Chair of Network Architectures and Services
TUM School of Computation, Information, and Technology
Technical University of Munich

Esolution
Place student sticker here

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 / Endterm	**Date:**	Monday 19th February, 2024
Examiner:	Prof. Dr.-Ing. Georg Carle	**Time:**	13:30 – 14:45

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
- 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.
- Physically turn off all electronic devices, put them into your bag and close the bag.

Left room from ______ to ______ / Early submission at ______

Problem 1 Quiz (18 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 ×■

a)* Which of the following is a correct IPv6 address?

- [ ] `fc21:2001:11:3223:3ff:fe74:150a`
- [x] `fc21::3223:3ff:fe74:150a`
- [ ] `fc21::11:3223:3fg:fe74:150a`
- [ ] `fc21::2001:11:3223::fe74:150a`

b)* Which of the following parameters requires DHCPv6 and is not available with ICMPv6-based configuration? ***Invalid question and not graded:*** *An option to provide DNS information was added in a later RFC, but not properly mentioned during the lecture.*

- [x] Static addresses
- [x] Netboot
- [ ] Prefix information
- [ ] DNS information

c)* According to the Association for Computing Machinery (ACM), which of the statements below is correct.

- [ ] Repeatability: **same** team using **different** experimental equipment.
- [x] Reproducibility: **different** team using **same** experimental equipment.
- [ ] Recreatability: **different** team using **different** experimental equipment.
- [ ] Replicability: **same** team using **same** experimental equipment.

0
1
d)* Name the IPv6 alternative for ARP and briefly explain its mechanism.

- In IPv6, the Neighbor Discovery Protocol is used.
- A host sends a neighbor solicitation to a solicited node multicast address and MAC
- All hosts with matching addresses respond with a neighbor advertisement

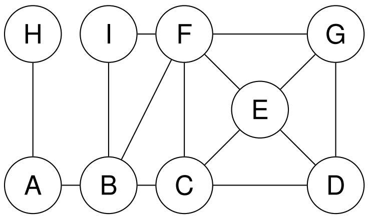
0
1
e)* What is used to identify a TCP connection? What is different with QUIC to achieve IP mobility?

- TCP: 5-tuple
- QUIC: Connection ID

0
1
2
f)* Shortly explain why traceroute might reveal non-existing paths and a solution from the lecture tackling this problem.

- Probes are independent → Can be routed independently on different paths
- Create probes that are load balanced the same, e.g., same 5-Tuple

g)* Perform the k-core algorithm for the topology shown in the solution box. For each k, list all removed nodes.



- $k = 1$ H, A
- $k = 2$ I, B
- $k = 3$ C, D, E, F, G $\rightarrow$ Done

h)* Table 1.1 contains all records a resolver receives as part of the authority response section while resolving `example.com`. The authoritative answer flag is not set. Shortly explain what this message means and name the next DNS message (including its destination) the resolver sends. Assume the resolver has no cache.

1	example.com.	300	IN	NS	dns1.lrz.de.
2	example.com.	7200	IN	NS	dns2.lrz.bayern.

Table 1.1: DNS Records received by the resolver. All records are part of the authority section.

- Delegation to the authoritative name servers of `example.com`
- The resolver must now resolve one of the two name server names and issues a query for the corresponding name to the DNS root servers

i)* As an administrator of an authoritative name server: Would you prefer 300 or 7200 as the NS record's TTL? Argue according to lecture information and Briefly explain.

NS records rarely change. Therefore, the larger TTL of record 2 results in fewer queries towards the authoritative name server

j)* A company wants to create a Layer 2 connection between two sites in different countries via Internet. The Internet service provider (ISP) of the company only offers Layer 3 connectivity. The administrator proposes to create a VLAN tunnel between the two sites. Argue if the proposed solution works.

VLAN requires control over Layer 2, which the administrator does not have, therefore it does not work

k)* How do TCP Cubic and TCP BBRv1 react to packet-loss in the network?

Cubic is loss-based and reduces its congestion window to 70 % on packet-loss.
BBRv1 does not react on packet-loss at all.

l)* **Name** and briefly **explain two** ways a TCP sender detects packet-loss and starts a retransmission.

Retransmission timeout: no acknowledgement arrives for a certain time
Fast Retransmit: three duplicate acknowledgements arrive indicating a lost segment

Problem 2 BGP and Routing (17.5 credits)

This problem investigates the autonomous system (AS) relationships in a given network and their impact on routing and traffic.

a)* Shortly explain the Valley-Free-Routing principle.

- upstream: sequence of Customer Provider links (possibly length = 0)
- then possibly **one** peering link
- the downstream: sequence of Provider Customer links (possibly length = 0)

b)* How can a Tier-2 provider be defined.

Providers are all Tier-1, otherwise only peering or customers.

c)* Based on the lecture, name the most important consideration for policy routing after longest prefix matching and a potential negative effect.

It is all about money, longer/slower paths might be chosen if more money can be generated.

d)* What is the goal of longest prefix matching, and how can it naively be implemented?

- Allows to create a hierarchical structure within the address space.
- LPM ensures that traffic is routed to the most specific prefix.
- A naive approach simply sorts the FIB, longest prefix first.

e)* Which is the smallest prefix fully covering `10.0.24.0/23` and `10.0.28.0/23`.

`10.0.24.0/21`, a /22 does not cover both /23.

For the following subproblems the topology in Figure 2.1 will be used.

- $\rightarrow$ represents a customer to ($\rightarrow$) provider relationship.
- $\leftrightarrow$ represents a peering relationship.
- AS E owns and announces `172.0.0.0/23` while AS F owns and announces `172.0.0.0/22`.
- All ASes apply standard routing behavior. Furthermore, the following policies are applied:
  - For routes with the same prefix, the AS selects the most cost-efficient route.
  - For routes with the same prefix and with an equal traffic cost, the shorter route is selected.

f)* A client in AS G wants to connect to 172.0.1.27. To which AS is traffic routed and via which path? Explain your answer.

- The traffic is routed towards AS E (LPM)
- It is routed via AS D - AS A - AS B
- AS C will not announce the route to AS D No money, only traffic

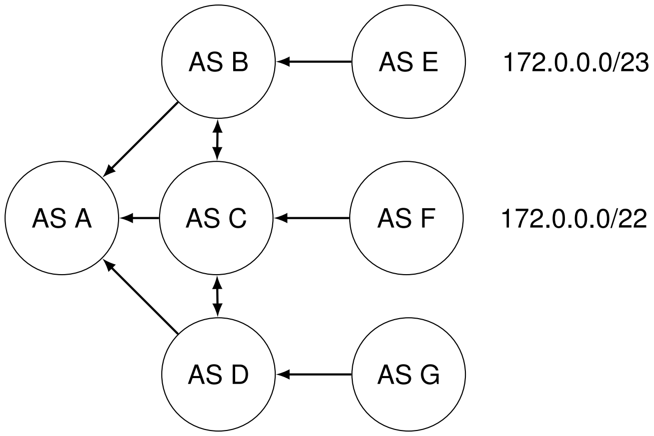


Figure 2.1: AS Network

g)* A client in AS G wants to connect to 172.0.2.27. To which AS is traffic routed and via which path? Explain your answer.

- The traffic is routed towards AS F (LPM)
- It is routed via AS D - AS C
- AS C will announce the route to AS D Peering and money from AS F

For the following subproblems, only consider the two prefixes `172.0.0.0/23` and `172.0.0.0/22`.

h)* How many different addresses can AS G reach within AS F and how many addresses can AS G reach in AS E?

- AS G can reach 512 addresses within each AS
- The /22 covers 1024 addresses but due to LPM, 512 are routed towards AS E.

i) Assume AS F additionally announces `172.0.1.0/24`. How is the number of addresses within ASes E and F impacted that can be reached from AS G?

The number of addresses within AS E is reduced by 256 while the number of addresses within AS F is increases due to LPM.

j)* How can AS A intercept all traffic as man-in-the middle, without exposing itself as the origin of a route announcement? Explain your solution.

- AS A needs to announce routes towards the original AS (fake path hijack), but with more specific prefixes.
- This makes sure, that the new route is chosen by its customers, even though they have to spend money (LPM).

Problem 3 Hexdump (12.5 credits)

This problem investigates a captured Ethernet frame.

```
              Destination MAC                          Ethertype
0x0000   3c ec ef 93 23 b9 3c ec   ef 99 18 67 08 00 45 c0
                            Protocol                    Dst IP
0x0010   02 40 52 e6 00 00 40 01   11 15 0a 00 00 02 0a 00
          Addr  TypeCode    H1                  H2
0x0020   00 01 03 03 77 fc 00 00   00 00 45 00 04 e0 b5 c0
                  Protocol                          H3
0x0030   40 00 40 11 6c 4a 0a 00   00 01 0a 00 00 02 b5 86
          Dst Port                    H4
0x0040   01 bb 04 cc d5 32 c6 00   00 00 01 08 c0 ac . . .
```

Figure 3.1: Hexdump of an Ethernet frame starting with the Ethernet header.

In this problem you **always** have to substantiate your answers using the bytes of the hexdump in Figure 3.1. Always make clear which bytes are relevant for each answer. 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] = 0x3cecef`. Note, counting starts at 0 and start and end are included. Hexadecimal notation is also allowed: e. g., `[0x10, 0x12] = 0x024052`.

a) An Ethernet Frame consists of the header, the payload, and the *Frame Check Sequence* (FCS). What is the purpose of the FCS and how is it computed?

The FCS is used to detect bit-errors and is computed using a CRC checksum.

b)* Mark the **Destination MAC Address** and note it in its common notation.

Destination MAC Address: `[0, 5] = 0x 3c ec ef 93 23 b9`
`3c:ec:ef:93:23:b9`

c)* Identify the used **Layer 3** protocol. (Do not forget to mark and name relevant fields.)

Ethertype: `[12, 13] = 0x0800` ⇒ IPv4

d) Mark the **Destination IP Address** and note it in its common notation.

Destination IP Address: `[30, 33] = 0x0a000001`
`10.0.0.1`

In Figure 3.1 different protocol headers **H1**, **H2**, **H3**, and **H4** are marked with dashed lines. In the following you will identify which protocols they belong to.

e) Identify the protocol of **H1**. Parse the header and explain the purpose the this message.

Protocol: `[23] = 0x01` ⇒ ICMPv4
Type: `[34] = 0x03` ⇒ Destination unreachable
Code: `[35] = 0x03` ⇒ Destination port unreachable

This ICMP message is sent to indicate to the sender that the destination port is not active.

f) Based on your result from e), identify which type of protocol **H2** belongs to.

In e) ICMPv4 was identified. The type and code indicate a destination port unreachable message. This message contains the full IPv4 header of the original datagram as well as parts of the original data. Therefore, the header **H1** belongs to an IPv4 packet.

g) Identify the type of protocol of the header **H3**.

Protocol: `[51] = 0x11` ⇒ UDP

h) Argue which **protocol and application** is transported in **H4**.

Source Port: `[62,63] > 1024` (not a well-known port)
Destination Port: `[64, 65] = 0x01bb = 443` well-known port for HTTPS
In combination with UDP as transport protocol, most likely the application was HTTP/3 running on top of QUIC.

Problem 4 Network Calculus (14 credits)

This problem investigates performance bounds in networks using Network Calculus.

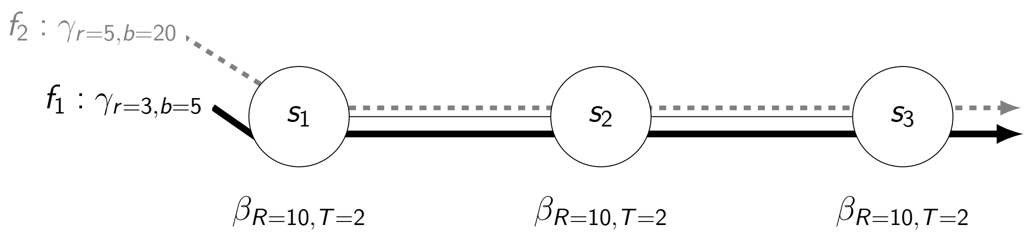


Figure 4.1: Topology with server- and flow specifications

Consider the topology in Figure 4.1. Assume each server employs strict priority queuing. Flow f_1 has the lowest priority while Flow f_2 has the highest priority.
We are interested in calculating the end-to-end delay bound of **Flow f_1** using the Separate Flow Analysis.
Hint: $\beta^{l.o.} = \beta_{R-r,\frac{b+R\cdot T}{R-r}}$ and $\alpha^{\star} = \gamma_{r,b+r\cdot T}$

a)* Perform the first step of the Separate Flow Analysis.

- Left-over service curve at s_1: $\beta^{l.o.1}_{s_1} = \left[\beta_{10,2} - \gamma_{5,20}\right]^+ = \beta_{10-5,\frac{20+10\cdot 2}{10-5}} = \beta_{5,8}$
- Output arrival curve of f_2 after s_1: $\alpha^{\star} = \gamma_{5,20+5\cdot 2} = \gamma_{5,30}$
- Left-over service curve at s_2: $\beta^{l.o.1}_{s_2} = \left[\beta_{10,2} - \alpha^{\star}\right]^+ = \left[\beta_{10,2} - \gamma_{5,30}\right]^+ = \beta_{10-5,\frac{30+10\cdot 2}{10-5}} = \beta_{5,10}$
- Output arrival curve of f_2 after s_2: $\alpha^{\star\star} = \gamma_{5,30+5\cdot 2} = \gamma_{5,40}$
- Left-over service curve at s_3: $\beta^{l.o.1}_{s_3} = \left[\beta_{10,2} - \alpha^{\star\star}\right]^+ = \left[\beta_{10,2} - \gamma_{5,40}\right]^+ = \beta_{10-5,\frac{40+10\cdot 2}{10-5}} = \beta_{5,12}$

b) Perform the second step of the Separate Flow Analysis.

$\beta^{l.o.}_{e2e} = \beta^{l.o.}_{s_1} \otimes \beta^{l.o.}_{s_2} \otimes \beta^{l.o.}_{s_3} = \beta_{min(5,5,5),8+10+12} = \beta_{5,30}$

c) Perform the third step of the Separate Flow Analysis.

$d_{e2e} = T_{e2e} + \frac{b}{R_{e2e}} = 30 + \frac{5}{5} = 31$

d) Assume the following changes to the scenario in Figure 4.1:

- Flow f_3 joins the network with the same path and priority as Flow f_2
- Flow f_3 has an arrival curve with $r_3 = 3$ and $b_3 = 1$

Argue how the delay bound of Flow f_1 changes. Be specific.

Delay bound is infinity because the rate of the left-over service curve is less than the rate of f_1

e)* A flow with arrival curve $\gamma_{r=5,b=10}$ is traversing a server with service curve $\beta_{R=20,T=2}$. Calculate the **backlog bound** of the flow.

backlog $= b + r \cdot T = 10 + 5 \cdot 2 = 20$

f)* A flow with arrival curve $\gamma_{r=5,b=10}$ is traversing 1,000 servers connected in series, each with the same service curve $\beta_{R=20,T=2}$. Calculate the **delay bound** of the flow. The method you choose should calculate a tight delay bound.

- Concatenate service curves: $\beta_{e2e} = \beta_{min(\bigcup_{s_i \in S} R_i), \sum_{s_i \in S} T_i} = \beta_{20,1000 \cdot 2} = \beta_{20,2000}$
- $d_{e2e} = T_{e2e} + \frac{b}{R_{e2e}} = 2000 + \frac{10}{20} = 2000.5$

g)* What is calculated by the following formula?

- $(\alpha \oslash \beta)(0)$

Backlog bound or maximal vertical deviation between arrival- and service curve

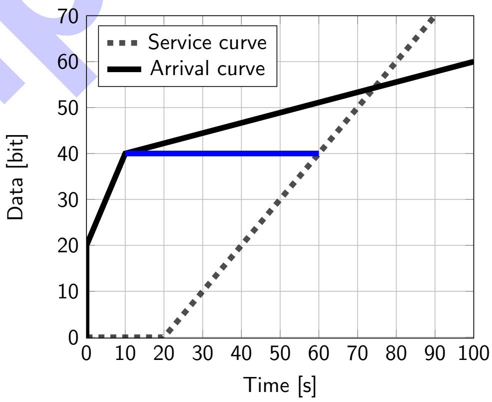


Figure 4.2: Arrival- and service curve

h)* Consider the two curves in Figure 4.2. What is the delay bound of a flow with this **non-**token-bucket arrival curve traversing a server with this rate-latency service curve?

50

Problem 5 Software-Defined Networking (13 credits)

This problem investigates a Software-Defined Network (SDN) powered by P4. For the following problems, consider the network given in Figure 5.1. Server A is configured to create and accept untagged frames, Server B only accepts VLAN-tagged frames (VLAN ID 15). PCP and DEI are always set to `0`. Switch 1 is a P4 switch handling the VLAN functionality. In the following subproblems, the P4 program running on Switch 1 is investigated. Listing 1 shows parts of the used P4 program.

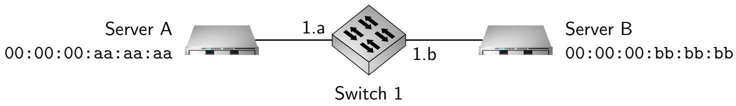
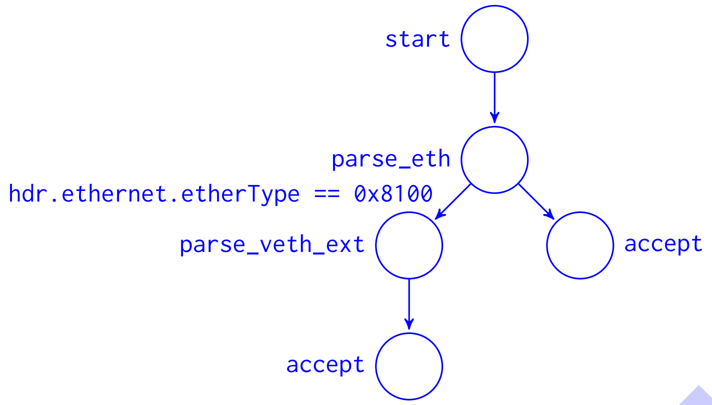


Figure 5.1: Network topology

```
header eth_t                { bit<48> dstAddr;
                              bit<48> srcAddr;
                              bit<16> etherType; }
header veth_ext_t           { bit<3> pcp;
                              bit<1> dei;
                              bit<12> vid;
                              bit<16> etherType; }
struct standard_metadata_t { bit<16> ingress_spec;
                              bit<16> egress_spec;
                              // ...
                            }
struct meta                 { /* unused */ }
struct headers              { eth_t eth;
                              veth_ext_t veth_ext; }

parser ParserImpl(packet_in packet, out headers hdr, inout meta meta, inout standard_metadata_t
    std_meta) {
  // to be defined in Subproblem a)
}

control Pipeline(inout headers hdr, inout metadata meta, inout standard_metadata_t std_meta) {
  action drop() {
    mark_to_drop();
  }
  action decap(bit<16> egress) {
    std_meta.egress_spec = egress;
    hdr.eth.etherType = // to be defined in Subproblem b)
    hdr.veth_ext.setInvalid() ;
  }
  action encap(bit<16> egress, bit<3> pcp, bit<1> dei, bit<12> vid) {
    std_meta.egress_spec = egress;
    hdr.veth_ext.etherType = // to be defined in Subproblem c)
    hdr.eth.etherType = // to be defined in Subproblem c)
    hdr.veth_ext.setValid();
    hdr.veth_ext.pcp = pcp;
    hdr.veth_ext.dei = dei;
    hdr.veth_ext.vid = vid;
  }
  table forward {
    actions = {
      encap;
      decap;
      drop;
    }
    key = {
      std_meta.ingress_spec: exact;
      hdr.eth.src: exact;
    }
    size = 4;
    default_action = drop;
  }
  apply {
    if (hdr.eth.isValid()) {
      forward.apply();
    }
  }
}

\\ ...
```

Listing 1: VLAN P4 program

a)* Visualize the parse graph of Listing 1 as state machine. The parser graph starts at a start state and must be able to accept tagged and untagged Ethernet frames. Annotate the nodes with the according names and the non-trivial edges with the matches performed for this state transition.



b)* Complete the `decap()` action of Listing 1 (Line 26).

```
hdr.eth.etherType = hdr.veth_ext.etherType;
```

c)* Complete the `encap()` action of Listing 1 (Line 31).

```
hdr.veth_ext.etherType = hdr.eth.etherType;
hdr.eth.etherType = 0x8100;
```

d)* The P4 program cannot work correctly without table data containing correct forwarding rules. Give the rules for Switch 1, to correctly encapsulate and forward frames of Clients A and B. Use the information given in Figure 5.1.

Match field(s)	Key	Action	Action data
std_meta.ingress_spec hdr.eth.src	1.a `00:00:00:aa:aa:aa`	encap	egress=1.b pcp=0 dei=0 vid=15
std_meta.ingress_spec hdr.eth.src	1.b `00:00:00:bb:bb:bb`	decap	egress=1.a

0 1 2

e)* The P4 program in Listing 1 is incomplete, an important control block is missing. Name the control block and briefly explain its task.

The deparser is missing. The deparser is used to define how a packet and its headers are reassembled.

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