Networks and Protocols
Jacobs University Bremen
Dr. Jürgen Schönwälder

Course: 320301
Date: 2007-12-12
Duration: 90 minutes

Final Examination

The Jacobs University's Code of Academic Integrity applies to this examination. Please fill in your name (please write readable) and sign below.

Name:

Signature:

This exam is "open book" in the sense that you are allowed to use course slides, papers and your personal notes during the exam. You are not allowed to use any electronic equipment such as calculators, computers, or cell phones.

Please answer the questions on the problem sheets. If you need more space, feel free to write on the back of the pages. Please keep the papers stapled. Write with a pen, do not use pencils.

Problem	Max. Points	Act. Points	Grader
F.1	10		
F.2	20		
F.3	20		
F.4	20		
F.5	20		
F.6	10		
Total	100		

Problem F.1: *Simple Mail Transfer Protocol* (2+2+2+2+2 = 10 points)

Indicate which of the following statements are correct or incorrect by marking the appropriate boxes. For every correctly marked box, you will earn two points. For every incorrectly marked box, you will loose one point. Statements which are not marked or which are marked as true and false will be ignored. The minimum number of points you can achieve is zero.

true	false	
☐	☐	The SMTP protocol supports persistent connections.
☐	☐	A positive reply code for a `DATA` command resets the state of SMTP dialogue and transfers the responsibility for the message from the sending SMTP server to the receiving SMTP server.
☐	☐	An SMTP server has to verify that the `From:` message header line matches the `MAIL FROM:` argument.
☐	☐	The SIEVE mail filtering language implemented on an SMTP server can be used to reject messages by failing the SMTP transaction.
☐	☐	The IMAP protocol supports pipelining of commands.

Solution:

true	false	
☒	☐	The SMTP protocol supports persistent connections.
☒	☐	A positive reply code for a `DATA` command resets the state of SMTP dialogue and transfers the responsibility for the message from the sending SMTP server to the receiving SMTP server.
☐	☒	An SMTP server has to verify that the `From:` message header line matches the `MAIL FROM:` argument.
☒	☐	The SIEVE mail filtering language implemented on an SMTP server can be used to reject messages by failing the SMTP transaction.
☒	☐	The IMAP protocol supports pipelining of commands.

Problem F.2: *Multipurpose Internet Mail Extension* (10+10 = 20 points)

a) What is the base64 encoding of the four character ASCII message `help`?

b) Explain the meaning of the following MIME header:

```
MIME-Version: 1.0
Content-type: multipart/alternative; boundary="NextPart"
```

Show an example MIME message using this MIME header which carries the message `hello` in plain text and HTML text.

Solution:

a) Transformation of the four character ASCII message `help`:

```
ascii        :  h        e        l        p
hexadecimal  :  68       65       6c       70
binary       :  01101000 01100101 01101100 01110000
6bit binary  :  011010 000110 010101 101100 011100 000000
6bit decimal :  26     6      21     44     28     0
base64       :  a      G      V      s      c      A      ==
```

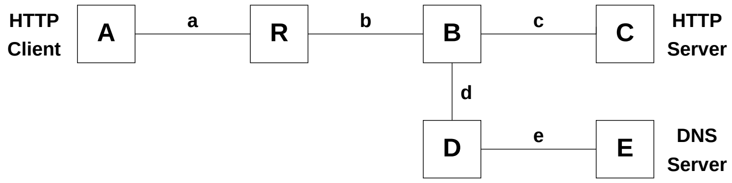
Note that base64 does not generate additional padding bytes if the final quantum of encoding input is exactly 8 bits. But we accept solutions resulting into 'aGVscAA==' as well since this we do not expect you to know this level of detail.

b) The `MIME-Version` header defines the MIME version being used. The `Content-type` header indicates a message consisting of multiple parts showing alternative representation of the same content and it defines that the boundary string to separate the messages is `NextPart`.

```
MIME-Version: 1.0
Content-type: multipart/alternative; boundary="NextPart"

This message is in MIME format.

--NextPart
Content-type: text/plain

hello
--NextPart
Content-type: text/html

<html><body>hello</body></html>
--NextPart--

This is the epilogue and will be ignored.
```

Problem F.3: *HTTP and DNS over Ethernet* (20 points)

Consider the following Ethernet network topology.



The devices A, C, and E are IP hosts, R is an IP router, and B and D are transparent bridges. The IP devices have static forwarding tables so that they can reach each other. The network just got initialized and no dynamic state information (except the static IP forwarding tables) is present in the devices. Let $mac(X)$ denote the MAC address of node X and $ip(X)$ denote the IP address of node X. The notation $mac(*)$ denotes an Ethernet broadcast address an $ip(*)$ an IP broadcast address.

An HTTP client is running on host A. The user selects a URL, e.g., by clicking on a link in a browser. The hostname of the selected URL is resolved by A's primary DNS server running on E to the IP address of device C. The host A then retrieves the document pointed to by the URL from the HTTP server running on C. Assume that the HTTP request and response each fit into a single TCP segment. Write down the messages that are exchanged over the various Ethernet segments by filling out the following table:

step	proto	mac-src	mac-dst	segments	ip-src	ip-dst	description
1							
2							
3							
4							
5							
6							
7							
8							
9							
10							
11							
12							
13							
14							
15							
16							
17							
18							
19							
20							
21							
22							
23							
24							
25							
...							

The *proto* column identifies the protocol contained in an Ethernet frame. The columns *mac-src* and *mac-dst* contain the Ethernet source and destination address while the columns *ip-src* and *ip-dst* contain the IP source and destination address where applicable. The column *segments* list the segments over which a frame is carried and the column *description* describes the contents of the frame.

Solution:

step	proto	mac-src	mac-dst	segments	ip-src	ip-dst	description
1	ARP	mac(A)	mac(*)	a			who has ip(R)?
2	ARP	mac(R)	mac(A)	a			ip(R) is at mac(R)
2	IP	mac(A)	mac(R)	a	ip(A)	ip(E)	ip/udp/dns query packet
4	ARP	mac(R)	mac(*)	b,c,d,e			who has ip(E)?
5	ARP	mac(E)	mac(R)	e,d,b			ip(E) is at mac(E)
6	IP	mac(R)	mac(E)	b,d,e	ip(A)	ip(E)	ip/udp/dns packet
7	IP	mac(D)	mac(R)	e,d,b	ip(E)	IP(A)	ip/udp/dns response packet
8	IP	mac(R)	mac(A)	a	ip(E)	IP(A)	ip/udp/dns response packet
9	IP	mac(A)	mac(R)	a	ip(A)	ip(C)	ip/tcp/syn packet
10	ARP	mac(R)	mac(*)	b,c,d,e			who has ip(C)?
11	ARP	mac(C)	mac(R)	b,c			IP(C) is at mac(C)
12	IP	mac(R)	mac(C)	b,c	IP(A)	IP(C)	ip/tcp/syn packet
13	IP	mac(C)	mac(R)	b,c	IP(C)	IP(A)	ip/tcp+syn+ack packet
14	IP	mac(R)	mac(A)	a	IP(C)	IP(A)	ip/tcp+syn+ack packet
15	IP	mac(A)	mac(R)	a	IP(A)	IP(C)	ip/tcp+ack/http+request
16	IP	mac(R)	mac(C)	b,c	IP(A)	IP(C)	ip/tcp+ack/http+request
17	IP	mac(C)	mac(R)	b,c	IP(C)	IP(A)	ip/tcp+ack+fin/http+reply
18	IP	mac(R)	mac(A)	a	IP(C)	IP(A)	ip/tcp+ack+fin/http+reply
19	IP	mac(A)	mac(R)	a	IP(A)	IP(C)	ip/tcp+ack+fin
20	IP	mac(R)	mac(C)	b,c	IP(A)	IP(C)	ip/tcp+ack+fin
21	IP	mac(C)	mac(R)	b,c	IP(C)	IP(A)	ip/tcp+ack
22	IP	mac(R)	mac(A)	a	IP(C)	IP(A)	ip/tcp+ack

Problem F.4: *Domain Name System* (3+3+3+3+4+4 = 20 points)

The DNS server serving the zone `eecs.jacobs-university.de.` has a configuration file that looks like the following (pretty much simplified):

```
;;
;; Zone file for eecs.jacobs-university.de.
;;
@               IN      SOA     ns.eecs.jacobs-university.de.   hostmaster.eecs.jacobs-university.de. (
                                2007101100      ; Serial (YYYYMMDDVV)
                                2H              ; Refresh
                                1H              ; Retry
                                4W              ; Expire
                                1D)             ; Minimum TTL
                        NS      ns
                        NS      dns.jacobs-university.de.
                        MX      10 peach.eecs.jacobs-university.de.
                        MX      20 hermes.jacobs-university.de.
;
; name          class   type    value
;
ns              IN      A       212.201.49.181
ns              IN      AAAA    2001:638:709:3000::5
;
mail            IN      CNAME   peach
www             IN      CNAME   orange
cnds            IN      CNAME   orange
;
orange          IN      A       212.201.49.182
peach           IN      A       212.201.49.184
;
orange          IN      AAAA    2001:638:709:3000::6
peach           IN      AAAA    2001:638:709:3000::8
```

a) What is the name of the EECS DNS server?

b) What are the addresses of the EECS DNS server?

c) Which mail server(s) accept email for the domain `eecs.jacobs-university.de.`?

d) Which web sites are likely served by the HTTP server running on the host with the IPv4 address 212.201.49.182?

e) How does the name for the IPv4 PTR record for the host `mail.eecs.jacobs-university.de.` look like?

f) How does the name for the IPv6 PTR record for the host `www.eecs.jacobs-university.de.` look like?

Solution:

a) `ns.eecs.jacobs-university.de.`

b) `212.201.49.181` and `2001:638:709:3000::5`

c) `peach.eecs.jacobs-university.de.` with a fallback to `hermes.jacobs-university.de.`

d) `www.eecs.jacobs-university.de.` and `cnds.eecs.jacobs-university.de`

e) `184.49.201.212.in-addr.arpa.`

f) `6.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.3.9.0.7.0.8.3.6.0.1.0.0.2.ip6.arpa.`

Problem F.5: *TCP Congestion Control* (2+2+6+10 = 20 points)

a) Explain briefly the difference between congestion control and flow control.

b) TCP is assumed to be fair in the sense that separate TCP streams achieve approximately the same transmission rate. Can an application use this knowledge to actually achieve some better throughput?

c) At the beginning of transmission round t, a TCP connection in congestion avoidance mode has a congestion window of $w = 60$ segments. Packet loss is observed during transmission rounds t, $t+10$, and $t+20$ by receiving multiple ACKs. What is the congestion window w at the end of transmission round t, $t+10$, and $t+20$? Assuming there is no further packet loss, when will the window of $w = 60$ segments be reached again?

d) Consider a TCP connection carrying 1000 byte packets with a round-trip time (RTT) of $100ms$. Suppose we want to achieve a data rate of 10 Gbps. What is the packet loss probability needed to achieve this data rate during congestion avoidance? Interpret the result.

Solution:

a) Flow control matches the speed of the sender to the capabilities of the receiver. Congestion control deals with the fair share assignment of resources (bandwidth and queuing capacity) in the network.

b) By using multiple TCP connections, every TCP connection will get $\frac{1}{n}$ of the maximum throughput if there are n connections. If an application uses significantly more TCP connections than other applications and the TCP connections get into congestion avoidance mode, then the application can get an overall advantage.

c) The congestion window is halfed during transmission round t, leading to $w = 30$. At the beginning of transmission round $t+10$, the window has increased to $w = 40$ but it will be halfed again during transmission round $t+10$ to $w = 20$. Similarly, after transmission round $t+20$, the window will be $w = 15$. With no further packet loss, 45 transmission rounds later at $t+65$, the window will reach again the original size of $w = 60$.

d) In steady state, TCP congestion avoidance reaches an average throughput of

$$\bar{X}(p) = \frac{1}{RTT}\sqrt{\frac{3}{2p}}$$

packets per second, where RTT is the round-trip time and p the packet loss probability. This equation can be rewritten as follows:

$$p = \frac{3}{2 \cdot (\bar{X}(p) \cdot RTT)^2}$$

The link capacity of $10Gbps = 10^9 bps$ divided by the packet size of $10^3 b$ leads to a maximum of 10^6 packets per second and we get:

$$p = \frac{3}{2 \cdot (0.1s \cdot 10^6 \frac{1}{s})^2} = \frac{3}{2} 10^{-10}$$

A packet loss probability of this order is unrealistic and hence a single TCP stream is not capable to saturate a $10Gbps$ link with an RTT of $100ms$.

Problem F.6: *Transmission Control Protocol* (2+2+2+2+2 = 10 points)

Indicate which of the following statements are correct or incorrect by marking the appropriate boxes. For every correctly marked box, you will earn two points. For every incorrectly marked box, you will loose one point. Statements which are not marked or which are marked as true and false will be ignored. The minimum number of points you can achieve is zero.

true	false	
☐	☐	The TCP state machine allows both endpoints to initiate an active close simultaneously, leading to an exchange of four TCP messages.
☐	☐	Early congestion notifications (ECN) assumes that routers mark network layer headers while packets travel to their destination and that the transport protocol instance at the destination carries the congestion notification bit back to the source in a transport layer header.
☐	☐	TCP retransmissions are always caused by an expiring retransmission timer.
☐	☐	The round-trip time (RTT) measured for retransmitted TCP segments is ambiguous and ignored for the estimation of the retransmission timeout.
☐	☐	The calculation of the retransmission timeout includes a smoothed estimate of the current round-trip time (RTT) and its variance.

Solution:

true	false	
☒	☐	The TCP state machine allows both endpoints to initiate an active close simultaneously, leading to an exchange of four TCP messages.
☒	☐	Early congestion notifications (ECN) assumes that routers mark network layer headers while packets travel to their destination and that the transport protocol instance at the destination carries the congestion notification bit back to the source in a transport layer header.
☐	☒	TCP retransmissions are always caused by an expiring retransmission timer.
☒	☐	The round-trip time (RTT) measured for retransmitted TCP segments is ambiguous and ignored for the estimation of the retransmission timeout.
☒	☐	The calculation of the retransmission timeout includes a smoothed estimate of the current round-trip time (RTT) and its variance.

Base64 Encoding Table

Value	Encoding	Value	Encoding	Value	Encoding	Value	Encoding
0	A	17	R	34	i	51	z
1	B	18	S	35	j	52	0
2	C	19	T	36	k	53	1
3	D	20	U	37	l	54	2
4	E	21	V	38	m	55	3
5	F	22	W	39	n	56	4
6	G	23	X	40	o	57	5
7	H	24	Y	41	p	58	6
8	I	25	Z	42	q	59	7
9	J	26	a	43	r	60	8
10	K	27	b	44	s	61	9
11	L	28	c	45	t	62	+
12	M	29	d	46	u	63	/
13	N	30	e	47	v		
14	O	31	f	48	w	(pad)	=
15	P	32	g	49	x		
16	Q	33	h	50	y		

Hexadecimal ASCII Character Set

00 nul	01 soh	02 stx	03 etx	04 eot	05 enq	06 ack	07 bel
08 bs	09 ht	0a nl	0b vt	0c np	0d cr	0e so	0f si
10 dle	11 dc1	12 dc2	13 dc3	14 dc4	15 nak	16 syn	17 etb
18 can	19 em	1a sub	1b esc	1c fs	1d gs	1e rs	1f us
20 sp	21 !	22 ‘	23 #	24 $	25 %	26 &	27 ’
28 (	29)	2a *	2b +	2c ,	2d -	2e .	2f /
30 0	31 1	32 2	33 3	34 4	35 5	36 6	37 7
38 8	39 9	3a :	3b ;	3c <	3d =	3e >	3f ?
40 @	41 A	42 B	43 C	44 D	45 E	46 F	47 G
48 H	49 I	4a J	4b K	4c L	4d M	4e N	4f O
50 P	51 Q	52 R	53 S	54 T	55 U	56 V	57 W
58 X	59 Y	5a Z	5b [	5c \	5d]	5e ˆ	5f _
60 ‘	61 a	62 b	63 c	64 d	65 e	66 f	67 g
68 h	69 i	6a j	6b k	6c l	6d m	6e n	6f o
70 p	71 q	72 r	73 s	74 t	75 u	76 v	77 w
78 x	79 y	7a z	7b {	7c |	7d }	7e ˜	7f del