



HOCHSCHULE KONSTANZ TECHNIK, WIRTSCHAFT UND GESTALTUNG (HTWG)  
**Fakultät Informatik**  
Rechner- und Kommunikationsnetze  
Prof. Dr. Dirk Staehle

# **Vorlesung Rechnernetze (AIN 5 – SS 2018)**

## **Klausur (K90)**

**Datum: 11. Juli 2018, 11:00**

**Bearbeitungszeit: 90 Minuten**

**Gesamtpunkte: 100**

**Name:**

**Matrikelnummer:**

**Semester:**

### **Hinweise:**

- Zur Klausur sind alle Unterlagen auf Papier zugelassen. Elektronische Hilfsmittel (Handy, Laptop, Tablet, etc.) außer einfachen Taschenrechnern sind nicht erlaubt.
- Falls Sie während der Klausur einen dringenden Anruf erwarten, bitte vorher anmelden. Ansonsten Handy ausschalten!
- Falls Sie sich gesundheitlich nicht in der Lage fühlen, die Klausur mitzuschreiben, bitte VOR der Klausur melden.
- Auf der Teilnehmerliste unterschreiben.
- Sie erhalten Papierbögen zur Bearbeitung der Klausur. Schreiben Sie ihren Namen auf alle Bögen. Legen Sie am Ende die Klausur in den oder die Bögen und schreiben Sie ihren Namen auf den äußeren Bogen.
- Wenn Sie sich beim Ausfüllen einer Tabelle oder Zeichnung verschrieben haben, können Sie Ersatzkopien erhalten, solange der Vorrat reicht.
- Sie können gerne einzelne Seiten aus der gehefteten Klausur entnehmen, wenn das die Bearbeitung erleichtert.

## Aufgabe 1 [Paketübertragung] (22 Punkte)

Gegeben sei die in Abbildung 1 dargestellte Übertragungsstrecke von einer Quelle Q zu einem Ziel Z, die über drei Router  $R_1$ ,  $R_2$  und  $R_3$  verläuft. Die Link-Kapazitäten sowie die Ausbreitungsverzögerungen der vier Links sind in der Abbildung angegeben. Ebenso können Sie der Grafik die Größe der Output-Buffer für alle Links entnehmen. Die Paketgröße beträgt 400 Bytes. ACKs haben eine Größe von 40 Byte und durchlaufen die Übertragungsstrecke in umgekehrter Richtung.

Hinweis: Geben Sie alle zeitlichen Ergebnisse in Millisekunden an.

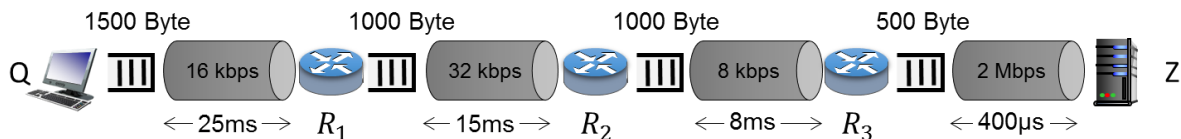


Abbildung 1: Übertragungsstrecke

- 1) Bestimmen Sie die Ende-zu-Ende Übertragungsdauer für ein Paket. (2P)
- 2) Die Quelle versendet alle 300ms zwei Pakete. Bestimmen Sie für jeden Link, welche Pakete verloren gehen. Betrachten Sie dazu die ersten 20 gesendeten Pakete. (8P)
- 3) Wie groß müssten die Puffer mindestens sein, um Paketverluste zu vermeiden? (2P)
- 4) Wie groß müssten Sie das Sendefenster eines Transportprotokolls wählen, um die Kapazität der Übertragungsstrecke vollständig auszunutzen? (2P)
- 5) Auf der Transportschicht wird ein Selective-Repeat-Protokoll mit einem Sendefenster von  $N=15$  Paketen eingesetzt. Der Timeout beträgt 2s.
  - a) Erklären Sie anhand einer Skizze, welches Paket als letztes bei Z ankommt, wenn Q die Übertragung nach 20 Paketen beendet. (5P)
  - b) Bestimmen Sie auch den Zeitpunkt, wann das letzte Paket eintrifft. (3P)

## Aufgabe 2 [HTTP] (18 Punkte)

In dieser Aufgabe wird eine Web-Seite betrachtet, die aus einem 4000 Byte großen HTML-Code und zwanzig Bildern mit einer Größe von 30000 Bytes besteht. Die Requests für den HTML-Code sowie für die Bilder haben eine Größe von 2000 Bytes. Im Browser und auf dem Web-Server läuft die http-Version „Persistent http ohne Pipelining“.

Der Download der Web-Seite erfolgt über eine Übertragungsstrecke, die durch eine Bottleneck-Link-Kapazität von 8 Mbps sowie eine Ende-zu-Ende-Ausbreitungsverzögerung von 8 ms charakterisiert ist. Die Übertragungsverzögerungen für Header sind zu vernachlässigen.

Nehmen Sie für die Aufgabe die folgenden TCP-Parameter an:

- MSS: 1500 Bytes
- Initial Window: 3 MSS
- Retransmission Timeout: 100ms

- 1) Skizzieren Sie den Verlauf der Seitenübertragung beginnend mit dem ersten Request bis zur vollständigen Übertragung des zweiten Bildes. Die Skizze soll die Anzahl und den Inhalt der Segmente enthalten, die pro RTT vom Browser zum Server bzw. vom Server zum Browser gesendet werden. (5P)

Gehen Sie davon aus, dass alle Pakete, die gleichzeitig gesendet werden auch gleichzeitig und in der richtigen Reihenfolge ankommen. Sie können für die Aufgabe die Vorlage in Abbildung 2 verwenden.

- 2) Bestimmen Sie die Page-Load-Time. (3P)
- 3) Skizzieren Sie die Übertragung des zehnten und elften Bildes, wenn das 11te Segment des 10ten Bildes verloren geht. (5P)

Sie können für die Aufgabe die Vorlage in Abbildung 3 verwenden.

- 4) Bestimmen Sie die Zeit, die zur Übertragung der 20 Bilder benötigt wird, wenn vier TCP Verbindungen parallel genutzt werden. (5P)

Name:

Matrikelnummer:

Verwenden Sie folgende Notation:

- Requests:  $MO:Req$ ,  $IO_n:Req$  bzw.  $IO_{n-m}:Req$
- Segmente:  $MO:Sx-y$ ,  $IO_n:Sx-y[k \text{ Bytes}]$  bzw.  $IO_n[k \text{ Bytes}]$   
(Anzahl k Bytes nur, wenn Teile von Segmenten übertragen werden)
- ACKs:  $MO:Ax-y$ ,  $IO_n:Ax-y$ ,  $MO:AReq$ ,  $IO_n:AReq$ , (hier genügt auch die Anzahl)

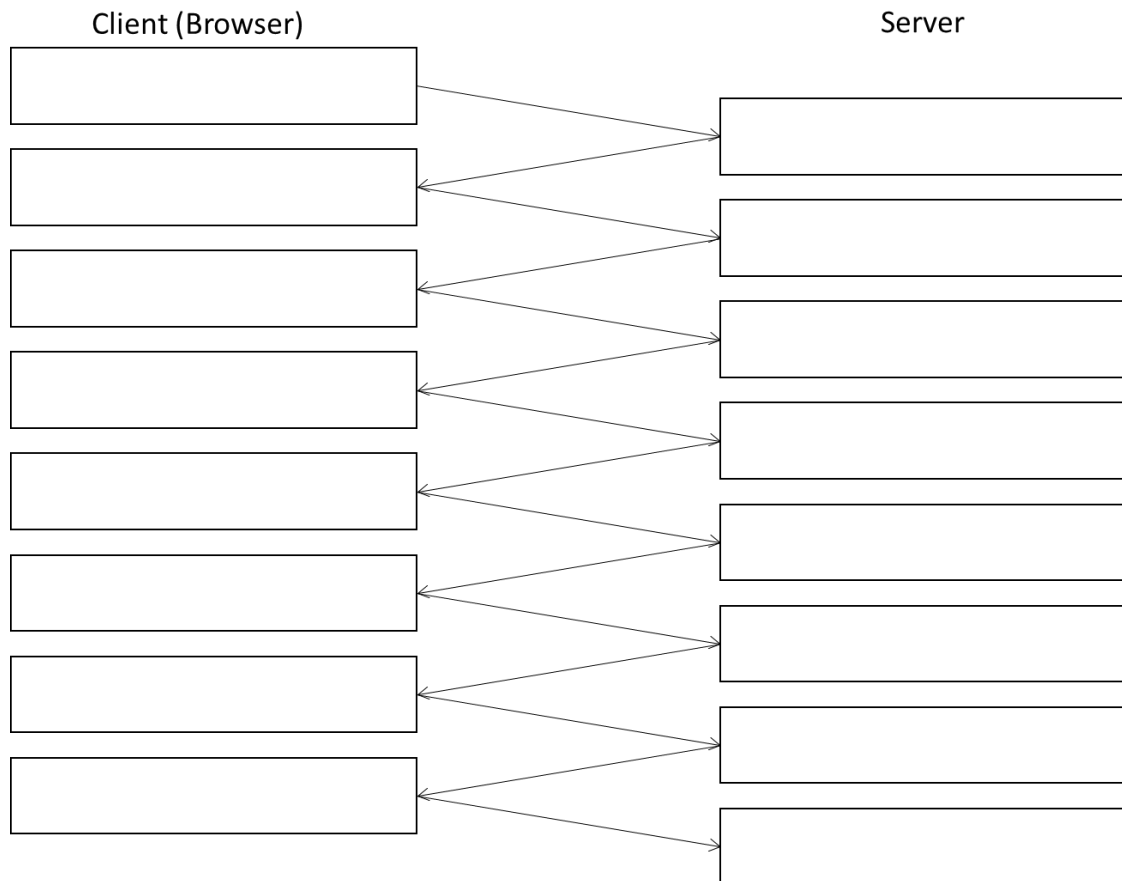


Abbildung 2: Vorlage für Skizze in Aufgabe 2.1)

Name:

Matrikelnummer:

Verwenden Sie folgende Notation:

- Requests:  $MO:Req$ ,  $IO_n:Req$  bzw.  $IO_{n-m}:Req$
- Segmente:  $MO:Sx-y$ ,  $IO_n:Sx-y[k \text{ Bytes}]$  bzw.  $IO_n[k \text{ Bytes}]$   
(Anzahl k Bytes nur, wenn Teile von Segmenten übertragen werden)
- ACKs:  $MO:Ax-y$ ,  $IO_n:Ax-y$ ,  $MO:AReq$ ,  $IO_n:AReq$ , (hier genügt auch die Anzahl)

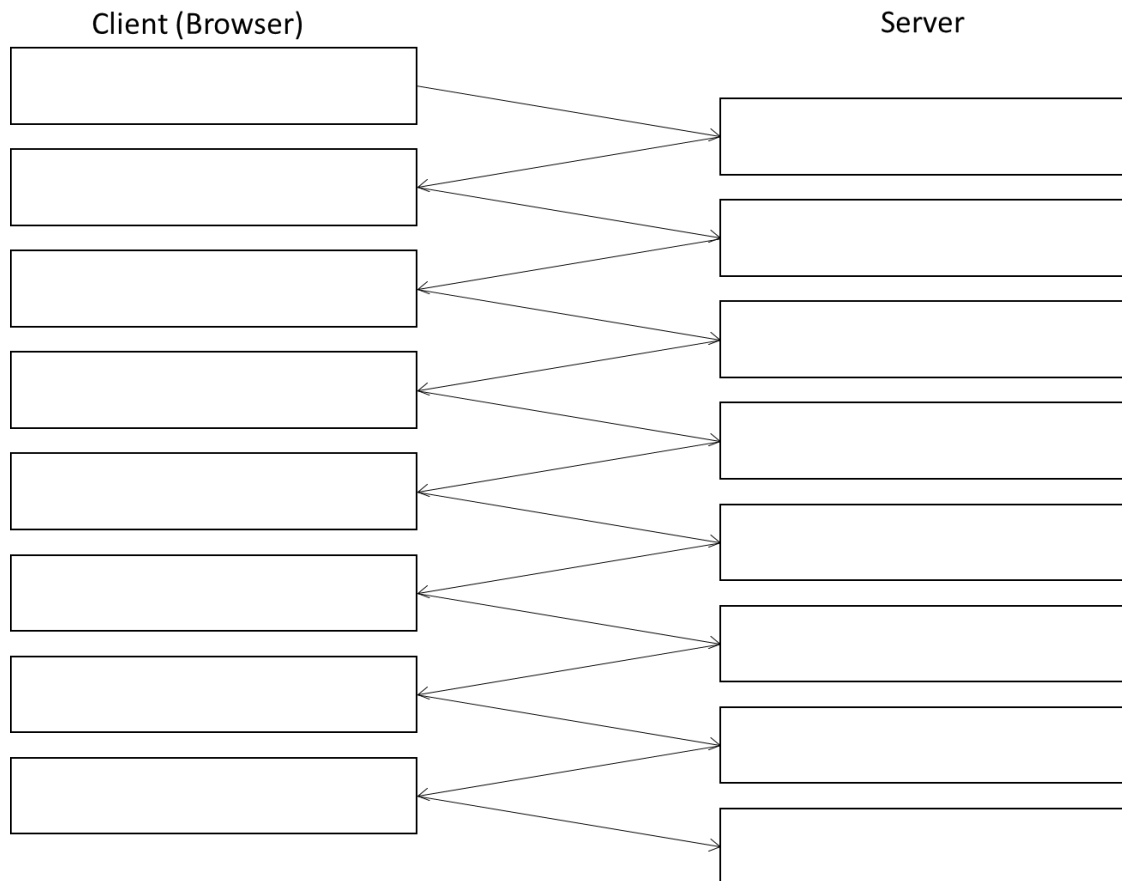


Abbildung 3: Vorlage für Skizze in Aufgabe 2.3)

### Aufgabe 3 [Sockets und Pakete] (10 Punkte)

In **Fehler! Verweisquelle konnte nicht gefunden werden.** und **Fehler! Verweisquelle konnte nicht gefunden werden.** sind die Python-Codes „client.py“ für einen Client und „server.py“ für einen Server gegeben. Zunächst wird der Server auf einem Rechner mit IP Adresse 141.37.168.35 mit dem Befehl `python server.py` gestartet. Eine Sekunde später wird der Client mit dem Befehl `python client.py 141.37.168.35` auf einem Rechner mit IP-Adresse 141.37.168.36 gestartet.

- 1) Erklären Sie kurz, was die Ausführung der beiden Skripte bewirkt. (4P)
- 2) Skizzieren Sie in **Fehler! Verweisquelle konnte nicht gefunden werden.** die Sequenz der ausgetauschten Pakete. Eine exakte Angabe der Zeiten ist nicht notwendig, es kommt lediglich darauf an, welche (Größe und Typ) und wie viele Pakete jeweils gesendet werden. (6P)

Gehen Sie von folgenden Parametern aus:

- Ausbreitungsverzögerung zwischen Client und Server: 20ms
- Bottleneck-Datenrate zwischen Client und Server: 1Gbps
- Maximum Segment Size: 1500 Bytes
- Initial Window: 3 Segmente

Sie können Abbildung 4 als Vorlage für die Skizze verwenden.

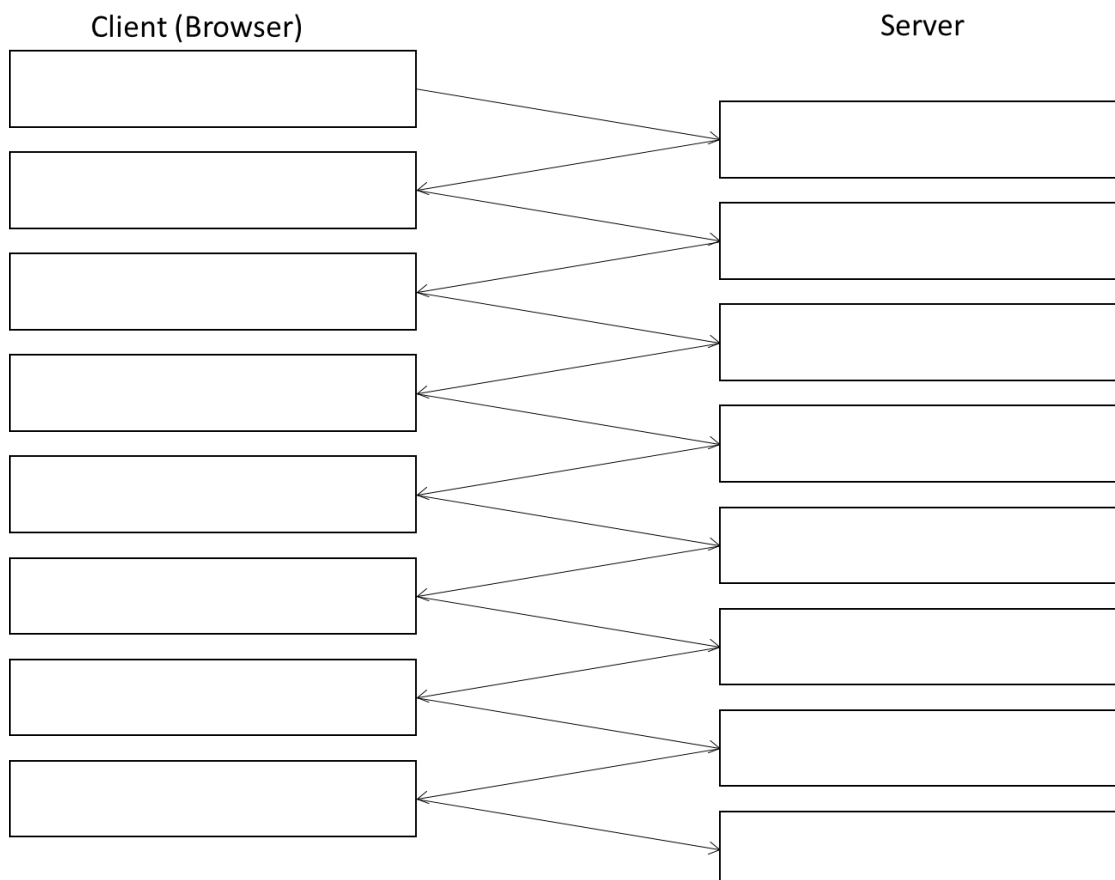


Abbildung 4 Vorlage für Skizze in Aufgabe 3

Name:

Matrikelnummer:

```
1 import socket
2 import sys
3 import time
4
5 socket.setdefaulttimeout(30)
6
7 My_PORT = 50000
8 Remote_PORT=50000
9
10 def main(Remote_IP):
11     sock = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
12     sock.bind(('',My_PORT))
13     try:
14         sock.connect((Remote_IP,Remote_PORT))
15     except socket.timeout:
16         return
17     start_task(sock,5)
18
19 def start_task(sock, num):
20     for i in range(0,num):
21         sock.send('A'.encode('utf-8'))
22         time.sleep(0.02)          # Programm schläft für 20ms
23         sock.send('S'.encode('utf-8'))
24     while True:
25         try:
26             print(sock.recv(1).decode('utf-8'))
27         except socket.timeout:
28             break
29     sock.close()
30
31 if __name__ == '__main__':
32     main(sys.argv[1])
```

**Abbildung 5: Python Code „Client.py“**

Name:

Matrikelnummer:

```
1  import socket
2  import sys
3
4  socket.setdefaulttimeout(30)
5
6  My_PORT = 50000
7  Remote_PORT=50000
8
9  def main(My_IP):
10     sock = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
11     start_server(My_IP)
12
13  def start_task(sock,num):
14     while True:
15         try:
16             c=sock.recv(1).decode('utf-8')
17             if c=='A':
18                 sock.send(bytes(num)) # siehe Hinweis
19             else:
20                 sock.close()
21                 break
22         except socket.timeout:
23             sock.close()
24             break
25
26
27  def start_server(My_IP):
28     sock = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
29     sock.bind((My_IP, My_PORT))
30     sock.listen(1)
31     try:
32         conn, addr = sock.accept()
33         start_task(conn,9000)
34     except socket.timeout:
35         pass
36
37
38  if __name__=='__main__':
39     main(sys.argv[1])
```

#### Abbildung 6: Python Code „Server.py“

##### Hinweis:

- Der Befehl `bytes(n)` liefert ein Byte-Array, das `n` Nullen erhält. Der Befehl ist gleichwertig mit `'0...0'.encode('utf-8')` bzw. `b'0...0'`, wobei in dem String jeweils `n` Nullen stehen.



Name:

Matrikelnummer:

## Aufgabe 4 [TCP] (18 Punkte)

Ein Sender überträgt über eine TCP Verbindung, über die bereits einige Daten übertragen wurden, eine Anzahl von  $N=34$  Segmenten. Die TCP Verbindung und die Übertragungsstrecke sind durch die folgenden Parameter spezifiziert:

- Initial Window=3 Segmente,  $IW=3 \cdot MSS$
- Retransmission Timeout:  $10 \cdot RTT$
- ssthresh wird mit 500 Segmenten initialisiert
- Die RTT wird von der Ausbreitungsverzögerung und nicht der Bottleneck-Bandbreite dominiert. Es kann davon ausgegangen werden, dass alle Segmente eines Sendefensters gleichzeitig aber nacheinander am Empfänger ankommen und auch die daraufhin gesendeten Acknowledgements wieder gleichzeitig aber nacheinander am Sender ankommen.

Protokollieren Sie aus Sicht des Senders den Ablauf der Übertragung, wenn Segmente 11 bis 13 der übertragenen 34 Segmente verloren gehen. In der ersten Zeile von Tabelle 1 ist der aktuelle Status der TCP Verbindung eingetragen. Füllen Sie die Tabelle bis zur Bestätigung der erfolgreichen Übertragung von Segment 34 aus. Sie können auch eine eigene Tabelle anfertigen.

### Tabelle 1 Ablauf einer TCP Verbindung

[illegible]

## Aufgabe 5 [Routingprotokoll] (6 Punkte)

- 1) Das Routingprotokoll EIGRP ist ein Distance-Vector-Protokoll und benutzt den Bellman-Ford-Algorithmus, um die beste Route zu bestimmen. Die zugrundeliegende Default-Routing-Metrik für einen Pfad P ist

$$M(P) = \frac{10Gbps}{\min_{links \ell \in P} bw_{\ell}} + \sum_{links \ell \in P} \frac{delay_{\ell}}{10\mu s}$$

Bestimmen Sie das Routing von A nach C im Beispielnetz in Abbildung 7 nach der EIGRP Default-Metrik. Begründen Sie ihre Antwort. (3P)

- 2) Betrachten Sie nun OSPF mit folgender Linkkosten-Metrik:

$$M(\ell) = \frac{10Gbps}{bw_{\ell}} + \frac{delay_{\ell}}{1ms}$$

Welches Routing erhalten Sie für den Knoten A? Begründen Sie ihre Antwort? (3P)

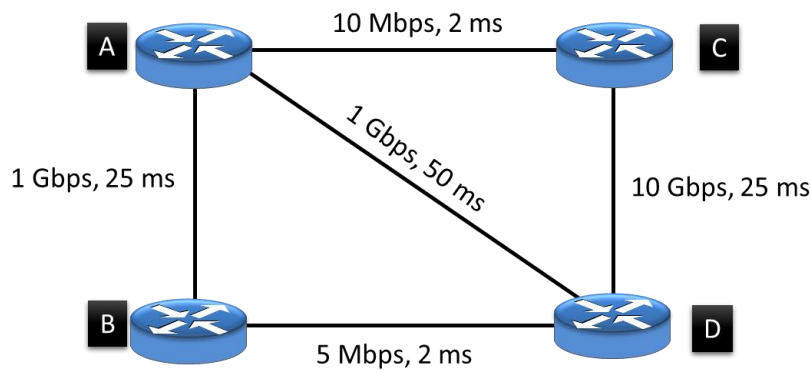


Abbildung 7: Netz für Aufgabe 6

## Aufgabe 6 [IP-Adressen] (16 Punkte)

Betrachten Sie das in Abbildung 8 dargestellte Netz, das aus den Routern  $R_0$  bis  $R_3$ , den Switches  $S_1$  bis  $S_5$  sowie diversen Hosts und Servern besteht. Das Netz ist über den Router  $R_0$  mit dem Internet verbunden.

- 1) Kennzeichnen Sie alle Subnetze/Netzwerksegmente und bestimmen Sie, wie viele Internet-Adressen in dem jeweiligen Subnetz vergeben sind. (4P)
- 2) Für die Hosts C9 und W5 sind bereits die IP Adressen 100.100.200.15 vergeben. Bestimmen Sie die Subnetzmasken für die beiden Rechner, so dass das jeweilige Netzwerksegment möglichst wenige IP-Adressen enthält. (2P)
- 3) Vergeben Sie für den Rest des Netzes IP-Adressen aus dem Adressbereich 100.100.200.0 bis 100.100.200.127 zu. (3P)
- 4) Stellen Sie für den Router  $R_3$  eine Routing-Tabelle auf. Wenn Sie die vorherigen Aufgaben nicht lösen konnten, geben Sie den Netzwerksegmenten Bezeichnungen und verwenden Sie diese in der Routing-Tabelle. (3P)
- 5) Ein Router R habe die in der linken Hälfte von Tabelle 2 dargestellte Routing-Tabelle. Bestimmen Sie für die IP-Adressen auf der rechten Seite der Tabelle, auf welcher Route der Router R ein Datagramm mit dieser Ziel-IP-Adresse weiterleiten würde. (4P)

**Hinweis:** Die Spalte „IP Adressen“ muss nicht ausgefüllt werden, kann aber für „Zwischenergebnisse“ genutzt werden.

Name:

Matrikelnummer:

**Tabelle 2: Routing Tabelle und Liste mit Ziel-IP-Adressen**

Route	Adress-Bereich	Subnetzmaske	IP Adressen	Liste der Ziel IP-Adressen	Route
1	0.0.0.0	0.0.0.0		143.32.0.1	
2	128.0.0.0	192.0.0.0		143.32.64.17	
3	140.0.0.0	252.0.0.0		143.37.168.39	
4	143.0.0.0	255.128.0.0		144.35.56.98	
5	143.0.0.0	255.240.0.0			
6	143.32.0.0	255.224.0.0			
7	143.32.0.0	255.248.0.0			
8	143.32.0.0	255.255.192.0			
9	143.32.8.0	255.255.248.0			
10	143.37.168.0	255.255.248.0			
11	143.37.168.0	255.255.255.224			
12	143.37.168.36	255.255.255.252			
13	143.37.168.39	255.255.255.255			
14	144.0.0.0	255.0.0.0			

Dezimal	Binär	Dezimal	Binär		
128	1000 0000	1	0000 0001	140	1000 1000
192	1100 0000	8	0000 1000	143	1000 1011
224	1110 0000	17	0001 0001	144	1000 1100
240	1111 0000	32	0010 0000	168	1010 1000
248	1111 1000	36	0010 0100		
252	1111 1100	37	0010 0101		
254	1111 1110	39	0010 1001		
255	1111 1111	64	0100 0000		

Name:

Matrikelnummer:

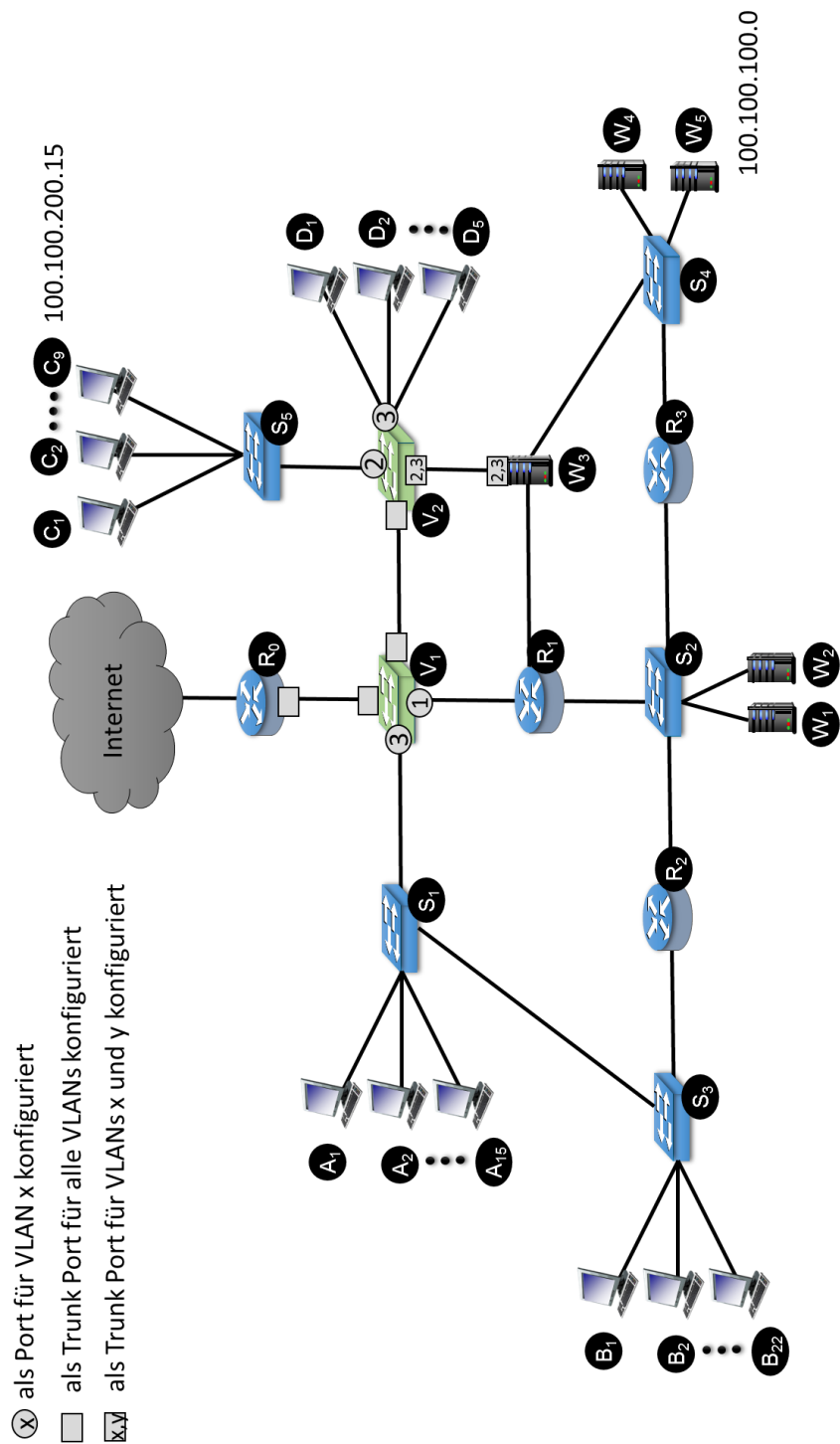


Abbildung 8: Netz mit mehreren Netzwerksegmenten

## Aufgabe 7 [Verständnisfragen] (10 Punkte)

Entscheiden Sie für die folgenden Aussagen, ob sie richtig oder falsch sind, indem Sie sie mit einem R (richtig) oder einem F (Falsch) kennzeichnen. Für korrekte Antworten erhalten Sie einen Punkt. Für falsche Antworten erhalten Sie einen Minuspunkt. Das Gesamtergebnis dieser Aufgabe kann also auch eine negative Punktzahl sein.

1. Autoritative DNS Server sind die einzigen Server in der DNS Hierarchie, die die IP-Adresse zur angefragten URL zurückliefern.
2. Ein Pfad-Vektor-Routing Protokoll hat gegenüber einem einfachen Vektor-Routing-Protokoll den Vorteil, dass Loops vollständig ausgeschlossen werden können.
3. Die verwendete TCP Version wird zwischen Client und Server beim Verbindungsaufbau vereinbart.
4. Das ECN Bit im IP-Header dient zur Differenzierung von Diensten mit Hilfe einer RED-Queue, so dass beispielsweise VoIP-Pakete priorisiert werden können, um im Überlastfall eine gute Sprachqualität zu gewährleisten.
5. Ein ISP exportiert alle ihm bekannten Routen an seine Kunden und Peers.
6. Ein Host A, der sich hinter einem Symmetric NAT befindet, kann einem Host B, der sich ebenfalls hinter einem Symmetric NAT befindet, kein Paket schicken. Es gibt auch keine zuverlässigen Lösungen, um das zu ermöglichen.
7. Zwei Hosts, die sich beide hinter symmetric NATs befinden, können generell nicht miteinander kommunizieren.
8. Zwei Backbone-Router können über einen Interior-Gateway-Router verbunden sein.
9. Wenn das dritt-letzte Segment einer TCP Verbindung verloren geht, gibt es grundsätzlich einen Retransmission-Timeout.
10. Die http-Methode PUT wird eingesetzt, um Formulardaten auf einen Web-Server zu laden, wie beispielsweise bei der Authentisierung durch Benutzername und Passwort.