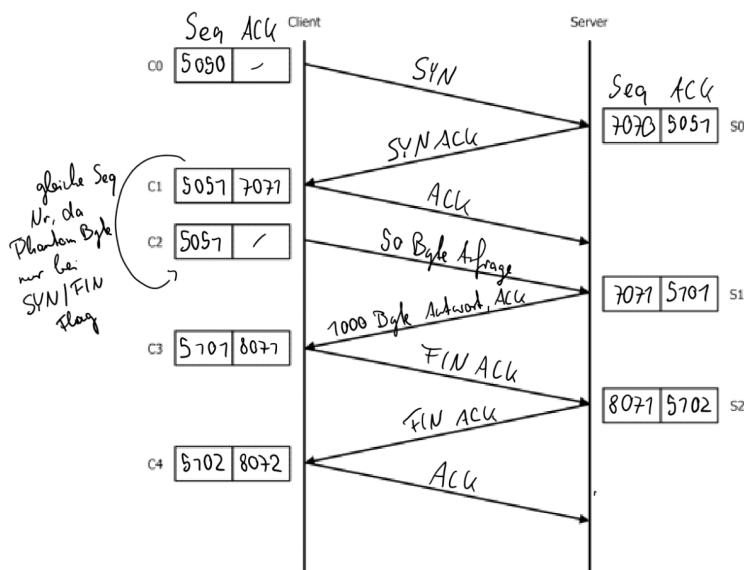
S1: Seq = 9000, ACK Nr = 2050

(nicht mehr nötig, aber fürs Verständnis: C1: Seq = 2050, ACK Nr = 9001 (0 Byte Nutzdaten, aber Phantom Byte wird zu 2051 addiert, da FIN Flag in S1 gesetzt))

## 2015 Klausur

### VII. TCP

15. Folgendes Diagramm beschreibt einen HTTP-Anfrage-Antwort-Dialog zwischen einem Client und einem Server inkl. des TCP Auf- und Abbaus.  
Die Anfrage sei 50 Byte groß, die Antwort 1000 Byte.
- a. Tragen Sie in das Sequenzdiagramm alle gesendeten TCP-Flags ein (und nichts sonst)!



Zu Beginn der Übertragung stehe der Sequenznummernzähler des Clients auf 5050, der des Servers auf 7070.

- b. Welche Sequenznummer und welche ACK-Nummer wird in Paket S0 übertragen?  $\rightarrow \text{Seq} = 7070, \text{ACK} = 5051$   
 c. In welchem Paket geschieht die HTTP-Anfrage?  $\rightarrow C2$   
 d. Die Serverantwort trage die Sequenznummer  $x$  und die ACK-Nummer  $y$ . Wie lauten die Sequenznummer und die ACK der Bestätigung/Quittung dazu?  $\rightarrow \text{Seq} = y, \text{ACK} = x + 1000$   
 e. Wie groß ist der Unterschied der Sequenznummern zwischen C3 und C4?  $\rightarrow 8071 - 7071 = 1$  (=Phantom Byte der FIN Flag)  
 f. Um wieviel wurde der Sequenznummernzähler des Servers während des Dialogs insgesamt erhöht?

$\hookrightarrow 7071$  ist die sichere Antwort, aber  $7002$  ist auch nicht ganz verkehrt wegen der letzten FIN Flag.

Seq. Nr. wird direkt nach dem Absenden einer Nachricht 1 berechnet,  
während ACK Nr. vor dem Absenden der Nachricht 2 berechnet werden.

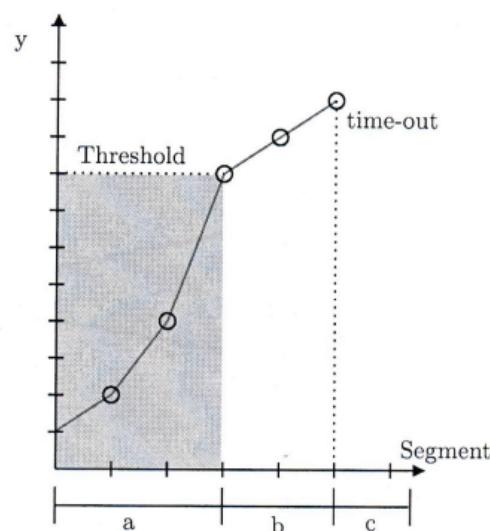
Man könnte also argumentieren, dass der Server die Seq Nr.  
 $\underbrace{7000+7+1}$  erhält, das kann ich nicht sagen ohne die Lösung zu kennen...

3 mal VS 2 mal bei ACK Nr.

## 2017 Klausur

## 25. Überlastkontrolle

Die folgende Abbildung zeigt die Zeitliche Entwicklung einer kritischen Kenngröße bei der TCP-Tahoe Überlastkontrolle. Der Ablauf ist in die Phasen a, b und c eingeteilt.



- (a) Wie lautet die Beschriftung für die y-Achse?

Congestion windows

- (b) Benennen Sie Phase a:

exponentielle Phase (Slow Start)

- (c) Benennen Sie Phase b:

lineare Phase

- (d) Was ist ein frühzeitiger Indikator für Paketverluste, noch bevor ein *timeout* für ein Paket auftritt?

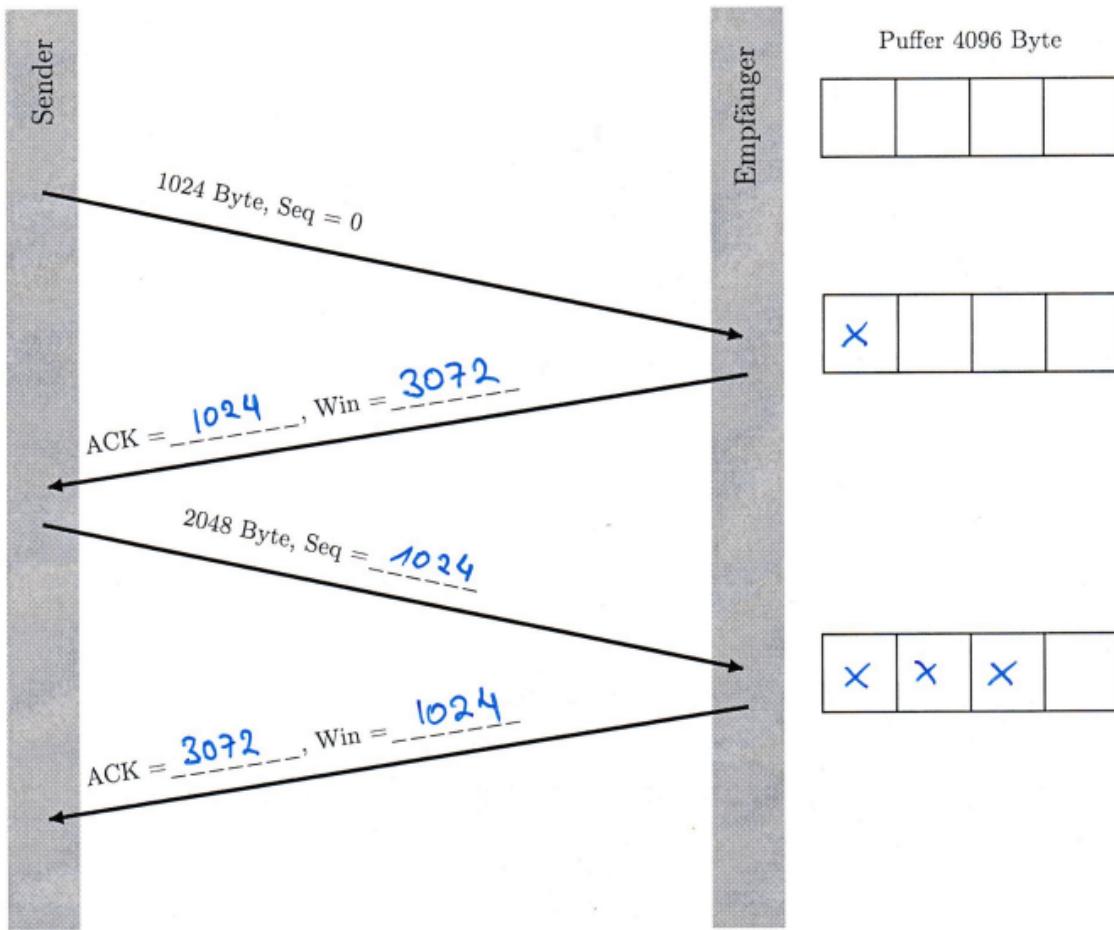
Reitungsduplikate

- (e) Nennen Sie ein Verfahren zur Optimierung des Überlastkontrollalgorithmus nach Auftreten eines  *timeouts*.

Threshold halten und never Slow Start

## 26. Flusssteuerung

Die Folgende Abbildung zeigt eine Datenübertragung unter Verwendung von TCP unter Vernachlässigung des Verbindungsau- und -abbaus.



- (a) Vervollständigen Sie die fünf fehlenden Angaben in der Abbildung mit Sequenznummern (*Seq*, 1x), Beatätigungsnummern (*ACK*, 2x) und Fenstergrößen (*Win*, 2x).
- (b) Markieren Sie den **belegten** Pufferspeicher des Empfängers in oben stehender Abbildung. Nehmen Sie an, dass der Pufferspeicher des Empfängers zu Beginn der skizzierten Übertragung leer ist. *Hinweis:* 1 Kästchen entspricht 1024 Byte.
- (c) Durch welches Ereignis wird der Pufferspeicher auf Empfängerseite wieder frei?

Einlesen/Verarbeiten der Daten auf Empfängerseite

## 2019 Klausur

### VII. Transmission Control Protocol (TCP)

#### 19. Staukontrolle

Beantworten Sie folgende Fragen zu TCP Tahoe und zeichnen Sie den zeitlichen Ablauf wie beschrieben in die untenstehende Abbildung ein.

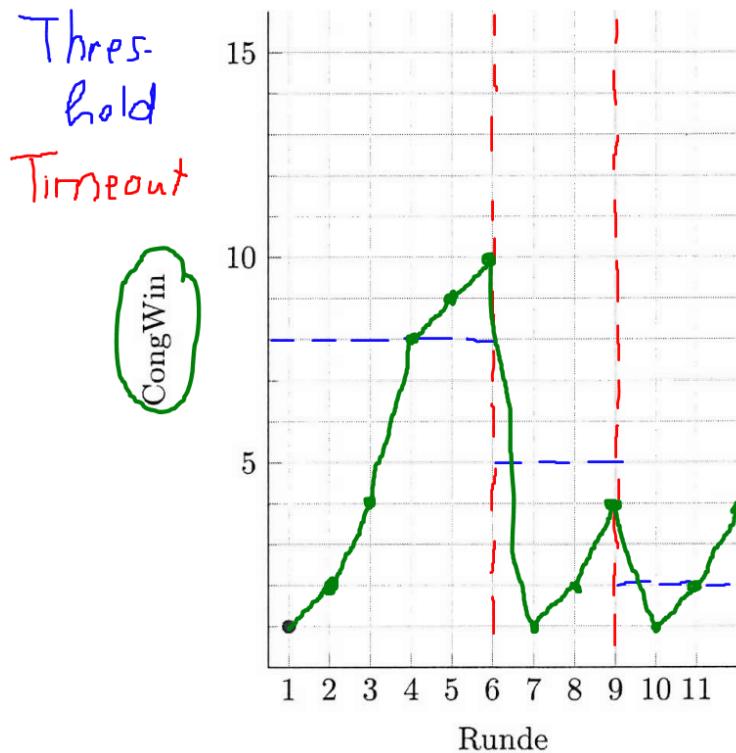


Abbildung 2: Staukontrolle unter TCP Tahoe

- (a) Visualisieren Sie in Abbildung 2 die Fenstergröße über alle elf Runden. Starten Sie mit  $\text{CongWin} = 1$ . (4)
- Threshold sei zu Beginn 8.
  - Sowohl in Runde 6 als auch 9 tritt ein Segmentverlust durch Timeout auf.

*Hinweis:* Auf Seite 15 befinden sich zusätzliche, leere Abbildungen. Gewertet wird *nur* Ihre Lösung in Abbildung 2.

- (b) In welcher Einheit wird die Y-Achse gemessen? (1)

- (c) Benennen Sie die Phase von Runde 1 bis Runde 4. (1)

- (d) Benennen Sie die Phase von Runde 4 bis Runde 6. (1)

- (e) Welchen Wert hat Threshold in Runde 7. (1)

- (b) In MSS = Maximum Segment Size  
 (c) Slow Start / exponentielles Wachstum  
 (d) Congestion Avoidance / Lineares Wachstum  
 (e) 5

20. Nehmen Sie nun an, wir verwenden in Aufgabe 19 das optimierte TCP Reno (Fast Recovery) Verfahren.

(a) Bei welchem Ereignis wechselt TCP Reno in die *Fast Recovery* Phase?

(b) Welchen Wert hat CongWin, wenn in Runde 6 ein Segmentverlust durch 3 Quittungsduplikate festgestellt wird?

(a) Bei 3 erhaltenen Quittungsduplikaten / duplizierten ACKs

(b) 8, da  $10 / 2 + 3 = 8$

## 2022 Übung

### 2. TCP Reno (H)

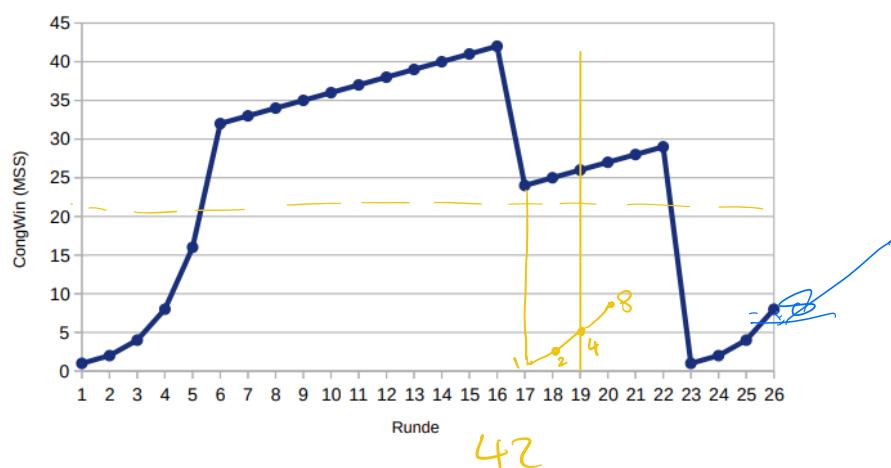


Abbildung 1: Verhalten von TCP Reno

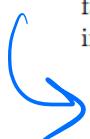
In Abbildung 1 ist das Verhalten vom TCP-Reno zu sehen. TCP-Reno verhält sich ähnlich wie TCP-Tahoe, jedoch mit Unterstützung für *Fast Recovery*:

Empfängt der Sender 3 ACK-Duplikate, geht er – anstelle von Slow Start – in den Fast Recovery Zustand über. Dabei wird zwar auch  $\text{Threshold} = \frac{\text{CongWin}}{2}$  gesetzt, allerdings wird  $\text{CongWin} = \frac{\text{Threshold}}{2} + 3$  MSS gesetzt. In der Fast Recovery Phase wird CongWin für jedes empfangene ACK-Duplikat um 1 MSS erhöht. Erreichen den Sender wieder *neue* Quittungen (also keine ACK-Duplikate), geht Reno in lineare Phase (Congestion Avoidance) über.

In dieser Aufgabe nehmen wir an, dass während Fast Recovery keine weiteren Duplikate auftreten. Das heißt, dass das erneut übertragene Segment – die Ursache für die ACK-Duplikate – erfolgreich quittiert wird.

thre  
Congwin

- (a) Identifizieren Sie die Intervalle, in denen TCP Slow Start aktiv ist.
- (b) Identifizieren Sie die Intervalle, in denen TCP Congestion Avoidance aktiv ist.
- (c) Wurde der Paketverlust nach der 16. Runde durch duplizierte ACKs oder durch die Überschreitung des Timeouts erkannt? Warum?
- (d) Wurde der Paketverlust nach der 22. Runde durch duplizierte ACKs oder durch die Überschreitung des Timeouts ausgelöst? Warum?
- (e) Welchen Wert hat *Threshold* zu Beginn (in der 1. Runde)?
- (f) Welchen Wert hat *Threshold* in der 18. Runde?
- (g) Welchen Wert hat *Threshold* in der 24. Runde?
- (h) In welcher Runde wird das 70. Segment gesendet?
- (i) Angenommen in der 26. Runde wird ein Paketverlust durch ein (dreifaches) ACK-Duplikat festgestellt. Wie werden die Werte von *CongWin* und *Threshold* anschließend sein?
- (j) Angenommen es würde TCP Tahoe statt Reno genutzt und in der 16. Runde wird durch ein (dreifaches) ACK-Duplikat ein Paketverlust festgestellt. Welche Werte haben *Threshold* und *CongWin* in der 19. Runde?



$$\begin{aligned}
 & \text{Threshold} \quad \text{Congwin}/2 = 1 \\
 & \text{Congwin } 17 = 1 \\
 & \sim \quad \sim \quad 18 = 2 \\
 & \sim \quad \sim \quad 19 = 4
 \end{aligned}$$

**Aufgabe 2: TCP Reno**

a)

Runde 1-6, Runde 23-26

b)

Die linearen Phasen: Runden 7-16, 17-22

c)

Duplizierte ACKs

TCP Reno wechselt nach Timeout in Slow Start (nicht in Fast Recovery)

Fast Recovery:  $\text{CongWin} = (\text{CongWin}/2) + 3$ 

d)

Timeout, da CongWin auf 1 MSS gesetzt wird

e)

 $2^5 = 32$ , da Threshold nach 5 Runden erreicht

f)

Beim vorherigen Paketverlust wurde  $\text{Threshold} = \text{CongWin}/2 = 42/2 = 21$  gesetzt

g)

 $\text{Threshold} = \text{CongWin}/2 = 29/2 = 14$  gesetzt

h)

Runde	CongWin	Segmente
1	1	1
2	2	3
3	4	7
4	8	15
5	16	31
6	32	63
7	33	96

in der 7. Runde

i)

 $\text{Threshold} = \text{CongWin}/2 = 8/2 = 4$ Dann  $\text{CongWin} = \text{Threshold} + 3 = 7$ 

j)

 $\text{Threshold} = \text{CongWin}/2$ Dann  $\text{CongWin} = 1$  und Slow-Start beginntRunde 17:  $\text{CongWin} = 1$ Runde 18:  $\text{CongWin} = 2$ Runde 19:  $\text{CongWin} = 4$

## 2. TCP Sequenznummern (H)

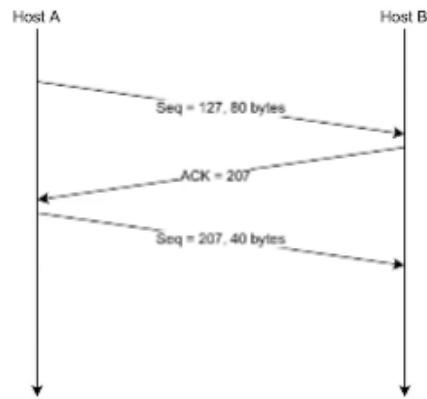
Zwei Hosts A und B kommunizieren über eine TCP Verbindung. Host B hat bereits 126 Bytes von Host A vollständig empfangen und Host A sendet zwei weitere Segmente der Größen 80 sowie 40 Bytes. Die Sequenznummer des ersten Segments ist 127, der Quellport ist 302 und der Zielport ist 80. Host B sendet ein Acknowledgement immer, sobald es ein Segment von Host A empfangen hat.

- Wie lauten Sequenznummer, Quell- sowie Zielport des *zweiten* Segments von Host A an B?
- Falls das erste Segment *vor* dem zweiten Segment bei B eintritt, wie lauten im ACK (Quittung) die ACK-Nr., Quell- und Zielport?
- Falls das erste Segment *nach* dem zweiten Segment bei B eintritt, wie lauten im ACK (Quittung) die ACK-Nr., Quell- und Zielport?
- Angenommen, beide Segmente kommen in der richtigen Reihenfolge von A zu B. Das erste ACK von B geht verloren und das zweite ACK erreicht A nach dem ersten Timeout-Intervall. Zeichnen Sie ein Sequenzdiagramm und beschriften Sie den jedes versendete Segment vollständig mit Sequenznummer, Anzahl der Nutzdaten-Bytes. Beschriften Sie des Weiteren alle Quittungen (ACKs) mit der korrekten ACK Nummer.

### Aufgabe 2: TCP Sequenznummern

a)

ACK = 207, Quellport = 302, Zielport = 80



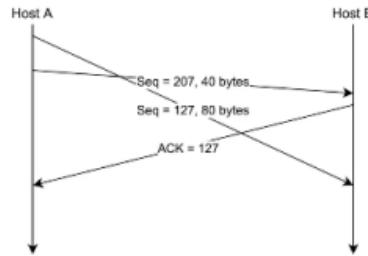
zu c): Erstes ACK (Seq 1, 126 Bytes -> ACK 127) wird wiederholt, da bekommene Seq 207 != 127 erwartete Seq

b)

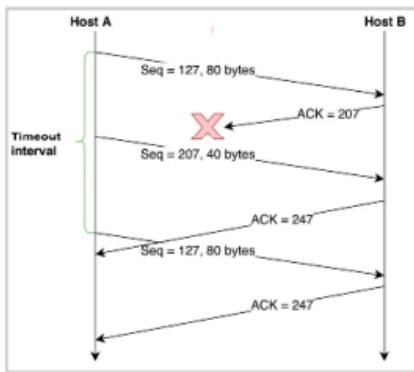
Erste Segment kommt auf jeden Fall vor dem zweiten Segment ein, daher ist ACK = 207. Quellport und Zielport sind genau andersrum, also Quellport = 80 und der Zielport = 302.

c)

ACK = 127, Quellport = 80, Zielport = 302



d)



## Übertragungsraten / Von Signalen zu Bitströmen / Codierungsverfahren

### Codierungsverfahren

Start bei beiden unten (einfach weil es in der 2014 Probeklausur auch so war)

**(dif.) Manchester Codierung darf immer nur maximal Taktlänge \* 2 auf einer Seite sein!**

Manchester:

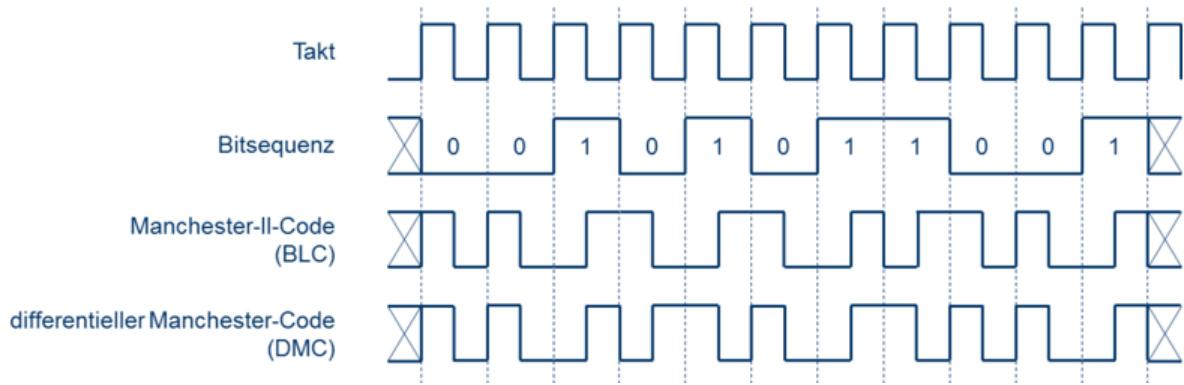
- Bei 0 → 0 oder 1 → 1 Bitwechsel wird zwischen den beiden Bits gewechselt
- Bei 0 → 1 oder 1 → 0 Bitwechsel wird während des 1. und 2. Bit gewechselt

dif. Manchester:

- Bei 0 → 0 oder 1 → 0 Bitwechsel wird zwischen den beiden Bits gewechselt
- Bei 1 → 1 oder 0 → 1 Bitwechsel wird während des 1. und 2. Bit gewechselt

Übertragungsraten in bit/s: Anzahl Bits / Übertragungsdauer.

Baud-Rate = 2\*Übertragungsraten.

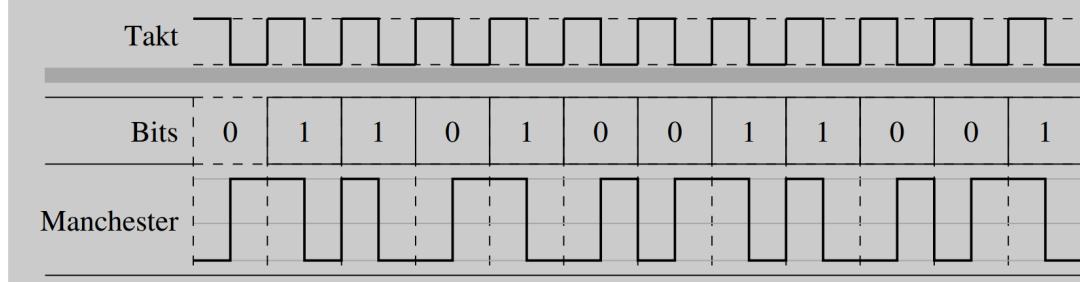


2009 Klausur = 2014 Probeklausur

#### Codierungsverfahren

- (a) Geben Sie das in Manchestercodierung dargestellte Bitmuster an! (2)

**Lösung:**



- (b) Angenommen, das obige Bitmuster für Manchestercodierung wird in 1 ms übertragen. (2)

- i. Wie hoch ist die Übertragungsrate?

**Lösung:**

$$\text{Übertragungsrate} = \frac{12\text{Bit}}{1\text{ms}} = 12000\text{bit/s}$$

- ii. Wie hoch ist die Baud-Rate des Signals?

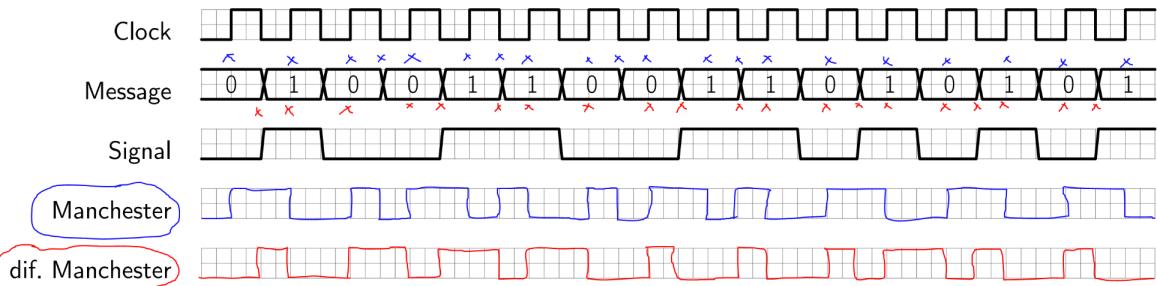
**Lösung:**

$$\text{Manchestercodierung} \Rightarrow \text{Baud-Rate} = 2 * \text{Übertragungsrate} = 24000 \text{ Baud}$$

2022 Übung

#### 2. Leitungscodierung

Das Signal 0100 1100 1101 0101 soll leitungscodiert werden.



- (a) Kann die Modulationsart bestimmt werden? Wenn ja, um welche Modulationsart handelt es sich?  
 (b) Zeichnen Sie den Manchester-Code in die Abbildung ein.  
 (c) Zeichnen Sie den differentiellen Manchester-Code ein.

# Übertragungsraten / Von Signalen zu Bitströmen TODO

2009 Klausur = 2014 Probeklausur

## Übertragungsraten

Über ein Medium mit einer Bandbreite von 1 MHz wird mit einer 2-Stufencodierung übertragen. Wie viele Bits pro Sekunde können maximal übertragen werden, wenn... .

*Hinweis:* Geben Sie für jede Teilaufgabe jeweils eine Rechnung (Formel) und ein Ergebnis an!

- (a) ... kein Rauschen vorkommt (ideales Medium)?

**Lösung:**

$$\begin{aligned} \text{Ansatz mit Nyquist: } C &= 2 \cdot B \cdot \log_2 M \\ &= 2 \cdot 1 \text{MHz} \cdot \log_2 2 = 2 \cdot 10^6 \text{bit/s} \end{aligned}$$

- (b) ... ein Verhältnis zwischen Signal und Rauschen von  $S/N = 1023$  vorherrscht?

**Lösung:**

$$\begin{aligned} \text{Ansatz mit Shannon: } C &= B \cdot \log_2(1 + S/N) \\ &= 1 \text{MHz} \cdot \log_2(1 + 1023) = 10^7 \text{bit/s} \end{aligned}$$

2011 Klausur

## Aufgabe 5 - Von Signalen zu Bitströmen

- a) Ein periodisches Signal s kann als Funktion in Abhängigkeit von der Zeit t dargestellt werden.

$$s(t) = A * \sin(2 * \pi * f * t + \omega)$$

Nennen Sie drei Größen, die mit Modulation verändert werden können und erklären Sie ihre jeweilige Bedeutung.

- b) Ordnen Sie jedem der folgenden Diagramme einen Term aus der obigen Gleichung zu, der zur Modifikation des Signals verändert wurde.

- c) Wieviele Bits werden pro Signalschritt übertragen, wenn 8 Signalzustände unterscheidbar sind?

- d) Gegeben seien die Grenzfrequenzen  $f = 100 \text{ MHz}$  und  $f = 50 \text{ MHz}$  eines idealen Mediums.

- Wie hoch ist die Bandbreite des Mediums?
- Berechnen Sie die maximale Übertragungsrate, die mit einer Binärcodierung über dieses Medium erreicht werden kann.

7. Modulation von analogen Daten:

Ein periodisches Signal  $s$  kann als Funktion in Abhängigkeit von der Zeit  $t$  dargestellt werden:  $s(t) = A * \sin(2 * \pi * f * t + \omega)$

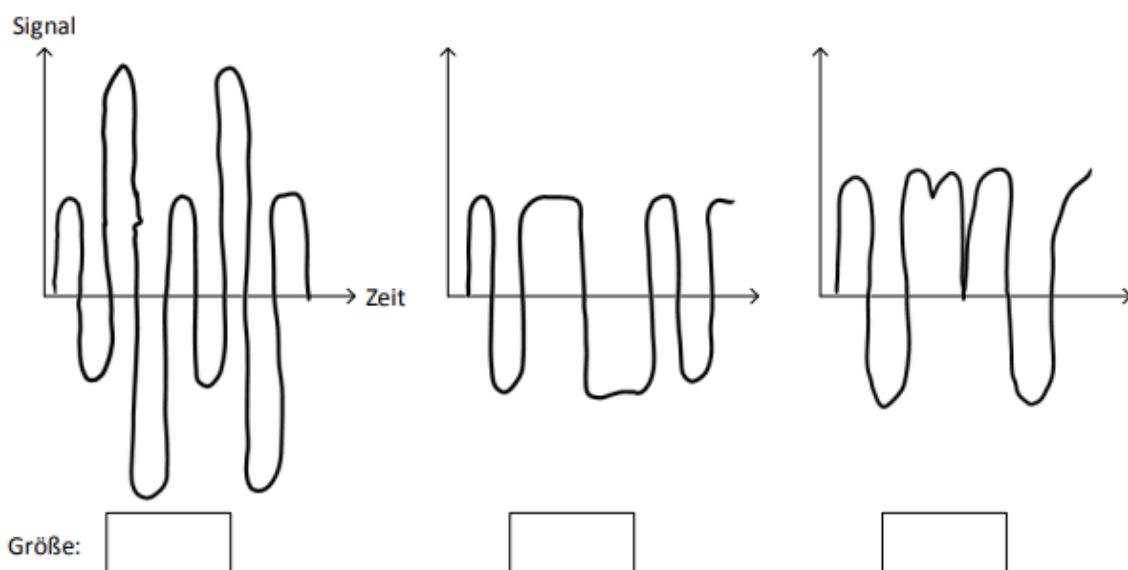
- Nennen Sie drei Größen, die mit Modulation verändert werden können und erklären Sie ihre jeweilige Bedeutung.

Größe 1:

Größe 2:

Größe 3:

- Ordnen Sie jedem der folgenden Diagramme einen Term aus der obigen Gleichung zu, der zur Modifikation des Signals verändert wurde:



$f$  beeinflusst "Dichte" der Kurven, d.h. bei hohem  $f$  sind FM Wellen dichter, bei niedrigem weiter auseinander

$A$  beeinflusst "Höhe" der Kurven, d.h. bei niedrigem  $A$  sind AM Wellen kleiner in der Höhe, bei hohem  $A$  höher

$\omega$  beeinflusst Verschiebung der Kurven. Wird  $\omega$  mit -1 multipliziert, so tritt ein Phasenwechsel ein.

## IX. Von Signalen zu Bitströmen

30. Im Folgenden finden Sie leere Koordinatensysteme zur Visualisierung von Signalmodulation. Zeichnen Sie in den Koordinatensystemen jeweils ein Beispiel der geforderten Modulationsart, sodass die **Bitfolge** 1001 übertragen wird.

*Hinweis:* Wie eine logische 1 bzw. 0 kodiert wird, ist für jedes Modulationsverfahren frei wählbar.

(a) Amplituden-Modulation

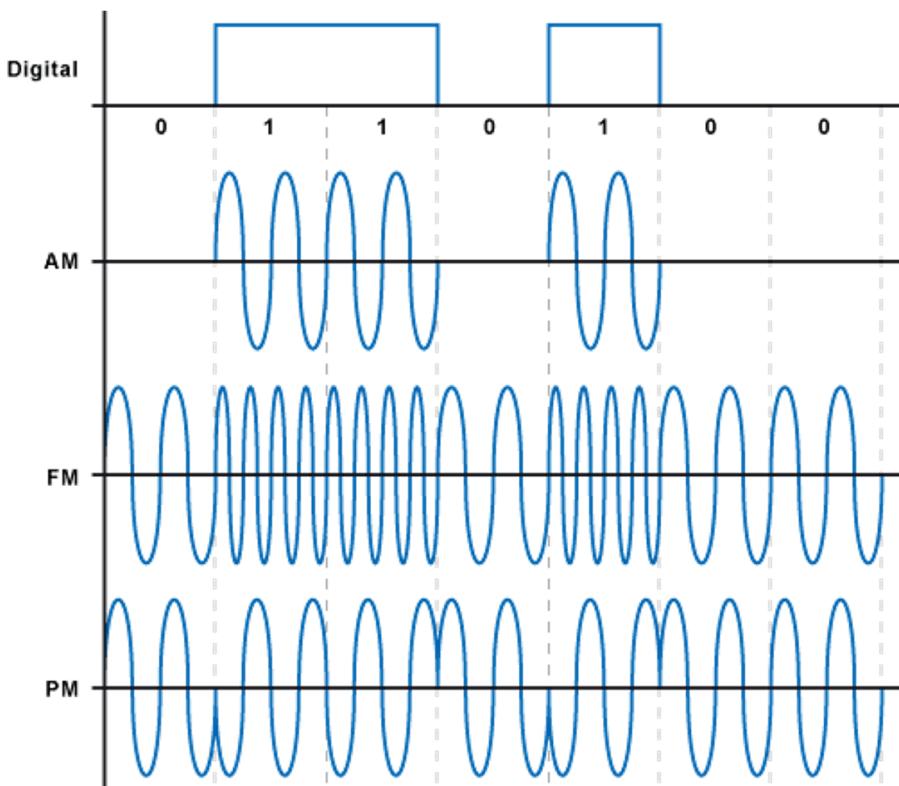


(b) Frequenz-Modulation

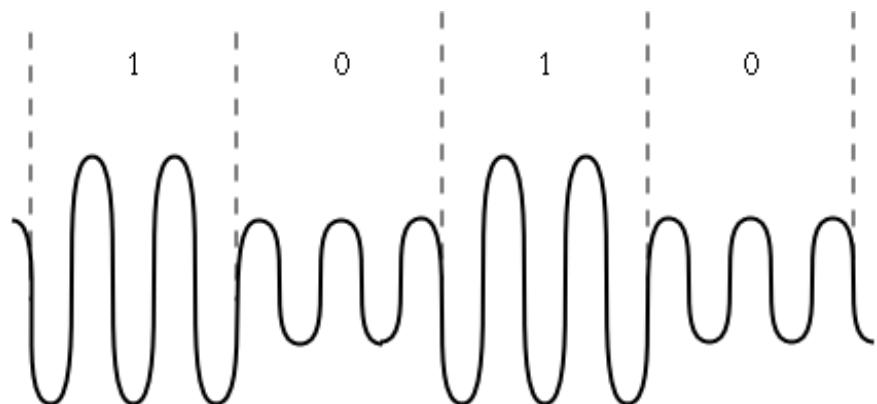


(c) Phasen-Modulation





AM:



31. Wie viele Bits pro Signalschritt werden übertragen, wenn 16 Signalzustände (Symbole) unterscheidbar sind? Wie verhalten sich somit Bit- und Baudrahte?

32. Gegeben seien die *obere* Grenzfrequenz  $f_o = 2,1 \text{ GHz}$  sowie die *untere* Grenzfrequenz  $f_u = 2,8 \text{ GHz}$  eines idealen Mediums.

(a) Wie hoch ist die Bandbreite des Mediums?

(b) Wie hoch muss  $f_{\text{Abtast}}$  gemäß dem Abtasttheorem von Shannon und Nyquist für zeitdiskrete Signale sein?

(a) Betragsdifferenz von  $f_o$  und  $f_u$ :  $|2,1 \text{ GHz} - 2,8 \text{ GHz}| = 0,7 \text{ GHz}$

# Wireshark

## 2019 Klausur

### II. Wireshark

Gegeben sei folgender Mitschnitt eines Netz-Interfaces, der durch das Programm Wireshark erzeugt wurde.

#	Quelle	Ziel	Protokoll	Länge	Info
1	192.168.217.23	141.84.218.30	DNS	82	Standard query 0xcd27 AAAA www.gnu.org OPT
2	192.168.217.23	141.84.218.30	DNS	82	Standard query 0x281d A www.gnu.org OPT
3	141.84.218.30	192.168.217.23	DNS	135	Standard query response 0xcd27 AAAA www.gnu.org CNAME wildebeest.gnu.org AAAA 2001:470:142:3::5 OPT
4	141.84.218.30	192.168.217.23	DNS	123	Standard query response 0x281d A www.gnu.org CNAME wildebeest.gnu.org A 209.51.188.148 OPT
5	192.168.217.23	209.51.188.148	TCP	74	39712 > 80 [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM=1 TSval=2716571269 TSecr=0 WS=128
6	209.51.188.148	192.168.217.23	TCP	74	80 > 39712 [SYN ACK] Seq=0 Ack=1 Win=28960 Len=0 MSS=1460 SACK_PERM=1 TSval=485785247 TSecr=2716571269 WS=128
7	192.168.217.23	209.51.188.148	TCP	66	39712 > 80 [ACK] Seq=1 Ack=1 Win=29312 Len=0 TSval=2716571363 TSecr=485785247
8	192.168.217.23	209.51.188.148	HTTP	142	HEAD / HTTP/1.1
9	209.51.188.148	192.168.217.23	TCP	66	80 > 39712 [ACK] Seq=1 Ack=77 Win=29056 Len=0 TSval=485785271 TSecr=2716571364
10	209.51.188.148	192.168.217.23	HTTP	407	HTTP/1.1 200 OK
11	192.168.217.23	209.51.188.148	TCP	66	39712 > 80 [ACK] Seq=77 Ack=342 Win=30336 Len=0 TSval=2716571465 TSecr=485785273

7. Beantworten Sie dazu die folgenden Fragen:

- (a) Welche Protokolle der Anwendungsschicht gemäß dem Internetmodell sind in dem Mitschnitt (2) enthalten?

- (b) Welche PDUs waren am TCP drei-Wege-Handschlag beteiligt? (1)

- (c) Über welches Protokoll der Schicht 4 wurde PDU 1 übertragen? (1)

- (d) Wie lautet die IPv6-Adresse des Hosts `www.gnu.org`? (1)

- (e) Über welches Protokoll der Schicht 3 (inklusive Version) findet die HTTP-Anfrage und -Antwort statt? (1)

- (f) An welcher PDU kann man erkennen, dass die HTTP-Anfrage erfolgreich war? (1)

- (g) Wie viele Bytes enthält die HTTP-PDU der Antwort (#10)? Hinweis: die PDU wurde per Ethernet übertragen. Eine Ethernet-PCI ist 14 Bytes lang. Im TCP-Header sind 12 Byte für Optionen genutzt. (1)

- (a) DNS und HTTP. TCP nicht, da nicht Teil der Anwendungsschicht
- (b) 5, 6 und 7. Siehe [SYN], [SYN ACK] und [ACK] -> 3 Way Handshake
- (c) (vermutlich) UDP
- (d) 2001:470:142:3::5
- (e) IPv4
- (f) 10
- (g)  $407 - 14 - 12 = 381$  Byte

## 2022 Übung

### **3-Way-Handshake und Sequenznummern bei TCP (H)**

Protokollkonzepte wie *3-Way-Handshaking* und *Sequenznummern* sind Mechanismen für das Verbindungsmanagement und für die zuverlässige Kommunikation. Diese Mechanismen werden z.B. im *Transmission Control Protocol (TCP)* eingesetzt. Zur Bearbeitung dieser Aufgabe wird die Trace-Datei **trace3.pcap** bereitgestellt, die mitgeschnittenen TCP-Verkehr enthält.

Die Datei lässt sich z.B. mit dem freien Programm Wireshark<sup>1</sup> öffnen, das den mitgeschnittenen TCP-Verkehr grafisch aufbereitet anzeigen und filtern kann.

Wireshark stellt die PDUs verschiedener Schichten tabellarisch dar. In dieser Aufgabe soll es um TCP-PDUs (Segmente) gehen. Sie sind daran zu erkennen, dass in der *Protocol*-Spalte TCP steht. Die Informationen der TCP-PCI werden von Wireshark übersichtlich aufbereitet und je Segment dargestellt.

- (a) Identifizieren Sie die zum 3-Way-Handshaking Vorgang gehörenden Segmente in **trace3.pcap**.
- (b) Identifizieren Sie die zum Verbindungsabbau gehörigen Segmente.
- (c) Berechnen Sie aus den Paketen des 3-Way-Handshake das so genannte *Round Trip Delay (RTD)*. Das ist die Zeit, die vom Versenden eines Segments bis zum Erhalt einer Antwort vergeht.
- (d) Welche absoluten und relativen TCP-Sequenznummern besitzen diese Segmente?
- (e) In dem Mitschnitt werden (in der Standardkonfiguration von Wireshark) auch PDUs vom Protokoll der Anwendungsschicht (*SSHv2*) angezeigt. Nutzt dieses Protokoll auch TCP? Begründen Sie Ihre Vermutung kurz.

## Aufgabe 1: 3-Way-Handshake und Sequenznummern bei TCP

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	192.168.0.101	128.153.4.131	TCP	62	1226 - 22 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1
2	0.049886	128.153.4.131	192.168.0.101	TCP	60	22 - 1226 [SYN, ACK] Seq=0 Ack=1 Win=1460 Len=0 MSS=1460
3	0.049935	192.168.0.101	128.153.4.131	TCP	54	1226 - 22 [ACK] Seq=1 Ack=1 Win=64240 Len=0 MSS=1460
4	0.105279	128.153.4.131	192.168.0.101	SSHv2	79	Server: Protocol (SSH-1.99-OpenSSH_3.6.1p2)
5	0.106616	192.168.0.101	128.153.4.131	SSHv2	83	Client: Protocol (SSH-1.99-2.0.13 F-SECURE SSH)
6	0.156893	128.153.4.131	192.168.0.101	TCP	60	22 - 1226 [ACK] Seq=26 Ack=30 Win=8760 Len=0
7	0.156940	192.168.0.101	128.153.4.131	SSHv2	494	Client: Key Exchange Init, Diffie-Hellman Key Exchange Init
8	0.156943	128.153.4.131	192.168.0.101	SSHv2	599	Server: Key Exchange Init
9	0.294668	192.168.0.101	128.153.4.131	TCP	54	1226 - 22 [ACK] Seq=470 Ack=579 Win=63671 Len=0
10	0.308588	128.153.4.131	192.168.0.101	TCP	60	22 - 1226 [ACK] Seq=579 Ack=470 Win=8760 Len=0
11	0.391539	128.153.4.131	192.168.0.101	SSHv2	702	Server: Diffie-Hellman Key Exchange Reply, New Keys
12	0.391765	192.168.0.101	128.153.4.131	SSHv2	76	Client: New Keys
13	0.628438	128.153.4.131	192.168.0.101	TCP	60	22 - 1226 [ACK] Seq=1218 Ack=486 Win=8760 Len=0
14	0.628487	192.168.0.101	128.153.4.131	SSHv2	106	Client: Encrypted packet (len=52)
15	0.790815	128.153.4.131	192.168.0.101	SSHv2	106	Server: Encrypted packet (len=52)
16	0.791399	192.168.0.101	128.153.4.131	SSHv2	122	Client: Encrypted packet (len=68)
17	0.802927	128.153.4.131	192.168.0.101	SSHv2	62	Server: Encrypted packet (len=578)
18	0.861134	192.168.0.101	128.153.4.131	SSHv2	138	Client: Encrypted packet (len=84)
19	0.985678	128.153.4.131	192.168.0.101	SSHv2	90	Server: Encrypted packet (len=36)

Time: Zeitstempel seit dem ersten Paket, wie viel Zeit ist vergangen

Source: Von welchem PC/IP wurde das Paket losgeschickt

Destination: Welche PC/IP hat das Paket empfangen

TCP: Das höchste Protokoll/die höchste Schicht, die repräsentiert

Length: Länge in Bytes

a)

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	192.168.0.101	128.153.4.131	TCP	62	1226 - 22 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1
2	0.049886	128.153.4.131	192.168.0.101	TCP	60	22 - 1226 [SYN, ACK] Seq=0 Ack=1 Win=1460 Len=0 MSS=1460
3	0.049935	192.168.0.101	128.153.4.131	TCP	54	1226 - 22 [ACK] Seq=1 Ack=1 Win=64240 Len=0
4	0.105279	128.153.4.131	192.168.0.101	SSHv2	79	Server: Protocol (SSH-1.99-OpenSSH_3.6.1p2)

Die ersten 3 Pakete.

Verbindungsaufbau

b)

169 25.9844884	128.153.4.131	192.168.0.101	SSHv2	238	Server: Encrypted packet (len=184)
170 25.967164	192.168.0.101	128.153.4.131	SSHv2	90	Client: Encrypted packet (len=36)
171 25.968266	192.168.0.101	128.153.4.131	TCP	54	1226 - 22 [FIN, ACK] Seq=2974 Ack=5634 Win=63496 Len=0
172 26.024317	128.153.4.131	192.168.0.101	TCP	60	22 - 1226 [ACK] Seq=5634 Ack=2975 Win=8760 Len=0
173 26.031474	128.153.4.131	192.168.0.101	TCP	60	22 - 1226 [FIN, ACK] Seq=5634 Ack=2975 Win=8760 Len=0
174 26.031492	192.168.0.101	128.153.4.131	TCP	54	1226 - 22 [ACK] Seq=2975 Ack=5635 Win=63496 Len=0

Die letzten 3 Pakete.

Verbindungsabbau

c)

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	192.168.0.101	128.153.4.131	TCP	62	1226 - 22 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1
2	0.049886	128.153.4.131	192.168.0.101	TCP	60	22 - 1226 [SYN, ACK] Seq=0 Ack=1 Win=1460 Len=0 MSS=1460
3	0.049935	192.168.0.101	128.153.4.131	TCP	54	1226 - 22 [ACK] Seq=1 Ack=1 Win=64240 Len=0
4	0.105279	128.153.4.131	192.168.0.101	SSHv2	79	Server: Protocol (SSH-1.99-OpenSSH_3.6.1p2)

Die Zeit die vergeht, wenn ein Paket hin und her geschickt wird.

Sogenannte Round Trip Delay.

d)

1 0.000000	192.168.0.101	128.153.4.131	TCP	02 1220 → 22 [SYN] Seq=0
2 0.049886	128.153.4.131	192.168.0.101	TCP	60 22 → 1226 [SYN, ACK] S
3 0.049935	192.168.0.101	128.153.4.131	TCP	54 1226 → 22 [ACK] Seq=1
4 0.105279	128.153.4.131	192.168.0.101	SSHv2	79 Server: Protocol (SSH-Init)
5 0.106616	192.168.0.101	128.153.4.131	SSHv2	83 Client: Protocol (SSH-Init)
6 0.156893	128.153.4.131	192.168.0.101	TCP	60 22 → 1226 [ACK] Seq=26
7 0.156946	192.168.0.101	128.153.4.131	SSHv2	494 Client: Key Exchange Init
8 0.169043	128.153.4.131	192.168.0.101	SSHv2	598 Server: Key Exchange Init
9 0.294668	192.168.0.101	128.153.4.131	TCP	54 1226 → 22 [ACK] Seq=47
10 0.308588	128.153.4.131	192.168.0.101	TCP	60 22 → 1226 [ACK] Seq=57
11 0.391539	128.153.4.131	192.168.0.101	SSHv2	702 Server: Diffie-Hellman
12 0.467653	192.168.0.101	128.153.4.131	SSHv2	70 Client: New Keys
13 0.628438	128.153.4.131	192.168.0.101	TCP	60 22 → 1226 [ACK] Seq=12
14 0.628487	192.168.0.101	128.153.4.131	SSHv2	106 Client: Encrypted packet
15 0.700815	128.153.4.131	192.168.0.101	SSHv2	106 Server: Encrypted packet
16 0.701399	192.168.0.101	128.153.4.131	SSHv2	122 Client: Encrypted packet
17 0.860627	128.153.4.131	192.168.0.101	SSHv2	630 Server: Encrypted packet
18 0.861134	192.168.0.101	128.153.4.131	SSHv2	138 Client: Encrypted packet
19 0.985678	128.153.4.131	192.168.0.101	SSHv2	90 Server: Encrypted packet

Source Port: 22  
 Destination Port: 1226  
 [Stream index: 0]  
 [TCP Segment Len: 0]  
 Sequence number: 1218 (relative sequence number)  
 Sequence number (raw): 1356605985  
 [Next sequence number: 1218 (relative sequence number)]  
 Acknowledgment number: 486 (relative ack number)  
 Acknowledgment number (raw): 3332460608  
 0101 = Header length: 20 bytes (5)

### Relative und absolute Sequenznummer

e)

4 0.185279	128.153.4.131	192.168.0.101	SSHv2	79 Server: Protocol (SSH-1.99-OpenSSH_3.6.1)
5 0.186616	192.168.0.101	128.153.4.131	SSHv2	83 Client: Protocol (SSH-1.99-2.0.13 F-)
6 0.156893	128.153.4.131	192.168.0.101	TCP	60 22 → 1226 [ACK] Seq=26 Ack=30 Win=87
7 0.156946	192.168.0.101	128.153.4.131	SSHv2	494 Client: Key Exchange Init, Diffie-Hellman
8 0.169043	128.153.4.131	192.168.0.101	SSHv2	598 Server: Key Exchange Init
9 0.294668	192.168.0.101	128.153.4.131	TCP	54 1226 → 22 [ACK] Seq=470 Ack=579 Win=87
10 0.308588	128.153.4.131	192.168.0.101	TCP	60 22 → 1226 [ACK] Seq=570 Ack=470 Win=87
11 0.391539	128.153.4.131	192.168.0.101	SSHv2	702 Server: Diffie-Hellman Key Exchange
12 0.467653	192.168.0.101	128.153.4.131	SSHv2	70 Client: New Keys
13 0.628438	128.153.4.131	192.168.0.101	TCP	60 22 → 1226 [ACK] Seq=1218 Ack=486 Win=87
14 0.628487	192.168.0.101	128.153.4.131	SSHv2	106 Client: Encrypted packet (len=52)
15 0.700815	128.153.4.131	192.168.0.101	SSHv2	106 Server: Encrypted packet (len=52)
16 0.701399	192.168.0.101	128.153.4.131	SSHv2	122 Client: Encrypted packet (len=68)
17 0.860627	128.153.4.131	192.168.0.101	SSHv2	630 Server: Encrypted packet (len=576)
18 0.861134	192.168.0.101	128.153.4.131	SSHv2	138 Client: Encrypted packet (len=84)
19 0.985678	128.153.4.131	192.168.0.101	SSHv2	90 Server: Encrypted packet (len=36)

> Frame 4: 79 bytes on wire (632 bits), 79 bytes captured (632 bits)  
 > Ethernet II, Src: LinksysG\_8d:be:id (00:06:25:8d:be:id), Dst: Intel\_53:87:d9 (00:07:e9:53:87:d9)  
 > Internet Protocol Version 4, Src: 128.153.4.131, Dst: 192.168.0.101  
 > Transmission Control Protocol, Src Port: 22, Dst Port: 1226, Seq: 1, Ack: 1, Len: 25  
 > SSH Protocol

Oberste Schicht bei dem Paket ist SSH Protokoll und darunter liegt SSH Protokoll.

Ja, es liegt immer das TCP Zugrunde.

### Aufgabe 3: Requests und Response

a)

```
GET /gnu/gnu.html HTTP/1.1<cr><lf>Host: www.gnu.org<cr><lf>User-Agent:  
Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:67.0) Gecko/20100101  
Firefox/67.0<cr><lf>Accept: text/html,application/xhtml+xml,  
application/xml;q=0.9,*/*;q=0.8<cr><lf>Accept-Language: de-DE,en-US;  
q=0.7,en;q=0.3<cr><lf>Accept-Encoding: gzip, deflate, br<cr><lf>  
Connection: keep-alive<cr><lf><cr><lf>
```

b)

```
GET /gnu/gnu.html HTTP/1.1<cr><lf>Host: www.gnu.org<cr><lf>User-Agent:  
Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:67.0) Gecko/20100101  
Firefox/67.0<cr><lf>Accept: text/html,application/xhtml+xml,  
application/xml;q=0.9,*/*;q=0.8<cr><lf>Accept-Language: de-DE,en-US;  
q=0.7,en;q=0.3<cr><lf>Accept-Encoding: gzip, deflate, br<cr><lf>  
Connection: keep-alive<cr><lf><cr><lf>
```

c)

```
GET /gnu/gnu.html HTTP/1.1<cr><lf>Host: www.gnu.org<cr><lf>User-Agent:  
Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:67.0) Gecko/20100101  
Firefox/67.0<cr><lf>Accept: text/html,application/xhtml+xml,  
application/xml;q=0.9,*/*;q=0.8<cr><lf>Accept-Language: de-DE,en-US;  
q=0.7,en;q=0.3<cr><lf>Accept-Encoding: gzip, deflate, br<cr><lf>  
Connection: keep-alive<cr><lf><cr><lf>
```

d)

```
GET /gnu/gnu.html HTTP/1.1<cr><lf>Host: www.gnu.org<cr><lf>User-Agent:  
Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:67.0) Gecko/20100101  
Firefox/67.0<cr><lf>Accept: text/html,application/xhtml+xml,  
application/xml;q=0.9,*/*;q=0.8<cr><lf>Accept-Language: de-DE,en-US;  
q=0.7,en;q=0.3<cr><lf>Accept-Encoding: gzip, deflate, br<cr><lf>  
Connection: keep-alive<cr><lf><cr><lf>
```

Nicht erkennbar. TCP-Verbindung besteht bereits.

e)

```
GET /gnu/gnu.html HTTP/1.1<cr><lf>Host: www.gnu.org<cr><lf>User-Agent:  
Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:67.0) Gecko/20100101  
Firefox/67.0<cr><lf>Accept: text/html,application/xhtml+xml,  
application/xml;q=0.9,*/*;q=0.8<cr><lf>Accept-Language: de-DE,en-US;  
q=0.7,en;q=0.3<cr><lf>Accept-Encoding: gzip, deflate, br<cr><lf>  
Connection: keep-alive<cr><lf><cr><lf>
```

Firefox 67 - Wird übermittelt damit Server Antwort für Browser anpassen kann. Der User-Agent Header ist nicht notwendig.

f)

```
HTTP/1.1 200 OK<cr><lf>Date: Thu, 23 May 2019 08:27:34 GMT<cr><lf>
Server: Apache/2.4.7<cr><lf>Content-Location: gnu.html<cr><lf>Accept
-Ranges: bytes<cr><lf>Content-Encoding: gzip<cr><lf>Content-Length:
5751<cr><lf>Keep-Alive: timeout=3, max=98<cr><lf>Connection: Keep-
Alive<cr><lf>Content-Type: text/html<cr><lf>Content-Language: en<cr
><lf><cr><lf><!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN
"<cr><lf>      "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd"><cr
><lf><html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="
en"><cr><lf><cr><lf><head><cr><lf><!-- start of server/head-include
-1.html -->
<... weitere Zeichen der Antwort wurden entfernt...>
```

Ja, da HTTP-Status-Code: 200

Antwort wurde am 23.Mai 2019 um 8:27 Uhr Greenwich Mean Time erstellt.

g)

```
HTTP/1.1 200 OK<cr><lf>Date: Thu, 23 May 2019 08:27:34 GMT<cr><lf>
Server: Apache/2.4.7<cr><lf>Content-Location: gnu.html<cr><lf>Accept
-Ranges: bytes<cr><lf>Content-Encoding: gzip<cr><lf>Content-Length:
5751<cr><lf>Keep-Alive: timeout=3, max=98<cr><lf>Connection: Keep-
Alive<cr><lf>Content-Type: text/html<cr><lf>Content-Language: en<cr
><lf><cr><lf><!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN
"<cr><lf>      "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd"><cr
><lf><html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="
en"><cr><lf><cr><lf><head><cr><lf><!-- start of server/head-include
-1.html -->
<... weitere Zeichen der Antwort wurden entfernt...>
```

Englisch

zu f): 4xx und 5xx sind fehlerhaft

h)

```
HTTP/1.1 200 OK<cr><lf>Date: Thu, 23 May 2019 08:27:34 GMT<cr><lf>
  Server: Apache/2.4.7<cr><lf>Content-Location: gnu.html<cr><lf>Accept
  -Ranges: bytes<cr><lf>Content-Encoding: gzip<cr><lf>Content-Length:
  5751<cr><lf>Keep-Alive: timeout=3, max=98<cr><lf>Connection: Keep-
  Alive<cr><lf>Content-Type: text/html<cr><lf>Content-Language: en<cr
  ><lf><cr><lf><!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN
  "<cr><lf>      "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd"><cr
  ><lf><html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en"><cr><lf><head><cr><lf><!-- start of server/head-include
  -1.html -->
<... weitere Zeichen der Antwort wurden entfernt...>
```

5751 Bytes

i)

```
HTTP/1.1 200 OK<cr><lf>Date: Thu, 23 May 2019 08:27:34 GMT<cr><lf>
  Server: Apache/2.4.7<cr><lf>Content-Location: gnu.html<cr><lf>Accept
  -Ranges: bytes<cr><lf>Content-Encoding: gzip<cr><lf>Content-Length:
  5751<cr><lf>Keep-Alive: timeout=3, max=98<cr><lf>Connection: Keep-
  Alive<cr><lf>Content-Type: text/html<cr><lf>Content-Language: en<cr
  ><lf><cr><lf><!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN
  "<cr><lf>      "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd"><cr
  ><lf><html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en"><cr><lf><head><cr><lf><!-- start of server/head-include
  -1.html -->
<... weitere Zeichen der Antwort wurden entfernt...>
```

Ersten 5 Bytes als Zeichen: <!DOC

Persistente Verbindung wurde bestätigt, siehe Keep-Alive

## Zusammenspiel verschiedener Protokolle / HTTP Anfragen

### 2013 Klausur

exakt wie 2018/19, nur mit Bridge statt Switch

# 2018 Klausur = 2019 Klausur

## IV. Zusammenspiel verschiedener Protokolle

Gegeben sei das in Abbildung 1 gegebene Rechnernetz, das mehrere Hosts mit einer noch nicht näher definierten Komponente X verbindet.

Auf dem Client wird ein Browser-Programm ausgeführt, das eine Verbindung zu einem Webserver namens www aufbaut, um ein HTML-Dokument abzurufen.

Annahmen:

- der Client kennt die IPv4-Adresse des DNS-Servers
- der Client kennt lediglich den Hostname www, nicht dessen IPv4-Adresse
- der DNS-Server kennt alle Hostnamen und die zugehörigen IPv4-Adressen

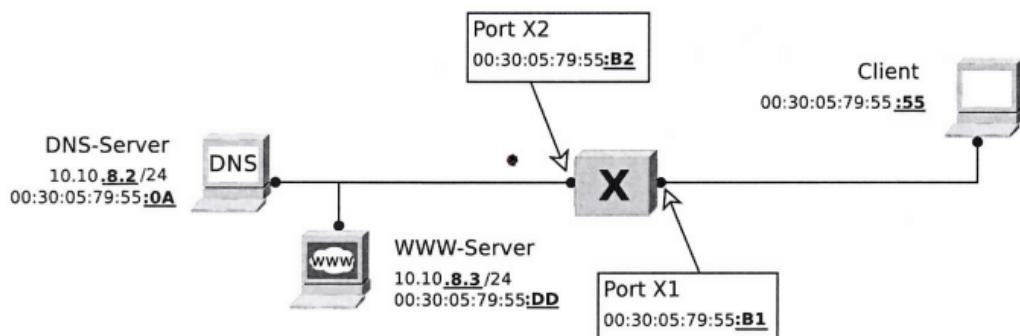


Abbildung 1: Netztopologie

10. Gehen Sie im Folgenden davon aus, dass Komponente X ein **Switch** und die IPv4-Adresse des Clients 10.10.8.4 mit 24-Bit langer Netz-ID ist.
- (a) An welche MAC-Adresse sendet der Client Rahmen, die DNS-Anfragen enthalten? (1)
- 
- (b) An welche IPv4-Adresse sendet der Client Pakete, die HTTP-Anfragen enthalten? (1)
- 
11. Gehen Sie im Folgenden davon aus, dass Komponente X ein **Router** ist. Die IPv4-Adressen sind 192.168.1.**2**/28 für den Client, 192.168.1.**1**/28 für Port X1 des Routers und 10.10.8.**1**/24 für Port X2 des Routers.
- (a) An welche MAC-Adresse sendet der Client HTTP-Anfragen? (1)
  - (b) An welche IPv4-Adresse sendet der Client DNS-Anfragen? (1)
- 
- (c) An welche MAC-Adresse versendet der Router einen Rahmen, mit der Ziel-IP 10.10.8.3 und dem Ziel-UDP-Port 53 (Standardport für DNS)? (1)
- 
12. Warum benötigen die Ports eines **Router**s IPv4-Adressen um Daten zwischen den Netzen übertragen zu können, die Ports eines **Switches** jedoch nicht? (2)

Fragen:

Unterschied Switch, Router und Bridge bezüglich MAC und IPv4- Adressierung?

Router: harter Cut, neue Verbindung nach außen

Switch / Bridge: Netzwerk verbunden, ein großes Netzwerk -> letzte DEST MAC ist auch die erste, wie IPv4

zu 10.

- (a) MAC DNS: ...:0A
- (b) IPv4 WWW: 10.10.8.3/24

zu 11.

- (a) MAC X1: ...:B1
- (b) IPv4 DNS: 10.10.8.2/24
- (c) MAC WWW Server: DD

zu 12.

Mit den Ports des Routers kommunizieren andere Netze über die jeweilige IPv4 Adresse, während Switches nicht von anderen Netzen adressiert werden können, sondern nur die Netze / Endsysteme, die der Switch verbindet.

## 2022 Übung

### 3. Zusammenspiel von IPv4 und ARP (H)

Abbildung 1 skizziert 3 lokale Netze (Subnetz 1 – 3), die über 2 Router miteinander verbunden sind.

- (a) Weisen Sie den Schnittstellen aller Hosts passende IP-Adressen zu. Verwenden Sie für die jeweiligen Subnetze folgende Adressbereiche.
  - Subnetz 1: 192.168.1.100/24
  - Subnetz 2: 192.168.2.100/24
  - Subnetz 3: 192.168.3.100/24
- (b) Weisen Sie jedem Interface eine eindeutige MAC Adresse zu.
- (c) Angenommen Sie senden ein IP-Paket von *Host E* zu *Host B*. Nehmen Sie dabei an, dass alle ARP Einträge gültig und bereits bekannt sind. Listen Sie alle Zwischenschritte der Übertragung auf. Nennen Sie bei jedem Schritt die Quell-IP und Ziel-IP sowie Quell-MAC und Ziel-MAC.
- (d) Gegeben sei dasselbe Szenario wie in Teilaufgabe c). Nehmen Sie nun an, dass die ARP Tabelle beim Sender *Host E* leer ist.

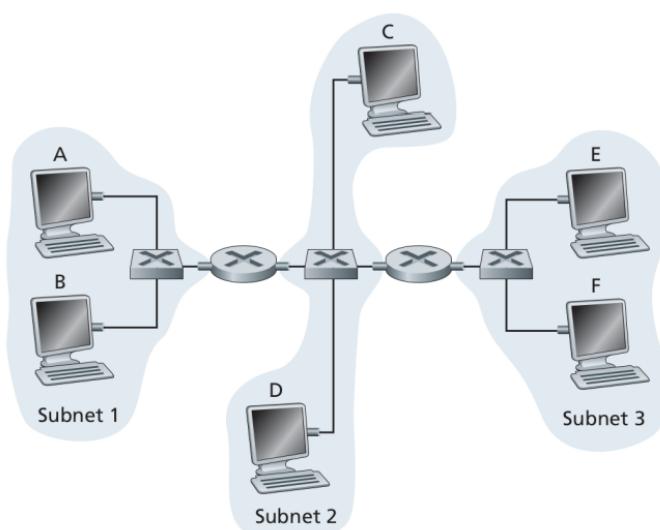
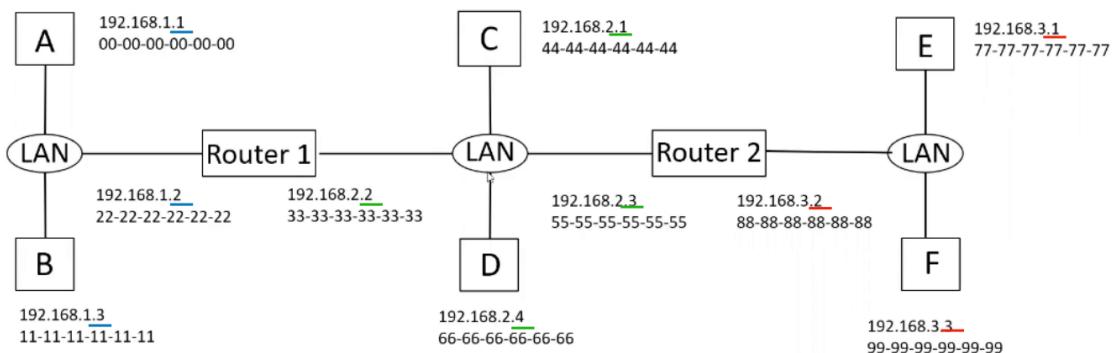


Abbildung 1: 3 Subnetze, verbunden über zwei Router

### Aufgabe 3: Zusammenspiel von IPv4 und ARP

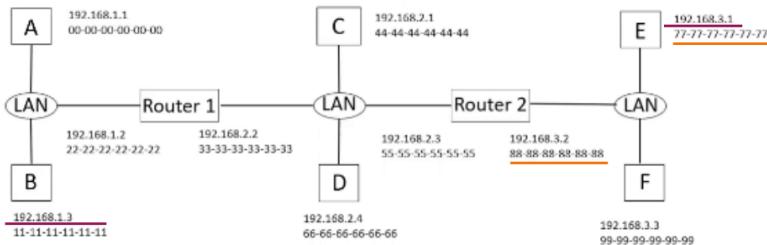
a) und b)



c)

1. Host E -> Router 2

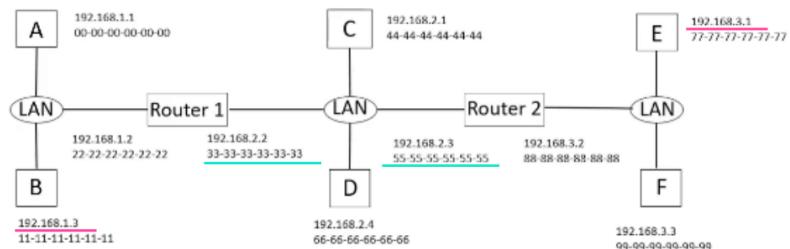
- Source MAC: 77-77-77-77-77-77
- Source IP: 192.168.3.1
- Dest MAC: 88-88-88-88-88-88
- Dest IP: 192.168.1.3



bleiben immer gleich!

2. Router 2 -> Router 1

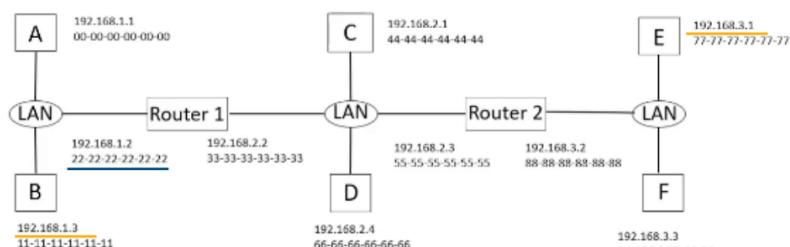
- Source MAC: 55-55-55-55-55-55
- Source IP: 192.168.3.1
- Dest MAC: 33-33-33-33-33-33
- Dest IP: 192.168.1.3



in Reihenfolge wandern wir von dem einen zum anderen!

3. Router 1 -> Host B

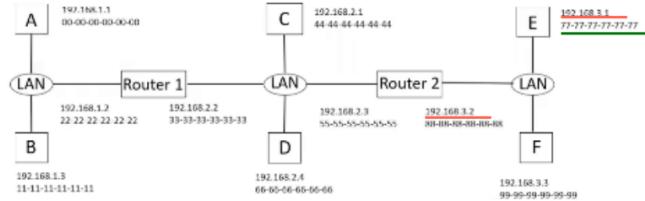
- Source MAC: 22-22-22-22-22-22
- Source IP: 192.168.3.1
- Dest MAC: 11-11-11-11-11-11
- Dest IP: 192.168.1.3



d)

Host E sendet ARP Query (ARP Broadcast):

- Source MAC: 77-77-77-77-77-77
- Source IP: 192.168.3.1
- Dest MAC: FF-FF-FF-FF-FF-FF
- Dest IP: 192.168.3.2



⇒ Router 2 antwortet mit seiner MAC Adresse

⇒ Host E kann sein Paket mit Ziel IP von Host B an Router 2 schicken, der das Paket weitergibt (analog zu c)

Also wenn ARP Tabelle beim Sender leer ist macht man bei Dest MAC: FF...

## Relevante Folien

### Unterschied Rahmen und Paket

#### Vergleichstabelle

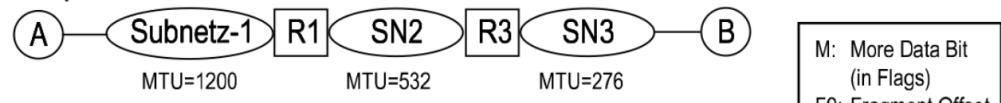
Basis zum Vergleich	Rahmen	Paket
Basic	Frame ist die Dateneinheit des Data Link Layer-Protokolls.	Paket ist die Netzwerkschichtprotokolldateneinheit.
Zugehörige OSI-Schicht	Datenübertragungsebene	Netzwerkschicht
Enthält	Quell- und Ziel-MAC-Adresse.	Quell- und Ziel-IP-Adresse.
Korrelation	Segment ist in einem Paket gekapselt.	Das Paket ist in einem Frame gekapselt.

# Fragmentierung

## IPv4

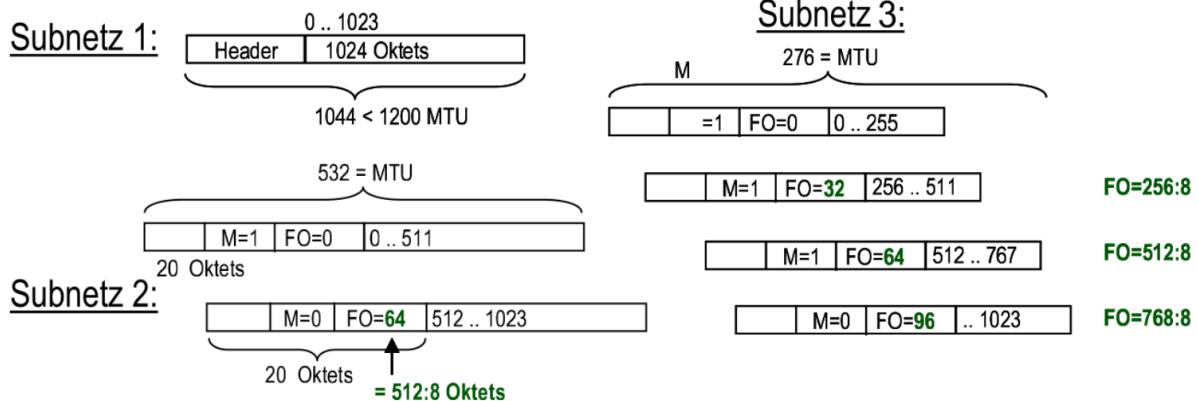
- Maximum Transmission Unit (MTU):
  - Menge an Daten, die ein Frame (auf Schicht 2) übertragen kann
  - MTU setzt obere Grenze für Größe von IP-Paketen
- Problem: jeder Link eines Pfades mit eigener MTU  
→ Pakete werden fragmentiert
- Sender (oder Router am Transitknoten) teilt Paket in Fragmente auf:
  - *Identifizierungsnummer* vom Paket übernommen
  - *Fragmentierungs-Offset* wird je Fragment gesetzt
  - *Flag* = 1, falls weitere Fragmente folgen, sonst 0

## Beispiel



M: More Data Bit (in Flags)  
FO: Fragment Offset

Aufgabe: Nachricht mit 1044 Oktets von A nach B  
IP-Header  $\geq$  20 Oktets



## IPv6

- Fragmentierung von IPv6 Paketen ausschließlich in Endsystemen (nicht im Transitsystem)
- Nicht-Fragmentierbarer Teil:
  - IPv6-Header
  - alle Header mit Ende-zu-Ende Relevanz
- Aufbau von Fragment-Paketen:
  - Nicht-Fragmentierbarer Teil
  - Fragment Header (nächste Folie)
  - Inhalt des ersten Fragments

## MAC-Adressen

# MAC Adressen

- Synonyme: LAN Adresse, physische Adresse
- Keine hierarchische Struktur (im Gegensatz zu IP)
- Länge: 6 Bytes →  $2^{48}$  mögliche MAC Adressen
- MAC Adressen werden „ab Werk“ vergeben und sind daher fest in der Hardware verankert.
- Zuweisung von MAC Adressen an Hersteller erfolgt zentral durch IEEE.
  - Die ersten 24 bit: Prefix für Hersteller (IEEE)
  - Die letzten 24 bit: definierbar vom Hersteller selbst.
- Broadcast MAC Adresse um ein Frame an alle Peers im selben LAN zu senden
  - FF-FF-FF-FF-FF-FF (48 aufeinanderfolgende Einsen)

## TCP

### Fast Retransmit / Fast Recovery

Fast Retransmit: bei **Tahoe und Reno Timeout** (CongWin = 1, neuer Slow Start bis Threshold)

Fast Recovery: bei **Reno 3 (!) Quittungsduplikaten** (CongWin = Threshold \* 0.5 + 3, Congestion Avoidance)

## Header Flags

# TCP Header (PCI) (3/4)

- **NB** (6 Bit): werden nicht benutzt.
- **Flags** (je 1 Bit):
  - URG: *Urgent Pointer* verwendet (Interrupt Data)
  - ACK: *Acknowledge* (Quittungsnummer gültig)
  - PSH: *Push Data* (nicht puffern, sondern sofort zustellen)
  - RST: *Reset* (abgestürzt, Aufbauwunsch abgelehnt, ...)
  - SYN: *Synchronize*
    - SYN=1, ACK=0 Aufbauwunsch
    - SYN=1, ACK=1 Bestätigung
  - FIN: *Finalize* (Abbauwunsch/keine weiteren Daten)

## Protokolle

# TCP Eigenschaften

- TCP-PDU wird „**Segment**“ genannt
- TCP-Ports: Unterstützen Multiplexen von Anwendungen
- TCP-Sockets: zur Adressierung von Anwendungen
- Puffern: Dienstdaten werden zwischengespeichert
- Protokolle die TCP verwenden:
  - HTTP (Hypertext Transfer Protocol)
  - SMTP (Simple Mail Transfer Protocol)
  - IMAP (Internet Message Access Protocol)
  - FTP (File Transfer Protocol)

## Unterschied Verbindungslos und Verbindungsorientiert

### Verbindungsloser Dienst (connectionless service)

- Keine Verbindung zwischen den Entitäten
- Keine Empfangsbestätigung (Quittierung)
- Sender hat keine Information darüber, ob die Nachricht das Ziel erreicht hat
- ICI enthält alle Informationen zum Ziel
- Nachricht sucht selbständig Weg durch das Netz
- Geringer Verwaltungsaufwand → Hohe Leistung  
(im Vergleich zur verbindungsorientierten Kommunikation)

### Verbindungsorientierter Dienst (connection-oriented service)

- Aufbau einer logischen Verbindung (Sitzung)
- Strategien zur Fehlerbehandlung:
  - Kein Verlust von Nachrichten
  - Keine Verdoppelung von Nachrichten
  - Keine Verfälschung der Inhalte von Nachrichten
  - Keine Vertauschung der Reihenfolge mehrere Nachrichten
- Verbindungsmanagement durch Flusssteuerung und Staukontrolle zur Handhabung von Verzögerungen bei der Nachrichtenübertragung