

**Note:**

- During the attendance check a sticker containing a unique code will be put on this exam.
- This code contains a unique number that associates this exam with your registration number.
- This number is printed both next to the code and to the signature field in the attendance check list.

## Advanced Computer Networking

**Exam:** IN2097 / Retake-Online  
**Examiner:** Prof. Dr.-Ing. Georg Carle

**Date:** Monday 11<sup>th</sup> April, 2022  
**Time:** 14:15 – 15:30

### Working instructions

- This exam consists of **12 pages** with a total of **5 problems**.  
Please make sure now that you received a complete copy of the exam.
- The total amount of achievable credits in this exam is 75 credits.
- Detaching pages from the exam is prohibited.
- Allowed resources:
  - one **analog dictionary** English ↔ native language without annotations
  - the **provided cheatsheet** without annotations (print or digital copy)
- Subproblems marked by \* can be solved without results of previous subproblems.
- **Answers are only accepted if the solution approach is documented.** Give a reason for each answer unless explicitly stated otherwise in the respective subproblem.
- Do not write with red or green colors nor use pencils.
- Code of conduct:
  - I participate without the help of others and only use the allowed resources.
  - I do not share, discuss, or exchange any information related to the exam with anybody.
  - I feel in good health and I am able to participate in the exam.
  - I understood the examination policy, agree to the video supervision, and adhere to this process.

## Problem 1 Quiz (15 credits)

The following questions cover multiple topics and can be solved independently of each other. The multiple choice questions need to be filled out as follows:

Mark correct answers with a cross



To undo a cross, completely fill out the answer option



To re-mark an option, use a human-readable marking



Each question has **one single correct answer** that is worth 1 credit.

a)\* What is the length of an IPv6 address?

  $2^{128}$  byte 128 byte  $2^{128}$  bit 16 byte

b)\* Which of the following IPv6 addresses is a Solicited Node Address?

 ff02::1:ffb9:fd7f ff01::1:ffb9:fd7f 2002::1:bbb9:fd7f ff02::1:bbb9:fd7f

c)\* How long does it roughly take to scan the complete IPv4 address space with ZMap and a rate of 10 000 packets per second? Assume the complete address space is probed.

 50 h 200 h 24 h 120 h

d)\* Which statement about a Software-Defined Network (SDN) **is wrong**?

 In SDNs, a single control plane may control several forwarding planes. SDNs rely on a physically centralized control plane. In SDNs, control plane and data plane are separate. An SDN is not required to realize Network Function Virtualization (NFV).

e)\* Which match type is **not** part of the core P4 language according to the P4 specification?

 exact range ternary lpm

f)\* Which congestion control algorithm is **loss-based**?

 TCP BBR TCP Vegas TCP Illinois TCP Reno

g)\* What is the Organisation Unique Identifier (OUI) and at which position can it be found?



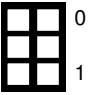
- First three bytes of a MAC address (in transmission order)
- Vendor specific code in a MAC address

h)\* Does the Spanning Tree Protocol (STP) prevent routing loops?



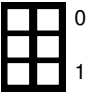
- STP works on Layer 2, routing happens on Layer 3.
- Routing loops can form independent of Layer 2, therefore STP cannot prevent that.

i)\* What is the main difference between classful and classless IP addressing, regarding prefix length?



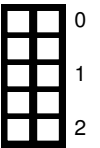
- Classful addressing only supports fixed prefix lengths (/8, /16, /24)
- Classless addressing allows arbitrary prefix lengths (up to 32 bit)

j)\* Name the two latest DNS encryption mechanisms discussed in the lecture.



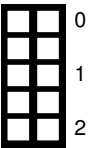
- DoT and DoH

k)\* Explain where DNS encryption protocols introduced in the lecture are operating and why that is useful.



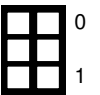
- DoT and DoH are used between the client and resolver (name servers are not included)
- Usually a large number of client uses the same public resolvers. Therefore, integrity between resolver and authoritative name server is not a concern.

l)\* What is the difference between an active and an inactive timeout and when would they be triggered?



- Inactive timeout: A flow ended and is exported after the timeout expired without new packets.
- Active timeout: Data is exported after the timeout for continuous flows even though the flow did not end.

m)\* Why do QUIC ZMap scans require more bandwidth than TCP ZMap scans?



- QUIC requires a meaningful client initial packet (at least 1200 B).
- For TCP, ZMap only sends a simple SYN.

## Problem 2 Longest Prefix Matching (LPM) (14 credits)

Longest Prefix Matching (LPM) is the algorithm that is performed by a router. The router performs the LPM for each packet to determine the next hop of a packet and perform an appropriate forwarding decision.

There are several notations to represent the network parts of IPv4 addresses. The slash-notation /x, the dotted-decimal notation a.b.c.d, or a subnetmask represented as a hex value 0xffffffff.



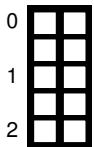
a)\* Convert /17 into the dotted-decimal notation.

/17 → 255.255.128.0



b)\* Convert 255.255.254.0 into a subnetmask as a hex value.

255.255.254.0 → 0xff ff fe 00



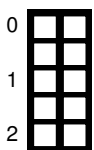
c)\* Explain briefly if 128.255.255.255 is a valid subnet mask.

- The binary value of valid subnet masks must either be 0 or start with a continuous sequence of 1 followed by a continuous sequence of 0, i.e., it must adhere to the regular expression  $1^*0^+$ .
- The subnet mask starting with 128.255 does not follow this shape and therefore cannot represent a valid subnet mask.

A major part of the LPM is the matching operation (cf. Listing 1). The given matching function has three unsigned 32-bit integers as input arguments. The first argument contains the IPv4 address of the packet that should be forwarded (`uint32 addr`). The second argument contains the IPv4 address of a routing entry that should be matched (`uint32 entry`). The third argument contains the subnet mask of the routing entry (`uint32 mask`). `match()` returns true if the entry matches, false otherwise.

```
1 boolean match(uint32 addr, uint32 entry, uint32 mask) {  
2     // enter code here  
3 }
```

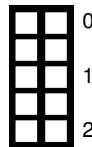
Listing 1: Matching function



d)\* Complete the match function of Listing 1.

```
return (addr & mask) == entry
```

e)\* Explain why the routing entry of the default gateway is typically using a subnet mask of 0.



- The default gateway should match against any entry.
- A subnet mask of 0 ensures that behavior, by disabling the extraction and matching of the network part.

For the following subproblems, we perform the LPM using a list of routing table entries as a routing table. This list contains a number of  $n$  routing table entries. LPM iterates over the list of routing table entries sequentially to determine the longest matching prefix.

f)\* LPM selects the "longest" entry. What does that mean?



For a number of matching entries, LPM selects the entry with the largest value of the subnet mask.

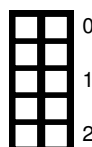
g) Briefly explain, how many matches must be performed for an **unsorted** list with  $n$  unique entries, before the longest entry can be selected.



- In an unsorted list, each following entry can be more specific than the previous entry.
- Therefore, all of the  $n$  entries have to be matched before the packet can be forwarded.

For the following subproblem, we sort the routing table using the **IPv4 address values**. The list is sorted in ascending order, i.e., the list begins with the routing entry containing the lowest address value.

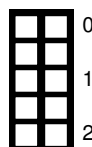
h) Briefly explain, how many matches must be performed for this **address-sorted** list with  $n$  unique entries, before the longest entry can be selected.



- Sorting by address value is irrelevant for the LPM that selects entries based on the subnet mask.
- One has to iterate over the list of possible entries until one reaches an address value greater than the one that is looked up. For the highest possible address values, this means that  $n$  entries have to be matched before the packet can be forwarded.

For the following subproblem, we sort the routing table using the **IPv4 subnet masks**. The list is sorted in ascending order, i.e., the list begins with the routing entry containing the lowest subnet mask value.

i) Briefly explain, how many matches must be performed for this **mask-sorted** list with  $n$  unique entries, before the longest entry can be selected.



- The longest entries are at the end of the list.
- This means that again almost all entries have to be matched before the packet can be forwarded.
- The algorithm can return its final result when a match for a /32 entry is found (without matching the rest of the /32 entries)

### Problem 3 Load Balancing (15 credits)

Load balancing is important to properly serve large amounts of clients with a good user experience. The following problem is based on the network shown in Figure 3.1.

A company operates two load balancers (LB1 and LB2) each connected to multiple content servers. Furthermore, they operate a nameserver (NS) used to resolve company domains to both load balancers.

On the opposite side, two clients (C1 and C2) frequently contact the company and access their content and a public DNS resolver (R) is available.

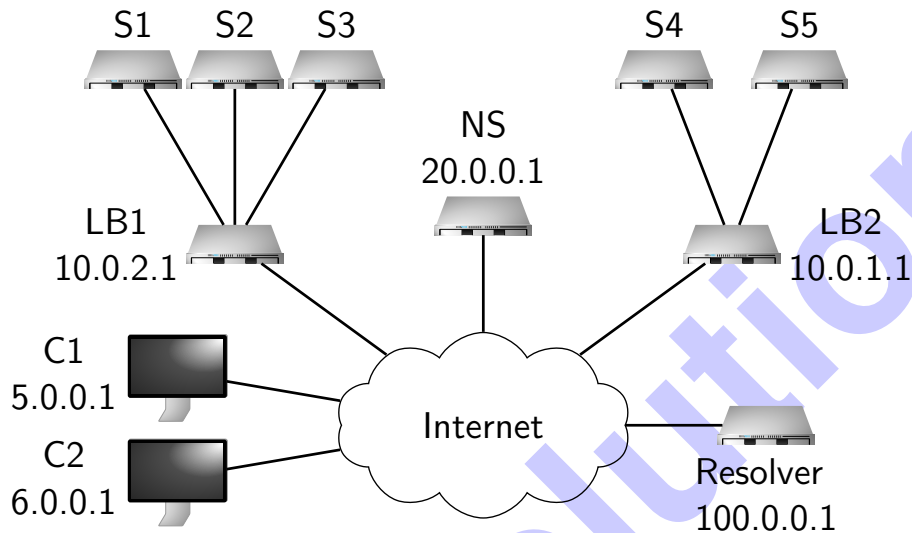


Figure 3.1: Important company with exam relevant content




0  a)\* The company nameserver's zone file contains resource records listed in Table 3.1. Fill in the missing information, so that the nameserver can actively balance the load in between the load balancers.

Table 3.1: Company NS Resource Records

relevant.exam	900	IN	A	10.0.2.1
relevant.exam	900	IN	A	10.0.1.1

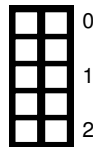
0  b)\* Explain why it is not a good idea for the company nameserver to return both records in one reply.

- The client decides which record it uses
- The load on the two LBs cannot be controlled and can be biased

0  c)\* If C1 and C2 use the public DNS resolver, why is geographical load balancing not possible with traditional DNS?

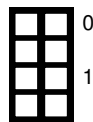
- The resolver queries the NS
- The NS only sees the IP address of the resolver and can not differentiate between C1 and C2's location

d) Name an extension of DNS that solves the previous problem and shortly explain how it works.



- ECS (EDNS Client Subnet)
- An extension that allows the resolver to forward a client subnets to the NS
- Nameservers can indicate for which prefix size the response is valid

e) What is the downtime (in minutes) of the service in the worst case if one of the load balancer fails?



- In the worst case, the client just received a fresh DNS record with the maximum TTL (900) seconds
- 900 seconds → 15 minutes

f)\* Based on the lecture, how can LB1 use modulo hashing to load balance connections to the content servers?



- LB1 can hash (h) the 5-tuple of the connection and select the server  $h \bmod N$

g) What happens if one of the content servers fails especially using modulo hashing?



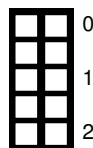
- Connections to that server are disrupted
- Changing N potentially hashes every client to a new client disrupting additional connections.

h)\* How does consistent hashing help with the previous problem?



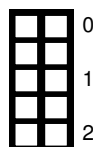
- Only Connections to the failing server are disrupted
- New connections are simply mapped to the next server on the circle

i) Assume consistent hashing is used but a planned update needs to be done on a content server. Describe what needs to be done to not disrupt services while using all available servers as long as possible.



- New connections, (e.g., TCP SYN) need to be hashed to all remaining servers
  - As long as established connections to the server exist, they need to be hashed to this server
- Only afterwards, the server can be updated

j) Instead of DNS load balancing, how else could traffic be load balanced between LB1 and LB2 using routing? Does it prevent service disruptions if a load balancer fails?



- Alternative: Both load balancers are in different locations announcing the same prefix
- Clients reach different load balancers based on routing decisions
- In case of a load balance failure, the service will still be unreachable for some clients until routing tables are updated

## Problem 4 Network Calculus (12.5 credits)

This problem investigates performance bounds in networks using Network Calculus.

Consider the following network topology with flow and server descriptions. Flow  $f_1$  traverses Servers  $s_1$ ,  $s_2$ , and  $s_3$ . Flow  $f_2$  traverses Servers  $s_2$  and  $s_3$ . Assume each server handles flows according to strict priority scheduling with preemption. Assume Flow  $f_1$  **has a low priority** and Flow  $f_2$  **has a high priority**. In the following, we want to apply the Separate Flow Analysis to compute an end-to-end delay bound **for Flow**  $f_1$ .

**Hint:** Use the following formula to calculate a left-over service curve:  $\beta^{l.o.} = [\beta_{R,T} - \gamma_{r,b}]^+ = \beta_{R-r, \frac{b+r \cdot T}{R-r}}$ . An output arrival curve is given by  $\alpha^* = \alpha_{r,b+r \cdot T}$ .

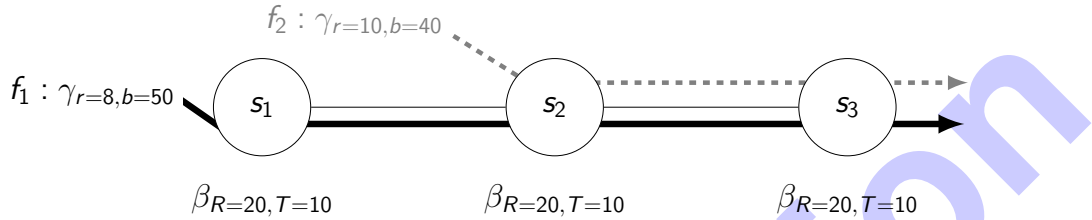


Figure 4.1: Network topology with flow and server specifications

a)\* Perform the first step (left-over service curves) of the Separate Flow Analysis for  $f_1$ .

Left-over service curve at  $s_1$ :

$$\beta_{s_1}^{l.o.} = \beta_{s_1} = \beta_{R=20, T=10}$$

Left-over service curve at  $s_2$ :

$$\beta_{s_2}^{l.o.} = [\beta_{s_2} - \gamma_{f_2}]^+ = \beta_{R=R_{s_2}-r_{f_2}, T=\frac{b_{f_2}+R_{s_2} \cdot T_{s_2}}{R_{s_2}-r_{f_2}}} = \beta_{R=20-10, T=\frac{40+20 \cdot 10}{20-10}} = \beta_{R=10, T=24}$$

Left-over service curve at  $s_3$ :

$$\gamma_{f_2}^* = \gamma_{r_{f_2}, b_{f_2}+r_{f_2} \cdot T_{s_2}} = \gamma_{10, 40+10 \cdot 10} = \gamma_{10, 140}$$

$$\beta_{s_3}^{l.o.} = [\beta_{s_3} - \gamma_{f_2}^*]^+ = \beta_{R=R_{s_3}-r_{f_2}^*, T=\frac{b_{f_2}^*+R_{s_3} \cdot T_{s_3}}{R_{s_3}-r_{f_2}^*}} = \beta_{R=R_{s_3}-r_{f_2}, T=\frac{b_{f_2}+r_{f_2} \cdot T_{s_2}+R_{s_3} \cdot T_{s_3}}{R_{s_3}-r_{f_2}}} = \beta_{R=20-10, T=\frac{40+10 \cdot 10+20 \cdot 10}{20-10}} = \beta_{R=10, T=34}$$

b) Perform the second step (end-to-end left-over service curve) of the Separate Flow Analysis.

End-to-end left-over service curve:

$$\beta_{e2e}^{l.o.} = \beta_{s_1}^{l.o.} \otimes \beta_{s_2}^{l.o.} \otimes \beta_{s_3}^{l.o.} = \beta_{R=\min(R_{s_1}^{l.o.}, R_{s_2}^{l.o.}, R_{s_3}^{l.o.}), T=T_{s_1}^{l.o.}+T_{s_2}^{l.o.}+T_{s_3}^{l.o.}} = \beta_{R=\min(20, 10, 10), T=10+24+34} = \beta_{R=10, T=68}$$

c)\* Perform the third step (calculate the delay bound) of the Separate Flow Analysis. If you have no solution to b), assume an end-to-end left-over service curve of  $\beta_{R=10, T=100}$ .

$$\text{End-to-end delay bound: } d = T_{e2e}^{l.o.} + \frac{b_{f_1}}{R_{e2e}^{l.o.}} = 68 + \frac{50}{10} = 73$$

$$(\text{Alternative: } d = 100 + \frac{50}{10} = 105)$$



d)\* Consider the following scenario to the network in Figure 4.1:

- The rate of server  $s_1$  is set to  $R = 10$

Explain the influence on the delay bound of flow  $f_1$  as determined by the Separate Flow Analysis.

The delay bound stays the same since a change in  $R_{s_1}$  only influences the the left-over service curve at  $s_1$ . The choice of  $R = 10$  does not change the end-to-end left-over service curve used to derive the delay bound.



e)\* Consider a token-bucket constrained flow, traversing a single rate-latency server. The corresponding arrival- and service curves are shown in Figure 4.2. Determine the **burst value of the flow** and the **rate of the server**.

Visual determination:

- Flow burst: 40 (bit)
- Server rate: 1 (bit/s)

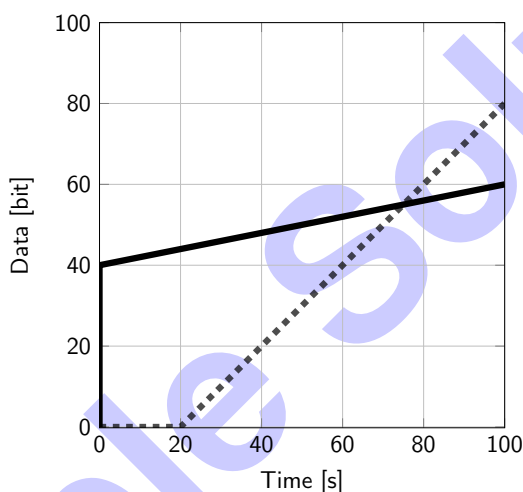
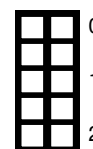


Figure 4.2: Arrival- and service curve

f)\* Consider the following statement:

- A rate-latency service curve is defined as  $\beta_{R,T}(s) = R \cdot s + T$  with rate parameter  $R$  and latency parameter  $T$ .

Argue whether or not the statement is correct.

Any one of:

- No, the definition of a rate-latency service curve is  $\beta_{R,T}(s) = R \cdot [s - T]^+$
- No, this is the definition of a token-bucket arrival curve



g)\* Name one mathematical framework that can be used to obtain soft real-time guarantees **and cannot** be used to obtain hard real-time guarantees.

Any one of:

- Stochastic Network Calculus
- Queuing Theory



## Problem 5 Wireshark (18.5 credits)

The ISO-OSI model defines seven layers in a communication system. For each layer multiple protocols exist. In this problem, a frame is analyzed, referring to the involved protocols. You are given a hexdump of an Ethernet frame including FCS, starting with the Ethernet header. For the following problems we already marked the headers of different layers in Figure 5.1.

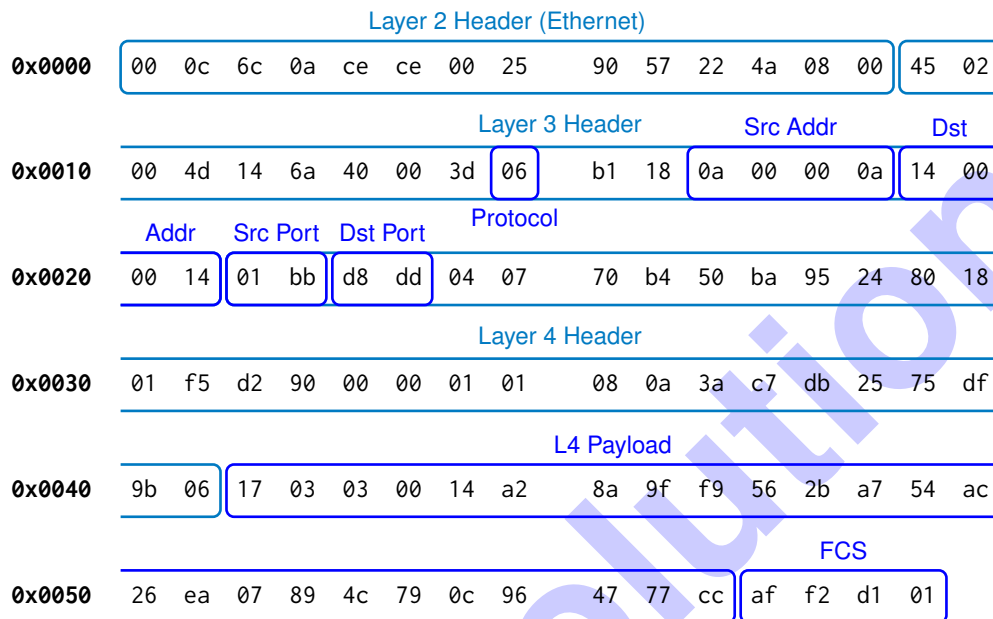


Figure 5.1: Hexdump of an Ethernet frame including FCS

In this problem you **always** have to substantiate your answers using the bytes in the hexdump in Figure 5.1. You can **either** mark the corresponding bytes directly in the figure **or** list the locations of the corresponding bytes using [...]. Example: the **three** bytes from position 0 to 2 can be written as [0, 2] = 0x000c6c.

0	
1	
2	

a)\* What is the name of **Layer 2** and **Layer 3** in the OSI model?

<b>Layer 2:</b> Data Link Layer	<b>Layer 3:</b> Network Layer
------------------------------------	----------------------------------

0	
1	

b)\* In Figure 5.1, mark the FCS.

FCS: [91, 94] ([0x5b, 0x5e])

0	
1	
2	

c)\* What is the size of the **Layer 3 PDU** in bytes.

L3PDU = L3 Header + L3 Payload = 77 B

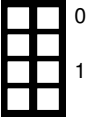
0	
1	
2	

d)\* What is the size of the **Layer 4 SDU** in bytes.

L4SDU = L4 Payload = 25 B

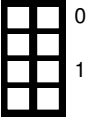
e) Compute the share of the L4 SDU in the whole hexdump.

$$\frac{\text{L4 SDU}}{\text{Frame Size}} = \frac{25 \text{ B}}{95 \text{ B}} = \frac{5}{19}$$



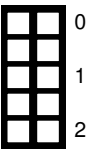
f)\* Identify the **Layer 3** protocol.

Ethertype: [12, 13] ([0xc, 0xd]) = 0x0800  
IPv4



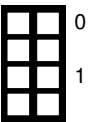
g) List all addresses contained in the **Layer 3** header in the common notation.

Source Address: [26, 29] ([0x1a, 0x1d]) = 10.0.0.10  
Destination Address: [30, 33] ([0x1e, 0x21]) = 20.0.0.20



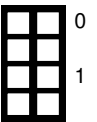
h) Identify the **Layer 4** protocol.

Protocol: [23] ([0x17]) = 0x06  
TCP



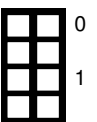
i) Make an educated guess which protocol can be expected **on top of Layer 4**.

Destination Port: [36, 37] ([0x24, 0x25]) = 0xd8dd → no well-known port  
Source Port: [34, 35] ([0x22, 0x23]) = 0x01bb = 443  
With TCP/443 usually HTTPS is used meaning the next expected protocol will be TLS.



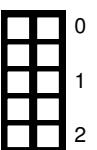
j)\* Can the two subnets 10.0.0.0/8 and 11.0.0.0/8 be merged? If yes, give the resulting network address and subnet.

Yes, they can be merged into 10.0.0.0/7.



k)\* Name **and** briefly explain **two** design goals of QUIC, which target weaknesses present with TCP / TLS.

Decrease handshake delay: transport layer and TLS handshakes are merged into one  
Get rid of head-of-line blocking: multiplex multiple streams within one connection  
Faster development cycles: implemented in user space  
IP mobility: uses connection IDs instead of 5-tuple



Additional space for solutions—clearly mark the (sub)problem your answers are related to and strike out invalid solutions.

