

COMPUTER NETWORKS

LAB LIST 4

1 General remarks

The network topology for today's classes is depicted on figure below. Interfaces eth0 are connected to switches. Paired computers are connected to each other via eth1 interface (refer to the figure for cable enumeration).

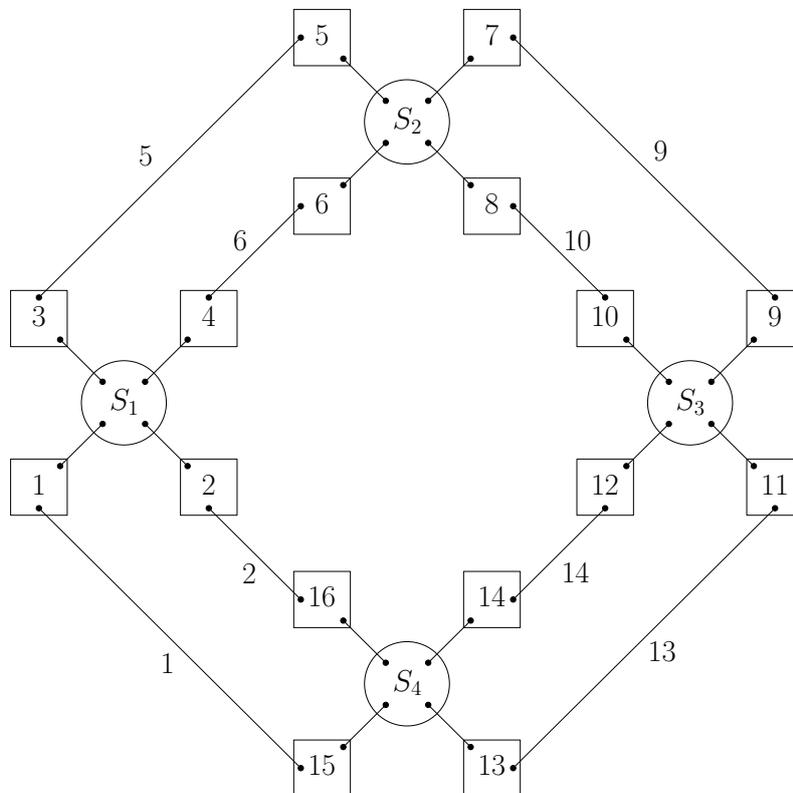


Figure 1: Current network configuration.

Remember to type in `sudo netmode lab` command before you begin.

2 Exercises

Exercise 1. Assign IP addresses $200.0.0.j/29$ to eth0 cards using:

```
#> ifconfig eth0 200.0.0.j netmask 255.255.255.248
```

where j is given by following table:

| | | | | | | | | | | | | | | | |
|---|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 |
| 9 | 10 | 11 | 12 | 17 | 18 | 19 | 20 | 25 | 26 | 27 | 28 | 33 | 34 | 35 | 36 |

Display current state of the routing table:

```
#> route -n
```

It should contain just one entry.

Note that each 4 computers connected to one switch have IP addresses that belong to one network $200.0.0.x/29$, where $x \in \{8, 16, 24, 32\}$. Would it be possible to choose netmask /30 for subnetworks?

Try to ping computers connected to the same switch. What broadcast address should you use? Try to ping an IP address outside of your network $200.0.0.x/29$. You should immediately get following message *network unreachable*, since your computer does not know how to route packet to these networks.

Exercise 2. Assign IP addresses to eth1 cards: for computers a and b connected with a cable, assign IP addresses $200.1.x.a$ and $200.1.x.b$, respectively, where $x = \min\{a, b\}$.

Ping the computer connected by interface eth1. Check the routing table that the entry corresponding to eth1 has netmask $255.255.255.0$ (why?).

Exercise 3. In this exercise, we will pass all packets to a destination in a clockwise manner. As the default gateway choose a computer with number $(x + 2) \bmod 16$, where x is the number of your computer. Refer to the figure on next page, the gateways are marked with incoming arrows. Note that gateway series are organized into two disjoint cycles.

To add an entry to the routing table, one has to enter following command:

```
#> route add network-description gw gateway-ip-address
```

where *network-description* is given in following form:

```
-net network-address netmask network-mask
```

Default gateway is expressed as $0.0.0.0/0$ address. In such case you have to type in:

```
#> route add -net 0.0.0.0 netmask 0.0.0.0 gw gateway-ip-address
```

There is also a handy shortcut „default” which can replace network description given above. So finally, the command takes following form:

```
#> route add default gw gateway-ip-address
```

After configuring the gateway, display the routing table; it should contain three entries. Check whether the default gateway is reachable by the correct interface. Note that we do not have to specify the interface through which the gateway should be reached – it will be deduced automatically on the basis of IP addresses assigned to network cards.

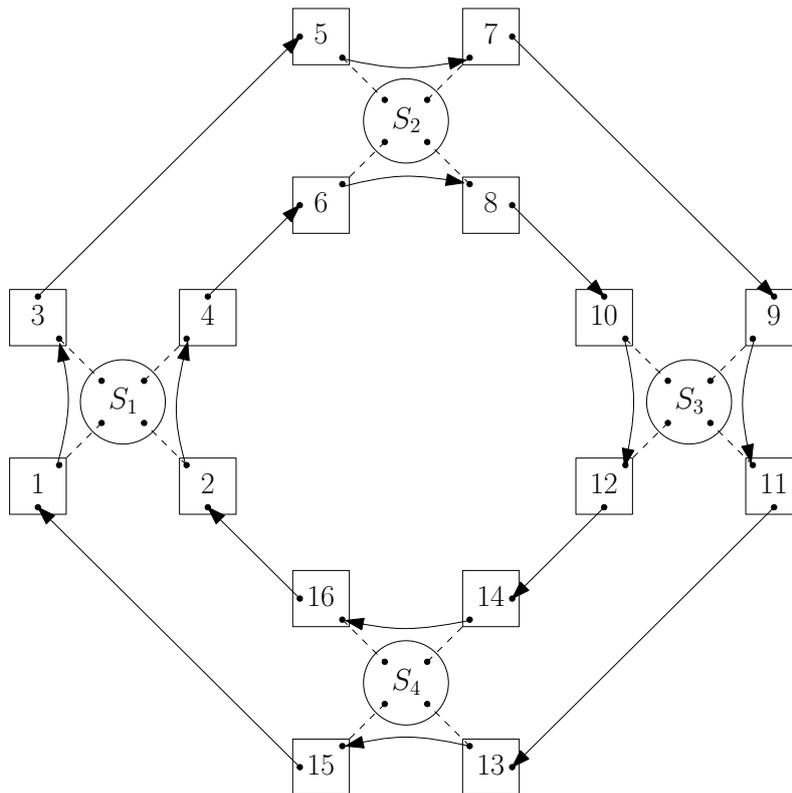


Figure 2: Routing for exercise no. 3.

Exercise 4. Use ping to check reachability of all computers in the network; use IP addresses assigned to their eth0 cards ¹. Following command should do the trick:

```
#> for i in 9 10 11 12 17 18 19 20 25 26 27 28 33 34 35 36; do
    ping -c 1 -W 1 200.0.0.$i | grep -A 1 stat;
done
```

Note that you're not receiving replies from all computers. Why is that? Run *wireshark* program and find all computers sending *ICMP echo requests* addressed to your computer (ignore packets being forwarded through your computer). What are source addresses of the packets? For which requests your computer can reply?

Exercise 5. Choose a relatively distant computer (in terms of route hops), that replied to your ping. Using *traceroute* command, display the route that a packet takes to reach the distant computer. Note that *traceroute* shows IP addresses that reply with *ICMP echo reply*, and not IP addresses that receive a packet with TTL field equal to 0.

Can you answer following question: „Why such a route is taken and not some other one?”

¹ Note that standard Linux kernel configuration disallows packet forwarding between network interfaces. However, computers in the lab room have kernel variable `net.ipv4.ip_forward` set to 1.

Exercise 6. Before starting this exercise make sure that everyone is done with the previous one.

Remove all default gateway entries:

```
route del default
```

Then add entries to the routing table, defining how to reach 10 out of 12 subnetworks (remaining two are reachable directly and corresponding entries are already in the routing table). Look at figure [1] again. Determine the shortest route from your computer the subnetworks and configure the routing table correspondingly (i.e. take such network card's IP address, that the path to the destination is the shortest). Otherwise you're likely to run into problems that will manifestate through packets dying (TTL) as they travel in a loop.

Add entries for subnetworks connected directly to switches:

```
#> route add -net 200.0.0.x netmask 255.255.255.248 gw ip-address
```

and entries for point-to-point networks:

```
#> route add -net 200.1.y.0 netmask 255.255.255.0 gw ip-address
```

Ping all eth0 and eth1 interfaces (all should be reachable now). What is the output of traceroute right now?

Marcin Bieńkowski
Translation: *Krzysztof Białkowski*