Chair of Network Architectures and Services
Department of Informatics
Technical University of Munich

Eexam
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 / Retake	**Date:**	Tuesday 30th March, 2021
Examiner:	Prof. Dr.-Ing. Georg Carle	**Time:**	08:00 – 09:15

Working instructions

- This exam consists of **14 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 **non-programmable pocket calculator**
  - 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.

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.

Left room from _____ to _____ / Early submission at _____

Problem 1 Quiz (15.5 credits)

The following questions cover multiple topics and can be solved independently of each other.

0 1 2 3

a)* Briefly explain why traceroute can result in wrong paths if load balancing is used. Support you argument with an example topology and an incorrect path.
You can draw a topology or provide a set of nodes and edges, e.g., S->A, A->E.

0 1

b)* How does ZMap ensure that TCP connections are closed and no state is left on scanned targets?

0 1

c)* Name two reasons from the lecture why Top Lists should be treated carefully.

0 1

d)* Explain what determines the number of entries in the first stage table in the DIR-24-8 or the DXR algorithm.

0 1

e)* A researcher wants to introduce a new kind of network layer protocol. Describe the changes that must be introduced in a OpenFlow-based SDN network architecture to support this new network layer protocol.

0 1 2

f)* Note the IPv6 address 2a03:0:20:0:2810::123 in the longest way possible using the hexadecimal representation with colons.

g)* AS1234 announces the prefix 200.0.0.0/18 via BGP. AS666 wants to hijack the complete prefix. What prefix or prefixes can it announce to successfully hijack the prefix and why?

0 1 2

h)* Explain how Consistent Hashing works in the context of Content Delivery Networks.

0 1 2

i)* Explain anycast-based load balancing.

0 1 2

Problem 2 Transport Protocols (17 credits)

In this problem you are designing a new file transfer protocol. The protocol is named *Super Fast Transfer* (SFT) and is supposed to fast and securely transfer large amounts of data through the network.

0 1

a)* SFT should be operable in the Internet which requires IP as underlying Layer 3 protocol. How many unique **IPv4** and **IPv6** addresses do exist?

Next, you have to decide which Layer 4 protocol you will use. You know that UDP and TCP are widely used. Especially, TCP is one of the most used protocols due to the many features it offers. For example, flow control and congestion control.

0 1

b)* Name 3 **additional** features offered by TCP compared to UDP.

These advantages convinced you to use TCP as Layer 4 protocol.

0 1

c)* To optimize the transmission speed of SFT you want to turn off congestion control. Briefly explain, what problems this can cause.

Knowing this you decide to move away from this idea and are now deciding which congestion control algorithm you want to use.

0 1

d)* TCP Cubic succeeded TCP Reno as default algorithm in the Linux kernel. Name **two advantages** of TCP Cubic over TCP Reno.

0 1

e)* To minimize the transmissions latency you test the delay-based TCP Vegas algorithm with a file transfer in the Internet. However, this results in a rather small transmission rate. Briefly explain the reason for this.

Next, you investigate the TCP BBR congestion control algorithm. Other than Vegas, BBR can combine high transmission rates with low latencies. You know, that BBR estimates BtlBw and RTprop to compute the connections BDP.

0 1 2

f)* How does BBR consider these estimated values in its design goals?

g)* Name **two** ways how a TCP **sender** can detect lost segments.

0 1 2

Beside, TCP you also search for other possibilities for transport layer protocols and find out about SCTP and QUIC. Both try to fix known problems with TCP following a similar approach. While QUIC seems to be quite successful, you hardly find any information about SCTP.

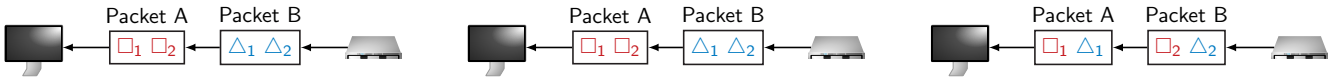
h)* Give two reasons why QUIC succeeds where SCTP failed.

0 1 2

i)* Briefly explain the relation between a UDP datagram, a QUIC frame, and a QUIC packet.

0 1

Multiple data streams can be multiplexed into one TCP connection, for example in HTTP/2. While this reduces the load in the endpoints' operating system which have to maintain less open sockets, it can cause head-of-line (HoL) blocking. In this scenario two files are multiplexed, each of them is split up into two parts, e.g. $\text{file}_1 = (\square_1, \square_2)$ and $\text{file}_2 = (\triangle_1, \triangle_2)$. The data stream is then distributed to the two IP packets Packet A and Packet B.



(a) Scenario 1 HTTP/2 (b) Scenario 2 HTTP/3 (c) Scenario 3 HTTP/3

Figure 2.1: File transfer from Server to Client.

j)* In Figure 2.1 scenarios using HTTP/2 and HTTP/3 are shown. **For each** scenario, briefly explain if or if not loosing **Packet A** can cause HoL blocking.

0 1 2 3

(a) Scenario 1 HTTP/2

(b) Scenario 2 HTTP/3

(c) Scenario 3 HTTP/3

Problem 3 **Network Calculus (10.5 credits)**

This problem investigates Network Calculus and its applications to determine delay bounds in networks. Always document your approach and simplify terms as much as possible, unless specified otherwise.

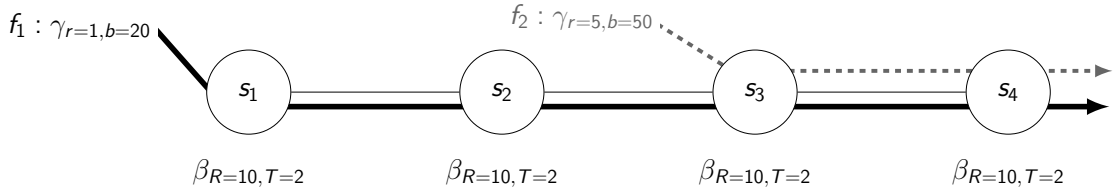


Figure 3.1: Network topology and flow description

Let the output envelope of a token-bucket constrained flow $\gamma_{r,b}$ traversing a rate-latency server $\beta_{R,T}$ be defined as $(\gamma_{r,b} \oslash \beta_{R,T})(t) = \gamma_{r,b+r\cdot T}(t)$
Let the left-over service curve for a token-bucket constrained flow and a rate-latency server be defined as $\beta^{l.o.} = \left[\beta_{R,T} - \gamma_{r,b}\right]^+ = \beta_{R-r,\frac{b+R\cdot T}{R-r}}$
Consider the network shown in Figure 3.1. Assume preemptive Strict Priority Queuing at each server. Flow f_1 has a low priority and Flow f_2 has a high priority. We are interested in calculating the delay bound for Flow f_1 using the Separate Flow Analysis.

Note: You are not required to use special symbols (e.g. $\beta_{R,T}$ and $\otimes$). But make sure your notation is consistent and understandable (e.g., *beta_R,T* and *convolution*).

0 1 2 3 4

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

0 1

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

0 1

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

d) Name the main advantage that can be gained by performing the second step of the Separate Flow Analysis.

0 1

e)* Give an alternative definition **or** explantion of the $[x]^+$ operator that is used during the left-over service curve calculation.

0 1

f)* Assume a token-bucket constrained flow $\gamma_{r=100,b=2}$ traversing a single rate-latency server $\beta_{R=80,T=300}$. Assume arbitrary multiplexing. Argue whether or not a finite delay bound for the flow can be computed.

0 1

Problem 4 DNS (17 credits)

This problem investigates the domain name system (DNS).

```
1     tcb.example.de.          SOA    ns1.tcb.example.de.
2     tcb.example.de.          NS     ns1.tcb.example.de.
3     tcb.example.de.          NS     ns2.tcb.example.de.
4     tcb.example.de.          NS     ns.example.com.
5     ns1.tcb.example.de.      A      192.0.0.1
6     ns2.tcb.example.de       A      192.0.0.2
7
8     example.com.             SOA    ns.example.com.
9     example.com.             NS     ns.example.com.
10    ns.example.com.          A      193.0.0.1
11    qmin.acn.example.com.    A      193.0.0.2
```

Listing 1: Relevant DNS records for this problem. Some records are shortened to the relevant part for readability.

0 1 2

a)* A DNS message starts with the header. Name all other parts of a DNS message besides the header.

0 1

b)* Shortly explain any difference in the message format of DNS queries and responses or how they can be differentiated.

0 1 2

c)* Define the term TCB in general and explain what you can learn from a TCB for a domain name.

0 1

d)* Shortly explain why you would prefer a larger or a smaller TCB size for a domain you own.

0 1 2

e)* The RFC 2182 requires a zones nameserver setup to be redundant and robust. Explain why neither `tcb.example.de` nor `example.com` fulfill this requirements.

f)* Draw the TCB for `tcb.example.de` using the records provided in Listing 1. Consider all other zones to be in-bailiwick. Alternatively to drawing you can also list all necessary connections in the graph.

0 1 2 3

g) Remove one record (use line number for reference) in order to reduce the TCBs size.

0 1

h)* If a resolver performs QNAME-minimization which queries would it perform to resolve `qmin.acn.example.com`? Assume the resolvers cache is empty but you are allowed to omit queries for the nameserver names A/AAAA record. Name the queried domain name and the target nameservers zone.
E.g. Query `google.com` A at nameserver from `google.com`

0 1 2 3

i)* Explain the difference to a resolution path without QNAME minimization and explain why that is.

0 1 2

Problem 5 P4 (15 credits)

This problem investigates a Software-Defined Network (SDN) powered by P4. The source code of a P4 switch program is given in Listing 2.

```
header eth_t { bit<48> dstAddr;
               bit<48> srcAddr;
               bit<16> etherType; }

header ip6_t { bit<4>   version;
               bit<8>   trafficclass;
               bit<20>  flowlabel;
               bit<16>  payloadlength;
               bit<8>   nextheader;
               bit<8>   hoplimit;
               bit<128> srcAddr;
               bit<128> dstAddr; }

header udp_t { bit<16> srcPort;
               bit<16> dstPort;
               bit<16> length;
               bit<16> checksum; }

struct metadata { /* unused */ }

struct headers  { eth_t  eth;
                  ip6_t  ipv6;
                  udp_t  udp; }

parser ParserImpl(packet_in packet, out headers hdr, inout metadata meta, inout standard_metadata_t
    standard_metadata) {

    state parse_udp { packet.extract(hdr.udp);
                      transition accept; }

    state parse_ip6 { packet.extract(hdr.ipv6);
                      transition select(hdr.ipv6.nextheader) { 0x11:    parse_udp;
                                                               default: accept; }}
    state parse_eth { packet.extract(hdr.eth);
                      transition select(hdr.eth.etherType) { 0x86dd:  parse_ip6;
                                                             default: accept; }}
    state start     { transition parse_eth; }

}

control DeparserImpl(packet_out packet, in headers hdr) {
    apply { packet.emit(hdr.eth);
            packet.emit(hdr.ipv6);
            packet.emit(hdr.udp); }
}

control Pipeline(inout headers hdr, inout metadata meta, inout standard_metadata_t standard_metadata) {

    action my_drop()               { mark_to_drop(standard_metadata); }
    action set_egress(bit<9> port) { standard_metadata.egress_spec = port; }
    action set_default_egress()    { standard_metadata.egress_spec = 1; }

    table filter { actions = { set_egress; my_drop; set_default_egress; }
                   key =     { standard_metadata.ingress_port: exact; }
                   default_action = set_default_egress(); }

    table forward { actions = { set_egress; my_drop; set_default_egress; }
                    key =     { standard_metadata.ingress_port: exact; }
                    default_action = set_default_egress(); }

    apply { if (hdr.udp.isValid()) {
                filter.apply();
            } else if (hdr.eth.isValid()) {
                forward.apply();
            }
    }

}

V1Switch(ParserImpl(), Pipeline(), DeparserImpl()) main;
```

Listing 2: Simple P4 program

For the following problems use the network topology given in Figure 5.1. Switch S is a P4 switch running the P4 program of Listing 2.

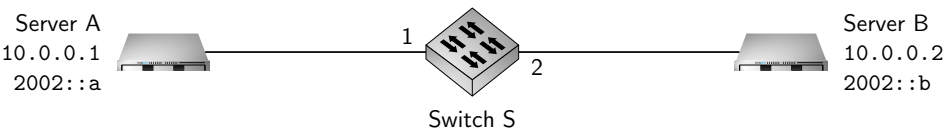


Figure 5.1: Network topology

Match field(s)	Key	Action	Action data
`standard_metadata.ingress_port`	1	`set_egress`	2

Table 5.1: Rule entered into the table `filter`

For the following subproblems you can assume that Servers A and B know the MAC address of each other, i.e., you do not need to describe or consider address resolution in your answers.

The administrator of the network in Figure 5.1 wants to test the connectivity in his network using `traceroute`. Therefore, he executes the command `traceroute 2002::b` on Server A. The traceroute tool of the administrator uses the UDP protocol.

a)* Name at least two other protocols that can be used by the `traceroute 2002::b` command besides UDP.

0 1

b)* Discuss if a normal switch (not a P4-switch) shows up in the output of traceroute.

0 1

c)* Discuss if the P4-switch running the program in Listing 2 shows up in the output of traceroute.

0 1

d)* Explain the path of the packet that is generated by the `traceroute 2002::b` on Server A through the `ParserImpl`.

0 1 2

0 1 2 3

e) Explain the path of the packet that is generated by the `traceroute 2002::b` on Server A through the `Pipeline` of the P4 program in Listing 2, mentioning all passed decisions, tables, table entries, actions, and what happens to the packet.

For the following subproblems you can assume that the packet that is generated by the `traceroute 2002::b` on Server A successfully arrived at Server B.

0 1

f)* What kind of reply is generated by Server B?

0 1 2 3

g) Explain the path of the reply packet generated by Server B through the `Pipeline` of the P4 program in Listing 2, mentioning all passed decisions, tables, table entries, actions, and what happens to the packet.

0 1 2

h)* What happens to the packet if the administrator executes the command `traceroute 10.0.0.2` on Server A (using the UDP protocol)?

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