Chair of Network Architectures and Services
Department of Informatics
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:**	Wednesday 12th February, 2020
Examiner:	Prof. Dr.-Ing. Georg Carle	**Time:**	10:30 – 11:45

	P 1	P 2	P 3	P 4	P 5	P 6	P 7
I							
II							

Working instructions

- This exam consists of **16 pages** with a total of **7 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 (8 credits)

0 1

a)* A **router** achieves an average throughput of 1 million packets per second for traffic with a packet size of 64 B. The same router achieves almost the same average throughput of 1 million packets per second for traffic with a packet size of 256 B. Explain why the packet size has almost no influence on the throughput.

- Router only processes header data.
- Therefore, packet rate rather than payload size it the relevant factor.

0 1

b)* A **VPN gateway** achieves an average throughput of 1 million packets per second for traffic with a packet size of 64 B. The same gateway achieves only a throughput of 0.25 million packets per second for traffic with a packet size of 256 B. Explain why the packet size has this significant impact on throughput.

- VPN gateway encrypts data.
- Encryption is done on the payload of the packets.
- Increasing the packet size by for, quadrupels the amount of payload to encrypt, therefore throughput decreases to one quarter of the original throughput.

0 1

c)* What is the shortest possible representation of the given IPv6 address?

```
2a01:00b0:0000:0000:0000:0a02:0000:2a0f
```

2a01:b0::a02:0:2a0f

0 1

d)* Name and shortly explain the main advantage of QNAME Minimization in DNS.

Advantages:

- Less information given to NS during recursive resolution by only sending the necessary part of the domain name

0 1

e)* Name and shortly explain two disadvantages of QNAME Minimization in DNS.

Disadvantages:

- More queries - Each label is queried
- Higher failure rate - e.g., wrong NS behaviour for empty non terminals

f)* Assume you want to register a domain. Explain the effect a parent zone has on your domain's security.

0
1

The larger the parents zone TCB the larger the domains TCB

A larger TCB is a larger attack surface

g)* An operator for data centers wants to connect two of its data centers using a VXLAN tunnel using an Internet connection from its Internet Service Provider. What is the problem if you want to transmit sensible customer data over the VXLAN tunnel?

0
1

- VXLAN tunnels are not encrypted
- Customer data could be transmitted unecrypted across the public Internet.

h)* Briefly explain the purpose of TCP BBR's Drain phase.

0
1

To remove the queue in the network which was created during Startup phase.

Problem 2 Programmable Packet Processing (10.5 credits)

Computer networks currently shift from single-purpose, fixed-function network devices towards flexible and programmable network devices. This problem investigates the concepts behind these modern network devices.

The researcher Prof. Cleanslate wants to introduce a new **application layer protocol** FancyP. He wants to use programmable network devices to increment a counter contained in the header of FancyP on certain network devices.

0 1

a)* Prof. Cleanslate has 10 hosts and wants to build an OpenFlow-enabled network. Which components does he need to add to build such a network.

- OpenFlow Switch/data plane
- OpenFlow Controller/control plane

0 1

b) Explain the tasks of the entities listed in Problem a).

- Data plane: actual forwarding of packets according to table entries
- Control plane: managing data plane devices (set table entries)

0 1 2

c)* Explain if FancyP **can or cannot** be realized using OpenFlow.

- Protocols must be supported by OpenFlow
- The novel FancyP is not supported by OpenFlow
- Solution 1: Send everything to the control plane and increment counter there.
- Solution 2: To support new protocols the OpenFlow standard must be updated.

Prof. Cleanslate recently heard about a new concept called Network Function Virtualization (NFV). Now he is interested in realizing FancyP with NFV.

0 1 2

d)* Explain if FancyP **can or cannot** be realized using NFV.

- NFV uses off-the-shelf servers running software to perform packet processing
- NFV is able to use FancyP as the protocol can be realized entirely in software

Prof. Cleanslate knows from a colleague that FancyP can be realized using P4.

0 1 2

e)* Name four different P4 targets.

CPU/Software, NPU/Network Processor, FPGA, ASIC

f) There are three potential implementations of FancyP (OpenFlow, NFV & P4). **Rank** the different implementations regarding their forwarding latency and **explain** your ranking.

0 1 2

- Implementations running directly on hardware are typically the fastest
- NFV uses software running on off-the-shelf servers, i.e., a higher latency is expected
- OpenFlow can be realized in hardware and can a lower latency as NFV
- Alternative solution for OpenFlow: If the OpenFlow implementation runs on the control plan, i.e., in software, a higher latency is expected
- P4 can be realized on different targets with the hardware targets having a lower latency

Problem 3 P4 Forwarding (13 credits)

This problem investigates a Software-Defined Network (SDN) powered by P4. The source code of a P4 switch program is given in Listing 1.

```
header eth_t       { bit<48> dstAddr;
                     bit<48> srcAddr;
                     bit<16> etherType; }
header ipv4_t      { bit<4>  version;
                     bit<4>  ihl;
                     bit<8>  tos;
                     bit<16> totalLen;
                     bit<16> identification;
                     bit<3>  flags;
                     bit<13> fragOffset;
                     bit<8>  ttl;
                     bit<8>  protocol;
                     bit<16> hdrChecksum;
                     bit<32> srcAddr;
                     bit<32> dstAddr; }
struct meta        { /* unused */ }
struct headers     { eth_t    eth;
                     ipv4_t   ipv4; }

parser ParserImpl(packet_in packet, out headers hdr, inout meta meta, inout standard_metadata_t
    std_meta) {
  state parse_eth {
    packet.extract(hdr.eth);
    transition select(hdr.ethernet.etherType) {
      ____________: parse_ipv4; // ************ see Problem a)
      default: accept;
    }
  }
  state parse_ipv4 {
    packet.extract(hdr.ipv4);
    transition accept;
  }
  state start {
    transition parse_eth;
  }
}

control Pipeline(inout headers hdr, inout metadata meta, inout standard_metadata_t std_meta) {
  action drop() {
    mark_to_drop();
  }
  action ipv4_fwd(bit<16> egress) {
    std_meta.egress_port = egress;
  }
  table forward {
    actions = {
      ipv4_fwd;
      drop;
    }
    key = {
      std_meta.ingress_port: exact;
      hdr.ipv4.srcAddr: exact;
      hdr.ipv4.dstAddr: exact;
    }
    size = 2;
    default_action = drop();
  }
  apply {
    if (hdr.ipv4.isValid()) {
      forward.apply();
    }
  }
}

control DeparserImpl(packet_out packet, in headers hdr) {
  // ************************************************** see Problem f)
}
```

Listing 1: Simple P4 program

For the following problems use the network topology given in Figure 3.1. Switch 1 is a P4 switch running the P4 program of Listing 1.

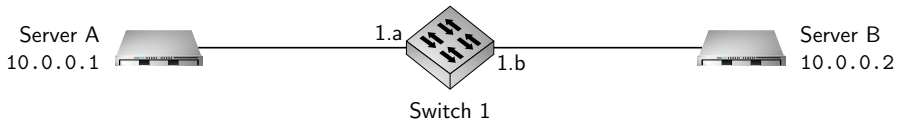


Figure 3.1: Network topology

a)* The parser of Listing 1 (**Line 24**) is incomplete. Fill in the missing value in the underlined area to get a correctly working parser.

0 1

Line 24: 0x0800_____: parse_ipv4;

b)* The P4 program cannot work correctly without table data containing correct forwarding rules. Give the rules for Switch 1 so Servers A and B can communicate with each other via IPv4. Frames not originating from Servers A or B should be dropped. Use the information given in Figure 3.1. You can assume that both servers know the MAC address of their communication partner respectively.

0 1 2 3 4

Match field(s)	Key	Action	Action data
std_meta.ingress_port hdr.ipv4.srcAddr hdr.ipv4.dstAddr	1.a 10.0.0.1 10.0.0.2	ipv4_fwd	egress=1.b
std_meta.ingress_port hdr.ipv4.srcAddr hdr.ipv4.dstAddr	1.b 10.0.0.2 10.0.0.1	ipv4_fwd	egress=1.a

c)* The table definition of the P4 program in Listing 1 (**Line 55**) contains a `default_action = drop()`. What conditions must a packet fulfill to be dropped by this default action?

0 1 2

- Packet must be IPv4 so the table is applied to the packet
- There must be no rule in the table to match exactly against the packet for the following combination: the ingress_port, IPv4 source address and IPv4 destination address

An ARP request arrives at port 1.a of Switch 1.

0 1 2 3

d)* Explain what happens to the ARP request in the parser and both control blocks.

- Parser accepts ARP request(accept in `parse_eth()`)
- Pipeline continues processing(processing stops in `apply` as hdr cannot be matched)
- Switch drops the ARP request
- ARP never reaches the deparser

0 1

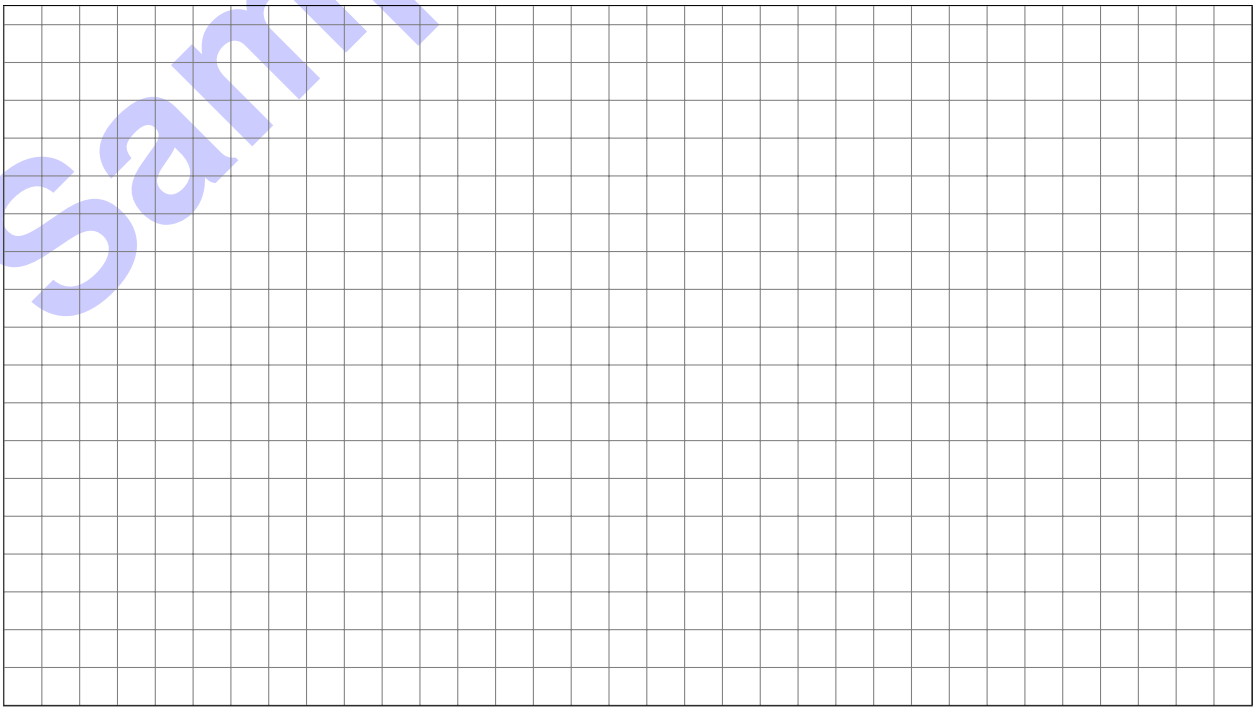
e)* The P4 program in Listing 1 uses the `exact` match type. Name two other match types supported by P4.

- lpm
- ternary
- (range)

0 1 2

f)* Create a valid deparser for the P4 program in Listing 1 (**Line 65)**.

```
control DeparserImpl(packet\_out packet, in headers hdr) {
2     apply { packet.emit(hdr.eth);
              packet.emit(hdr.ipv4);}
4   }
```



Problem 4 Network Calculus (4.5 credits)

This problem investigates the derivation of performance guarantees for a given network using Network Calculus.

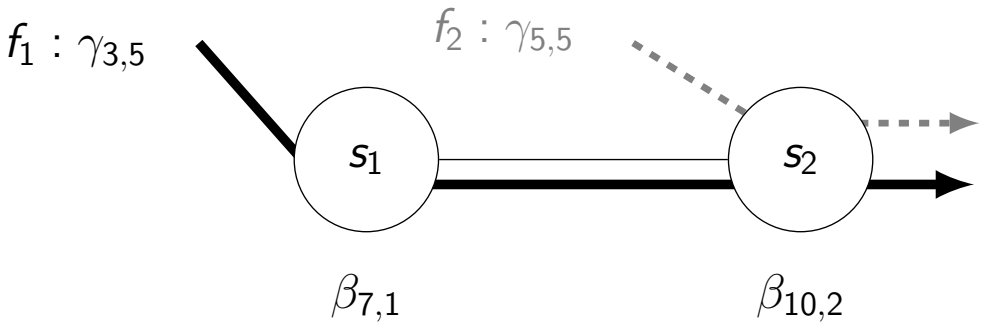


Figure 4.1: Network topology and flow description

Consider the network topology and flow definitions given in Figure 4.1. Flow f_1 traverses the Servers s_1 and s_2. Flow f_2 traverses the Server s_2.
The flows are defined as token-bucket arrival curves $\gamma_{r,b}$ with Rate r and Burst b.
The servers are defined as rate-latency service curves $\beta_{R,T}$ with Rate R and Latency T.
Assume preemptive static priority scheduling at both servers. Furthermore, assume Flow f_1 has a low priority and Flow f_2 has a high priority.

a)* Calculate the residual service curve for Flow f_1 at Server s_1. Specify your end result in the form $\beta_{R,T}$.

0
1

No interfering flows at s_1, therefore:
$\beta_{s_1}^{f_1} = \beta_{s_1} = \beta_{7,1}$

b)* Calculate the residual service curve for Flow f_1 at Server s_2. Specify your end result in the form $\beta_{R,T}$.

0
1

Flow f_2 has higher priority, therefore:
$\beta_{s_2}^{f_1} = \beta_{s_2} - \gamma_{f_2} = \beta_{10,2} - \gamma_{5,5} = [10 \cdot (t-2) - (5 \cdot t + 5)]^+ = [5 \cdot (t-5)]^+ = \beta_{5,5}$

c) Use the concatenation theorem to combine the two residual service curves into a single end-to-end service curve for f_1. Specify your end result in the form $\beta_{R,T}$.

0
1

$\beta_{e2e}^{f_1} = \beta_{s_1}^{f_1} \otimes \beta_{s_2}^{f_1} = \beta_{min(7,5),1+5} = \beta_{5,6}$

d) Calculate the delay bound for Flow f_1 traversing the Servers s_1 and s_2. Make use of the concatenation theorem and consider the scheduling strategy as well as the influence of other flows.
Hint: Re-use results from previous sub-problems.

0
1

Use the concatenation of the two residual service curves.
Under the assumption of rate-latency service curves and token-bucket arrival curves as well as $r \leq R$:
$d = T + \frac{b}{R} = 6 + \frac{5}{5} = 7$

Problem 5 Transport Layer (12.5 credits)

This problem is about transport layer protocols. The Transmission Control Protocol (TCP) is widely used for its reliability property. Lost segments can be detected and are retransmitted by the sender. Beside the reliable data transfer TCP also offers *Flow Control* and *Congestion Control*.

0 1

a)* How can a TCP **receiver** detect lost segments in the byte stream?

All sent bytes are sequentially numbered by the sender (Sequence Number). If numbers are missing at the receiver the segments are assumed to be lost.

0 1

b)* Briefly explain the *Fast Retransmit* algorithm for TCP.

Once the sender receives three duplicate acknowledgemnts it retransmits the acknowledged segment without waiting for the retransmission timer to expire.

0 1

c)* Explain one situation when *Fast Retransmit* is **not** triggered.

- If no packets arrive at the receiver after the lost one.
- If no acknowledgements can arrive at the sender.

0 1

d)* What is the goal of *Flow Control*?

Prevent the sender from overloading the receiver.

0 1

e)* How is the concept of *Flow Control* implemented in TCP?

The TCP header contains a *Window* field.With this field each side of a connection announces its receive window size.

0 1

f)* Name **three** different classes of congestion control algorithms.

- loss-based algorithms
- delay-based algorithms
- model-based algorithms
- hybrid algorithms

0 1

g)* Name one advantage and one disadvantage of TCP Vegas.

TCP Vegas targets the optimal operation point which allows maximum throughput with minimal delay. TCP Vegas performs badly when competing with loss-based algorithms.

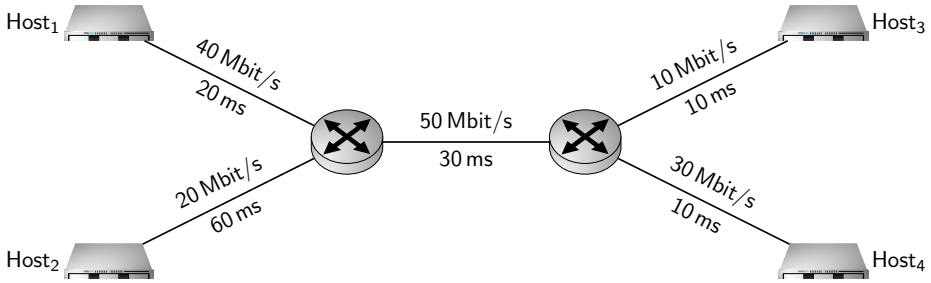


Figure 5.1: The maximum transmission rate and propagation delay is noted for each link.

In the following the network shown in Figure 5.1 is used. There, two TCP flows are started:

- Flow_1 sends data from Host_1 to Host_3
- Flow_2 sends data from Host_2 to Host_4.

Flow_1 ends before Flow_2 starts so you can assume that they do not influence each other.

h)* Compute the *Bandwidth Delay Product* (BDP) for Flow_1 and Flow_2 in kbit.

0 1 2

Flow_1: BDP = 10 Mbit/s · 60 ms = 0.6 Mbit = 600 kbit
Flow_2: BDP = 20 Mbit/s · 100 ms = 2.0 Mbit = 2000 kbit

i)* How is the BDP related to congestion control?

0 1

The BDP equals the optimal amount of data inflight to reach maximal throughput with minimal delay.

In HTTP/2 all files are sent multiplexed over one TCP connection. This can result in Head-of-Line (HoL) Blocking.

j)* Briefly explain how TCP causes HoL Blocking on the transport layer in this case.

0 1

If one segment is lost, all other streams have to wait until the lost segment is retransmitted.

k)* How does the QUIC protocol solve this issue?

0 1

When using QUIC one can have multiple streams within one connection.Each stream can be handled individually and they do not influence each other.

Problem 6 Wireshark (10.5 credits)

According to the OSI model network protocols are distributed to seven different layers each containing several protocols. In this problem a packet is analyzed referring to the involved protocols.

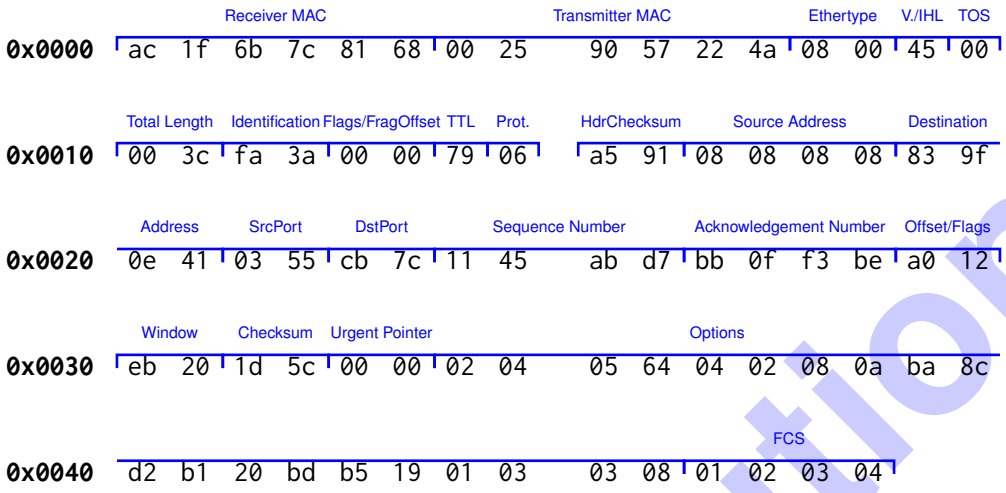


Figure 6.1: Hexdump of a complete Ethernet frame including FCS

0 1 a)* Mark and name all parts of the protocol specific information for layer 2 in Figure 6.1 **Note:** Put your solution directly in Figure 6.1

In the next four subproblems you are asked to identify which protocols were used for each layer. For each question do the following:

- mark the corresponding bytes in the hexdump and
- write the corresponding bytes in the solutionbox.

0 1 b)* Name the L3 protocol.

Ethertype: `0x0800`
IPv4

0 1 c) Name the L4 protocol.

Protocol: `0x06`
TCP

0 1 d) List the flags which are set in the L4 protocol.

`0x12`
SYN, ACK

e) Identify the application layer protocol.

0 1 2

SYN, ACK is a response therefore the source header must be evaluated
`0x0355`= 853
DNS over TLS

f) Determine the length of the L4 header in byte.

0 1

`0xa`= 10 × 32 bit= 40 B

g) Determine the size of the L4 payload in byte.

0 1

All header, and options. No payload.0 B

Problem 7 Internet Measurements (16 credits)

Internet-wide measurements are a major research area in the field of networking. This problem investigates properties and important considerations regarding active network scans.

0 1

a)* Name two ethical considerations relevant for Internet wide measurements.

- scan with a moderate rate
- distribute the load
- do not publish without anonymization and/or limited access
- inform about the scanning and react to complaints

0 1

b)* Explain why non BGP announced Prefixes should not be scanned.

The next BGP router will be loaded with unroutable packets. High load of ICMP destination unreachable messages.

0 1 2 3

c)* Given we want to scan the routable addresses in 138.0.0.0/8, how many targets are we going to scan with the prefixes in following table having an entry in the routing table?

Routed Prefixes
138. 64. 0.0/11
8. 0. 0.0/12
138. 0. 0.0/12
130.120. 0.0/16
138.132. 0.0/16
138. 30. 0.0/20
138.138. 0.0/20
138. 0. 2.0/22
138.138.12.0/22

Hint: The table might contain more specific announcements.

Hint: The answer does not need to compute the actual number of targets, the computation including power of two terms is also valid. Your solution approach should be comprehensible.

$2^{21}(for 138.64.0.0/11)+2^{20}(for 138.0.0.0/12)+2^{16}(for 138.132.0.0/16)+2^{12}(for 138.138.0.0/20)+2^{12}(for 138.30.0.0)$

0 1

d)* Assume that we now want to scan a /**8** subnet in IPv6. Explain why that is not feasible.

Too many addresses$2^{128-8} = 2^{120} = 2^{96} * 2^{24}$ (Number of IPv4 targets w/o blacklist)

0 1

e)* What Resource Record (RR) Type has to be queried to resolve domain lists to IPv4 and IPv6 addresses respectively.

A and AAAA (not NS!)

f)* Name and briefly explain two effects discussed during the lecture observed in domain top lists.

0 1 2

Weekend effect→ More private, entertainment traffic during the weekend
Clustering effect→ Large ranges with same rank, alphabetically sorted

g)* Name two further input sources for active Internet wide scans besides domain lists

0 1

Two out of e.g., traceroutes, rDNS walking, topology scans

Now consider an ICMP echo request scanner that sends minimal echo requests to each target.

h)* Calculate the total **frame** size of frames send by the ICMP scanner for IPv4 and IPv6.

0 1 2

IPv4: $14B + 20B + 8B + 18B + 4B = 64B$
IPv6: $14B + 40B + 8B + 4B = 66B$

Now, assume the scanner sends IPv4 ICMP packets with a frame size of 100 B and IPv6 ICMP packets with a frame size of 250 B. The scanner is configured with two rate limits. The maximum number of sent packets per second is 10 000 pps. The scanner sends a maximum L2 data rate of 10 Mbit/s.

i)* How many IPv4 targets can be scanned in 1 s and how much data will be sent in 1 s with the given configuration?

0 1 2

- Effective limit is: Packets per second
- $100\,\text{B} * 8 = 800\,\text{bit}$
- $800\,\text{bit} * 10\,000\,1/\text{s} = 8\,\text{Mbit/s} < 10\,\text{Mbit/s}$
- Number of Targets: 10 000
- Sent data: Targets * Framesize -> 8 Mbit

j)* How many IPv6 targets can be scanned in 1 s and how much data will be sent in 1 s with the given configuration?

0 1 2

- Effective limit is: Bandwidth
- $250\,\text{B} * 10\,000\,1/\text{s} = 2.5\,\text{MB/s} > \frac{10\,\text{Mbit/s}}{8} = 1.25\,\text{MB/s}$
- Number of Targets: $\frac{1.25\,\text{MB/s}}{250\,\text{B}} * 1\,\text{s} = 5000$
- Sent data: Targets * Framesize -> $5000 * 250\,\text{B} = 125\,\text{kB}$

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