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

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

Module: IN2097
Examiner: Prof. Dr.-Ing. Georg Carle
Date: Monday 10th April, 2017
Exam: Retake exam

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

Working instructions

- This exam consists of
  - **16 pages** with a total of **5 problems** and
  - a two-sided printed **cheat sheet**.

  Please make sure now that you received a complete copy of the exam.
- Detaching pages from the exam is prohibited.
- 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.
- The total amount of achievable credits in this exam is 60 credits.
- Allowed resources:
  - one **printed dictionary** English ↔ native language
- Physically turn off all electronic devices, put them into your bag and close the bag.

Problem 1 Quiz (6 credits)

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

0 1 2

a)* Explain an important difference between the two different variants of network calculus.

- Deterministic network calculus
- Stochastic network calculus

In deterministic network calculus no randomness is involved, performance bounds are always guaranteed (worst case assumption), whereas stochastic network calculus considers probability for its performance bounds.

0 1

b)* Consider an IP packet with the destination address 192.168.0.100 and a router configured with the given routing table. Matching the destination IP address against the routes would generate three possible matches. How is this ambiguity resolved using Classless Inter-Domain Routing (CIDR)?

The packet is routed according to the most specific, matching entry in this table.

Destination subnet	Destination interface
`192.168.0.0/24`	3
`192.168.0.64/26`	2
`192.168.0.96/27`	1

0 1

c)* In the predecessor of CIDR, the classful networks, ambiguous routes were impossible to create. Explain why.

The address to subnet mapping was fixed for classful networks, i.e. an address can only belong to one specified subnet. Hence, there is only one match possible in a routing table.

0 1

d)* To accelerate an Internet-wide scan, the amount of addresses to be scanned shall be reduced. Explain which addresses can be blacklisted for the scan if you have access to your upstream provider's BGP export.

Exclude the unannounced prefixes from the scan, as they are not reachable via your provider

0 1

e)* You want to save the complete IPv6 address space in an array. How many entries does the array have and how much memory (in TiB) do you need for this array. **Note:** 1 TiB = 1024 GiB

Entries: 2^{128}, needing $2^4 \times 2^{128} = 2^{132}B = 2^{92}TiB$

Problem 2 Loops (6 credits)

This problem is based on the topology depicted in Figure 2.1. The two Hosts H1 and H2 are connected to Switches S1 and S2 respectively. Switches S1 and S2 are connected with two separate cables.

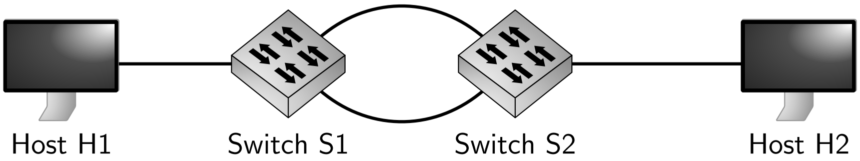


Figure 2.1: Network topology

a)* Hosts H1 and H2 are freshly booted. Host H1 wants to ping Host H2. Explain in detail which problems occur.

□ 0 □ 1 □ 2 □ 3

- Host H1 needs to know the MAC address of Host H2, therefore he requests the MAC address of H2 via ARP (or neighbor solicitation in case IPv6 is used)
- ARP is broadcasted (or multicast for neighbor solicitation) to Switch S1 where it is flooded to the two ports leading to Switch S2
- S2 floods the packet back to S1 (and to H2)
- S1 floods the packet again back to S2...
- Packet runs in a cycle is never discarded, fills up all available buffers, overloading the switches

(Note: Assuming that no L3 addresses are configured at H1 at all, it will most likely send DHCP requests or try to assign itself an APIPA (IPv4) or link-local (IPv6) address. In any case, we end up with the problem described above.)

b) Describe solutions how this problem can be solved on ISO/OSI Layer 2 and Layer 1 respectively.

□ 0 □ 1 □ 2

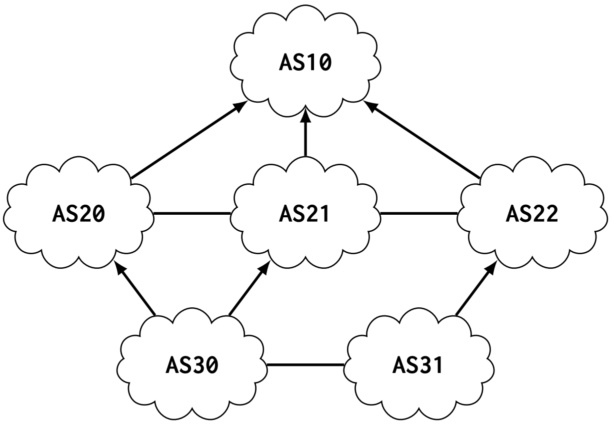
- Removing one cable between the switches to remove cycle in hardware
- Employing an algorithm, e.g. LLDP/STP/SPB to remove cycle in software

c) How is this problem solved for IP?

□ 0 □ 1

Routers do generally not forward L3 broadcasts.
Partially accepted: IP has a TTL / hop limit that ensures packets to not loop indefinitely. However, that mechanism does not prevent a packet from being multiplied resulting in a broadcast storm.

Problem 3 Routing - large scale (12.5 credits)



(a) AS topology

AS	Owned prefixes
AS10	`192.0.10.0/24`
AS20	`192.0.20.0/24`
AS21	`192.0.21.0/24`
AS22	`192.0.22.0/24`
AS30	`192.0.30.0/24`
AS31	`192.0.31.0/24`

(b) Owned prefixes

Figure 3.1: AS topology & prefixes

Figure 3.1a shows a number of different ASes and a topology reflecting their economical dependencies. Figure 3.1b lists the prefixes owned by the ASes.

Arrows show a customer provider relationship where the costumer (arrow bottom) pays money to a provider for transit traffic to the provider (arrow tip).

Furthermore, there are cost free traffic exchange arrangements between partner ASes, symbolized by lines. This agreement only holds for the two directly involved ASes. Every AS wants to get connectivity for its owned prefixes and wants to earn/save money by forwarding traffic to their customers/partners. In this example all providers announce the prefixes of their customers as they earn money from them.

0
½

a)* Give the technical term for an AS that has at least two upstream providers (like AS30 in Figure 3.1a).

Multi-homed (AS)

0
1

b)* How can an AS control the traffic it gets from its partner, provider or costumer ASes?

Selecting announcements allows to control the traffic. An AS can announce different prefixes to its peers than to its customers which influences the traffic it gets from these ASes.

A client with the IP address `192.0.20.10` wants to send a UDP packet to `192.0.31.100` (see Table 3.1b).

0
1

c)* Name the source and destination AS.

Source: AS20, Destination: AS31

d) Describe and justify the path (see Subproblem c)) for every AS the traffic takes through the topology given in Figure 3.1a.

From AS	To AS	Reason
AS20	AS10	AS20 pays AS10 for transit traffic.
AS10	AS22	AS10 earns money by forwarding traffic to AS22.
AS22	AS31	AS22 earns money by forwarding traffic to AS31.

e) There is a shorter way involving less ASes than the solution described in Subproblem d). Give a reason why this path is not taken.

The path AS20 to AS31 over AS30 would be shorter. However, AS30 does not announce this connection, as it would not earn money for forwarding that traffic.

f)* Which prefixes does AS30 announce to AS31, AS20 and AS21? Give short reasons for each announced prefix.

- AS30 to AS31: `192.0.30.0/24`, only own prefixes for connectivity, no upstream to avoid losses.
- AS30 to AS20/21: `192.0.30.0/24` , only own prefixes for connectivity, no peering as transit will not be paid.

g)* Which prefixes does AS21 announce to AS30, AS22 and AS10? Give a short reasons for each announced prefix.

- AS21 to AS10: `192.0.21.0/24`, own prefix for connectivity.
- AS21 to AS10: `192.0.30.0/24`, costumers prefix to earn money.
- AS21 to AS22: `192.0.21.0/24`, own prefix for connectivity.
- AS21 to AS22: `192.0.30.0/24`, costumers prefix to earn money.
- AS21 to AS30: `192.0.21.0/24`, own prefix to earn money.
- AS21 to AS30: `192.0.20.0/24`, `192.0.22.0/24`, peering partners' prefix to earn money.
- AS21 to AS30: `192.0.10.0/24`, upstream prefix as AS30 pays for traffic.
- AS21 to AS30: `192.0.31.0/24`, prefix learned from AS10.

Problem 4 Latency investigation (11.5 credits)

This problem takes a closer look at the latency of a packet processing application. An experiment measures the end-to-end latency of this application.

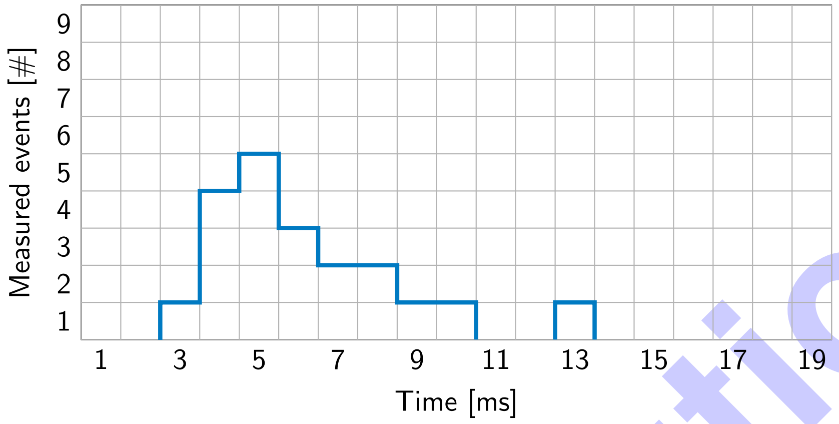


Figure 4.1: Latency distribution

An experiment measures 20 packets with the following latencies [in ms] at the receiving host:

5, 4, 10, 6, 7, 9, 8, 3, 6, 4, 4, 5, 7, 5, 5, 4, 5, 6, 13, 8

0 □□
1 □□
2 □□

a)* Create a histogram in Figure 4.1 for the latency distribution of the given experiment data.

b) Determine the 25%, the 50%, the 75% and the 95% percentile of the previously created histogram in Figure 4.1.

0 □□
1 □□
2 □□

- 25% percentile: lies between the 5th and the 6th element of 20 elements: $(4 + 5)/2 = 4.5$ ms
- 50% percentile: 5.5 ms
- 75% percentile: 7.5 ms
- 95% percentile: 11.5 ms

0 □□
1 □□

c)* Calculate median and average of the histogram in Figure 4.1. Shorten the results as much as possible.

- median: 5.5 ms (same as 50% percentile)
- average: $(3 + 16 + 25 + 18 + 14 + 16 + 9 + 10 + 13)/20 = 124/20 = 6.2$ ms

Assume that the shape of the latency distribution does not change even for longer experiments, i.e., the distribution for 1 million packets looks exactly the same as given in Figure 4.1.

d) Determine the probability of a packet never having a latency of more than 10 ms for a sequence of 100 packets sent. **Note:** Document your approach. You do not need to calculate the actual values.

□ 0
□ 1

- The probability p for a single packet meeting the latency requirements: $p = 0.95$
- For a sequence of 100 packets: 0.95^{100}

e) Determine the probability of **at most one** packet having a latency of more than 10 ms for a sequence of 100 packets sent. **Note:** Document your approach. You do not need to calculate the actual values.

□ 0
□ 1

$0.95^{100} + 100 \cdot 0.05 \cdot 0.95^{99}$

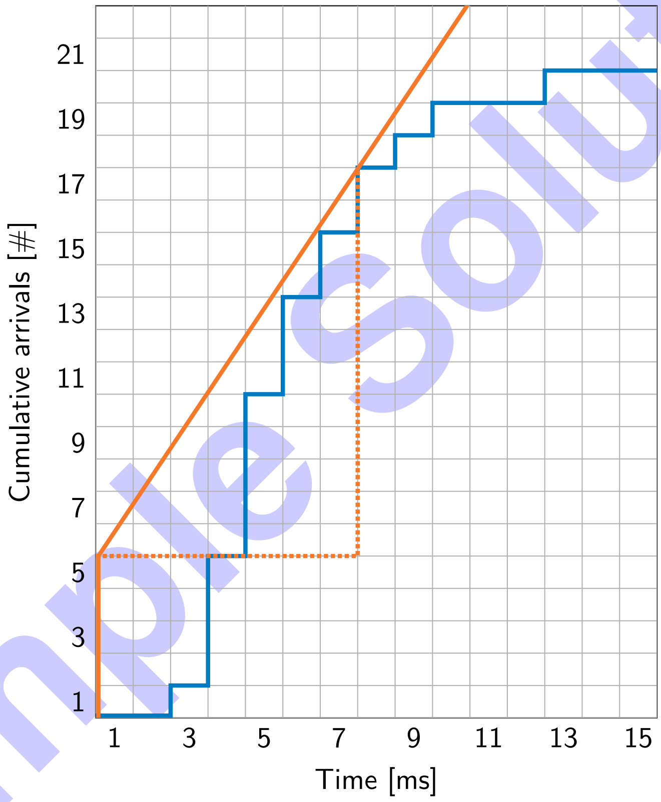


Figure 4.2: Cumulative arrival

f)* Create a graph of the cumulative arrival for the receiving host in Figure 4.2.

□ 0
□ 1
□ 2

g) Create a linear arrival function γ characterized by the rate r and the burstiness parameter b for the cumulative arrival depicted in Figure 4.2. The arrival function should approximate the worst case latency as close as possible and have a rate r which is as low as possible. Give the values for r and b.

□ 0
□ 1
□ 2

- $r = 12/7$
- $b = 5$

Problem 5 Network analysis (24 credits)

Consider the network topology depicted in Figure 5.1, where two hosts H1 and H2 want to communicate with each other. Host H1 accesses the Internet via an SDN-enabled switch and two routers R1 and R2. Host H2 is directly attached to the Internet.

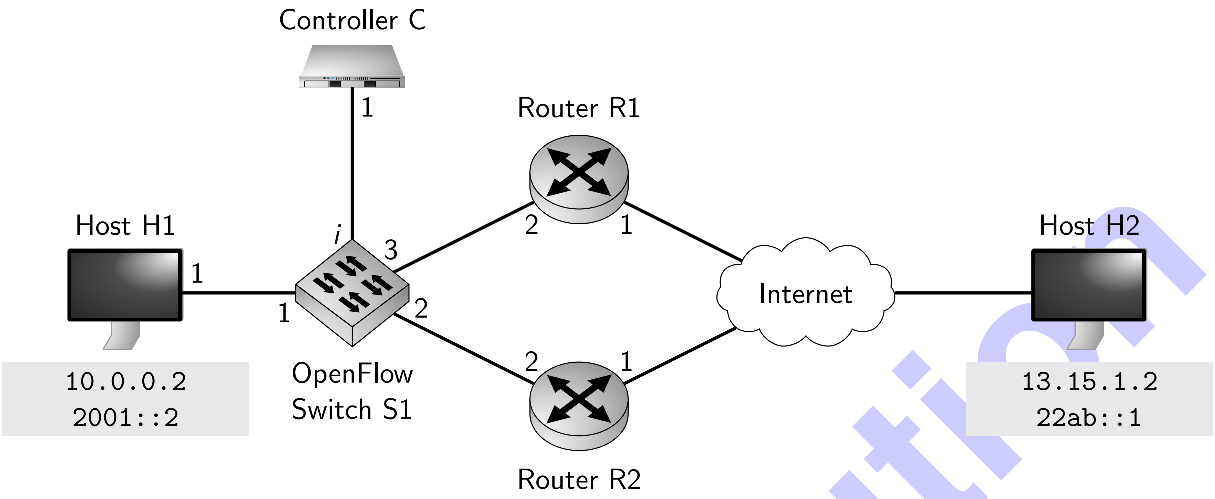


Figure 5.1: Network topology

The OpenFlow switch has a configuration interface named *i* reachable at `192.168.0.2`. Furthermore, the switch has three output ports named 1 to 3. Listing 1 gives the commands which configured this switch.

Listing 1: Open vSwitch commands

```
1 ovs-ofctl add-flow tcp:192.168.0.2 dl_type=0x86dd,nw_dst=22ab::1,priority=10000,actions=output:3
2 ovs-ofctl add-flow tcp:192.168.0.2 dl_type=0x0800,nw_dst=13.15.1.2,priority=10000,actions=output:2
3 ovs-ofctl add-flow tcp:192.168.0.2 priority=0,actions=controller
```

Note:

- To solve this problem, use the cheat sheet that is handed out separately.
- Give a reason for all answers, e.g., answering subproblem h) with “R1” without further comment gives no credit even if the answer might be right

a)* What is the general effect of specifying a `dl_type` in an OpenFlow rule? Where is the `dl_type` specified in Ethernet?

Rules are only used for the specified protocol type, `dl_type` explicitly states the type of the link layer payload. The Ethertype field of the Ethernet header specifies the `dl_type`.

b) Look at Line 1 and Line 2 of the commands shown in Listing 1. What is the effect of the different values for `dl_type`.

- Rule in Line 1 is only valid for `dl_type = 0x86dd` (IPv6).
- Rule in Line 2 is only valid for `dl_type = 0x0800` (IPv4).

c) Look at Line 1 of the commands shown in Listing 1. Explain what this command does and describe what the arguments `tcp:192.168.0.2`, `dl_type`, `nw_dst` and `actions` do in this example.

□ 0 □ 1 □ 2

- This rule specifies to send packets destined to a certain address to be sent out on a specified egress interface of the switch
- `dl_type` specifies IPv6 to be used as payload of the link layer
- `nw_dst` specifies that the destination IP address of the IPv6 packet has to be `2001::1`
- `actions` specify output port 2 as the egress interface of the switch .

d) The IP address of Interface ~~1~~ 2 of Router R2 is not given in Figure 5.1. Give a sensible example for an IP address for that interface.

□ 0

Any address from `10.0.0.0/8` except the address of H1.
Due to error in problem statement also accepted: any public IPv4 address that is not next to 13.15.1.2 (except for good reasoning since there is an Internet in between)

e) Host H1 and Router R2 already know the MAC addresses of each other. Host H1 pings Router R2. Despite ping packets arriving at R2, the response packets are not received by H1. Describe the way of the ping packets from Host H1 to ~~R1~~ R2 and back. Base your explanation on the interfaces (e.g., H1.1) given in Figure 5.1 and the rules specified in Listing 1.

□ 0 □ 1 □ 2

H1.1 → S1.1 → S1.2 → R2.2
R2.2 → S1.2 → S1.i (not matching rule) → C1.1 (drop)

Also accepted due to error in problem statement:

Echo request never reaches R2 since no SDN rule applies. Request is forwarded from S1.1 to S1.i to C and dropped.

Same as intended solution but the initial Echo request is forwarded to C and there either dropped or forwarded to S/R2.

Not accepted: inconsistent forwarding paths to/from R1/2 and forwarding via Internet.

f) What rule(s) has/have to be installed on Switch S1 to receive the replies at Host H1.

□ 0 □ 1

```
ovs-ofctl add-flow tcp:192.168.0.2 dl_type=0x0800,nw_dst=10.0.0.2,priority=10000,actions=output:1
```

For the following problems you can assume that Host H1 and Routers R1 and R2 know each other's MAC addresses. Figure 5.2 shows a hexdump of an Ethernet frame, that Host H1 sent.

```
                    Receiver MAC                    Transmitter MAC          Ethertype  IHL  TOS
0x0000   00  25  90  57  22  4a  68  5b     35  ae  0b  32  08  00  45  00

         Total Length  Identification Flags/FragOffset TTL  Prot.   HdrChecksum  Source Address  Destination
0x0010   00  34  84  57  00  00  01  11     00  00  0a  00  00  02  0d  0f

          Address   SrcPort   DstPort   Length       Checksum         Data
0x0020   01  02  84  56  82  9b  00  20     bb  7e  00  00  00  00  00  00

                                       Options
0x0030   00  00  00  00  00  00  00  00     00  00  00  00  00  00  00  00

                        FCS
0x0040   00  00  5c  d4  f6  00
```

Figure 5.2: Hexdump of Ethernet frame sent by Host H1 including FCS

g)* Mark and name all fields of the Ethernet frame.

h) Argue to which router this packet is forwarded to.

The destination IP and Ethertype match against rule 2, the action is to forward via S1.2 and thus to Router R2.

i) Take a look at the packet of the network layer protocol. What does the router do with such a packet? **Note:** You can assume that all checksums included are correct.

The packet has a TTL of 1. The router does not forward this packet but discards it.
Partially accepted: the router replaces the source IP (private IP / NAT) and forwards it to the Internet.

j) You assume that the packet was generated by the tool `traceroute`. Based on the observed packet, why is this a meaningful assumption.

The packet is sent by Host H1, therefore a TTL of 1 is unreasonably low for real traffic which should reach any real destination.
Partially accepted: UDP payload seems meaningless with ephemeral destination port.

k) Which L4 protocol is used by traceroute in this case?

UDP is used (protocol field `0x11`)

l)* Another protocol which can be used for traceroute is TCP. These packets use port 80 and have the SYN flag set. Why are these meaningful default values?

□ 0 □ 1 □ 2

TCP port 80 is usually not blocked by firewalls. The SYN flag must be set as usually no connection is established between the source and the destination, otherwise stateful firewalls will block such packets.

The router which receives the packet, processes this packet, and generates an answer packet. In the following subproblems you create a hexdump of this answer packet.

Note:

- Use hexadecimal values!
- Do not calculate checksums. Just fill in appropriately sized `0xFF` blocks.
- If values are unknown from Figures 5.1 or 5.2, create sensible values for the fields on your own.

m) Give the hexdump of the Ethernet frame created by the router after processing the packet. Replace the payload of the resulting frame with (. . .).

□ 0 □ 1 □ 2

- src address: `0x68 5b 35 ae 0b 32`
- dst address: `0x00 25 90 57 22 4a`
- Ethertype: `0x08 00`
- FCS: `0xFF FF FF FF`

n) Give the hexdump of the IPv4 header contained as payload for the Ethernet frame generated in Subproblem m).

□ 0 □ 1 □ 2 □ 3

- Version: 0x4
- Length: 0x0038
- TTL: 0x40 , Protocol: 0x01 (ICMP) , Header Checksum: 0xFF FF
- Src Addr: 0x0A 00 00 01 (free of choice however **not** 0x0D 0f 01 02)
- Dst Addr: 0x0A 00 00 02

o) Give the hexdump of the remaining payload.

□ 0 □ 1 □ 2

- ICMP type: `0x0b`
- ICMP code: `0x00`
- ICMP Checksum: `0xFF FF`
- Unused: `0x00 00 00 00`
- IP + UDP header of the original packet

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