Computer Networks

Exercise Session 11

Prof. Dr. Oliver Hahm

Frankfurt University of Applied Sciences
Faculty 2: Computer Science and Engineering
oliver.hahm@fb2.fra-uas.de
https://teaching.dahahm.de

January 20, 2023

General Schedule

All exercises will follow this general schedule

- Identify potential understanding problems
  → Ask your questions
  → Recap of the lecture
- Address the understanding problems
  → Answer your questions
  → Repeat certain topics
- Walk through the exercises/solutions → Some hints and guidance
  → Work time or presentation of results

ICMP

You have seen . . .

- the purpose of the Internet Control Message Protocol
- typical situations where ICMP messages are sent
- which ICMP message types are frequently used

Address Autoconfiguration

You have seen ...

- how to use Reverse ARP to automatically configure IPv4 addresses
- DHCP was introduced as a more feature-reach replacement
- that in any case a device may generate a link-local address
- how SLAAC is used for IPv6 networks to autoconfigure a network device

Inter-Networking

You have seen ...

- how different networks are connected via a router
- which mechanisms are involved when forwarding a packet to a different network
- what an AS is
- the difference between routing and forwarding

Network Layer: Routing Schemes

You have seen ...

- the requirements for a routing protocol
- how routing algorithms can be categorized
- flooding and hot-potato as examples for local routing algorithms
- the difference between source routing and hop-by-hop routing
- the difference between reactive and proactive routing algorithms
- how metrics are used to calculate the path costs

Exercise 1: IPv4 Addressing

IP Address:	151.175.31.100	10010111.10101111.00011111.01100100
Subnet mask:	255.255.254.0	11111111.11111111.11111110.00000000
Network address?	___.___.___.___	________.________.________.________
First host address?	___.___.___.___	________.________.________.________
Last host address?	___.___.___.___	________.________.________.________
Broadcast address?	___.___.___.___	________.________.________.________

Exercise 1: IPv4 Addressing

IP Address:	151.175.31.100	10010111.10101111.00011111.01100100
Subnet mask:	255.255.254.0	11111111.11111111.11111110.00000000
Network address?	151.175.30.0	10010111.10101111.00011110.00000000
First host address?	___.___.___.___	________.________.________.________
Last host address?	___.___.___.___	________.________.________.________
Broadcast address?	___.___.___.___	________.________.________.________

Exercise 1: IPv4 Addressing

IP Address:	151.175.31.100	10010111.10101111.00011111.01100100
Subnet mask:	255.255.254.0	11111111.11111111.11111110.00000000
Network address?	151.175.30.0	10010111.10101111.00011110.00000000
First host address?	151.175.30.1	10010111.10101111.00011110.00000001
Last host address?	___.___.___.___	________.________.________.________
Broadcast address?	___.___.___.___	________.________.________.________

Exercise 1: IPv4 Addressing

IP Address:	151.175.31.100	10010111.10101111.00011111.01100100
Subnet mask:	255.255.254.0	11111111.11111111.11111110.00000000
Network address?	151.175.30.0	10010111.10101111.00011110.00000000
First host address?	151.175.30.1	10010111.10101111.00011110.00000001
Last host address?	151.175.31.254	10010111.10101111.00011111.11111110
Broadcast address?	___.___.___.___	________.________.________.________

Exercise 1: IPv4 Addressing

IP Address:	151.175.31.100	10010111.10101111.00011111.01100100
Subnet mask:	255.255.254.0	11111111.11111111.11111110.00000000
Network address?	151.175.30.0	10010111.10101111.00011110.00000000
First host address?	151.175.30.1	10010111.10101111.00011110.00000001
Last host address?	151.175.31.254	10010111.10101111.00011111.11111110
Broadcast address?	151.175.31.255	10010111.10101111.00011111.11111111

Exercise 1: IPv4 Addressing

IP Address:	151.175.31.100	10010111.10101111.00011111.01100100
Subnet mask:	255.255.255.240	11111111.11111111.11111111.11110000
Network address?	___.___.___.___	________.________.________.________
First host address?	___.___.___.___	________.________.________.________
Last host address?	___.___.___.___	________.________.________.________
Broadcast address?	___.___.___.___	________.________.________.________

Exercise 1: IPv4 Addressing

IP Address:	151.175.31.100	10010111.10101111.00011111.01100100
Subnet mask:	255.255.255.240	11111111.11111111.11111111.11110000
Network address?	151.175.31.96	10010111.10101111.00011111.01100000
First host address?	___.___.___.___	________.________.________.________
Last host address?	___.___.___.___	________.________.________.________
Broadcast address?	___.___.___.___	________.________.________.________

Exercise 1: IPv4 Addressing

IP Address:	151.175.31.100	10010111.10101111.00011111.01100100
Subnet mask:	255.255.255.240	11111111.11111111.11111111.11110000
Network address?	151.175.31.96	10010111.10101111.00011111.01100000
First host address?	151.175.31.97	10010111.10101111.00011111.01100001
Last host address?	___.___.___.___	________.________.________.________
Broadcast address?	___.___.___.___	________.________.________.________

Exercise 1: IPv4 Addressing

IP Address:	151.175.31.100	10010111.10101111.00011111.01100100
Subnet mask:	255.255.255.240	11111111.11111111.11111111.11110000
Network address?	151.175.31.96	10010111.10101111.00011111.01100000
First host address?	151.175.31.97	10010111.10101111.00011111.01100001
Last host address?	151.175.31.110	10010111.10101111.00011111.01101110
Broadcast address?	___.___.___.___	________.________.________.________

Exercise 1: IPv4 Addressing

IP Address:	151.175.31.100	10010111.10101111.00011111.01100100
Subnet mask:	255.255.255.240	11111111.11111111.11111111.11110000
Network address?	151.175.31.96	10010111.10101111.00011111.01100000
First host address?	151.175.31.97	10010111.10101111.00011111.01100001
Last host address?	151.175.31.110	10010111.10101111.00011111.01101110
Broadcast address?	151.175.31.111	10010111.10101111.00011111.01101111

Exercise 1: IPv4 Addressing

IP Address:	151.175.31.100	10010111.10101111.00011111.01100100
Subnet mask:	255.255.255.128	11111111.11111111.11111111.10000000
Network address?	___.___.___.___	________.________.________.________
First host address?	___.___.___.___	________.________.________.________
Last host address?	___.___.___.___	________.________.________.________
Broadcast address?	___.___.___.___	________.________.________.________

Exercise 1: IPv4 Addressing

IP Address:	151.175.31.100	10010111.10101111.00011111.01100100
Subnet mask:	255.255.255.128	11111111.11111111.11111111.10000000
Network address?	151.175.31.0	10010111.10101111.00011111.00000000
First host address?	___.___.___.___	________.________.________.________
Last host address?	___.___.___.___	________.________.________.________
Broadcast address?	___.___.___.___	________.________.________.________

Exercise 1: IPv4 Addressing

IP Address:	151.175.31.100	10010111.10101111.00011111.01100100
Subnet mask:	255.255.255.128	11111111.11111111.11111111.10000000
Network address?	151.175.31.0	10010111.10101111.00011111.00000000
First host address?	151.175.31.1	10010111.10101111.00011111.00000001
Last host address?	___.___.___.___	________.________.________.________
Broadcast address?	___.___.___.___	________.________.________.________

Exercise 1: IPv4 Addressing

IP Address:	151.175.31.100	10010111.10101111.00011111.01100100
Subnet mask:	255.255.255.128	11111111.11111111.11111111.10000000
Network address?	151.175.31.0	10010111.10101111.00011111.00000000
First host address?	151.175.31.1	10010111.10101111.00011111.00000001
Last host address?	151.175.31.126	10010111.10101111.00011111.01111110
Broadcast address?	___.___.___.___	________.________.________.________

Exercise 1: IPv4 Addressing

IP Address:	151.175.31.100	10010111.10101111.00011111.01100100
Subnet mask:	255.255.255.128	11111111.11111111.11111111.10000000
Network address?	151.175.31.0	10010111.10101111.00011111.00000000
First host address?	151.175.31.1	10010111.10101111.00011111.00000001
Last host address?	151.175.31.126	10010111.10101111.00011111.01111110
Broadcast address?	151.175.31.127	10010111.10101111.00011111.01111111

Exercise 2.1: Inter-Networking

Sender:	11001001.00010100.11011110.00001101	201.20.222.13
Subnet mask:	11111111.11111111.11111111.11110000	255.255.255.240
Receiver:	11001001.00010100.11011110.00010001	201.20.222.17
Subnet mask:	11111111.11111111.11111111.11110000	255.255.255.240
Sender:	00001111.11001000.01100011.00010111	15.200.99.23
Subnet mask:	11111111.11000000.00000000.00000000	255.192.0.0
Receiver:	00001111.11101111.00000001.00000001	15.239.1.1
Subnet mask:	11111111.11000000.00000000.00000000	255.192.0.0

Exercise 2.1: Inter-Networking

Sender:	11001001.00010100.11011110.00001101	201.20.222.13
Subnet mask:	11111111.11111111.11111111.11110000	255.255.255.240
	11001001.00010100.11011110.00000000	=> Subnet ID: 0
Receiver:	11001001.00010100.11011110.00010001	201.20.222.17
Subnet mask:	11111111.11111111.11111111.11110000	255.255.255.240

Sender:	00001111.11001000.01100011.00010111	15.200.99.23
Subnet mask:	11111111.11000000.00000000.00000000	255.192.0.0
Receiver:	00001111.11101111.00000001.00000001	15.239.1.1
Subnet mask:	11111111.11000000.00000000.00000000	255.192.0.0

Exercise 2.1: Inter-Networking

Sender:	11001001.00010100.11011110.00001101	201.20.222.13
Subnet mask:	11111111.11111111.11111111.11110000	255.255.255.240
	11001001.00010100.11011110.0000**0000** => Subnet ID: 0	
Receiver:	11001001.00010100.11011110.00010001	201.20.222.17
Subnet mask:	11111111.11111111.11111111.11110000	255.255.255.240
	11001001.00010100.11011110.0001**0000** => Subnet ID: 1	
Sender:	00001111.11001000.01100011.00010111	15.200.99.23
Subnet mask:	11111111.11000000.00000000.00000000	255.192.0.0
Receiver:	00001111.11101111.00000001.00000001	15.239.1.1
Subnet mask:	11111111.11000000.00000000.00000000	255.192.0.0

Exercise 2.1: Inter-Networking

```
Sender:         11001001.00010100.11011110.00001101    201.20.222.13
Subnet mask:    11111111.11111111.11111111.11110000    255.255.255.240
                11001001.00010100.11011110.00000000 => Subnet ID: 0

Receiver:       11001001.00010100.11011110.00010001    201.20.222.17
Subnet mask:    11111111.11111111.11111111.11110000    255.255.255.240
                11001001.00010100.11011110.00010000 => Subnet ID: 1
```

The packet leaves the subnet and needs to be routed.

```
Sender:         00001111.11001000.01100011.00010111    15.200.99.23
Subnet mask:    11111111.11000000.00000000.00000000    255.192.0.0


Receiver:       00001111.11101111.00000001.00000001    15.239.1.1
Subnet mask:    11111111.11000000.00000000.00000000    255.192.0.0
```

Exercise 2.1: Inter-Networking

```
Sender:         11001001.00010100.11011110.00001101    201.20.222.13
Subnet mask:    11111111.11111111.11111111.11110000    255.255.255.240
                11001001.00010100.11011110.00000000 => Subnet ID: 0

Receiver:       11001001.00010100.11011110.00010001    201.20.222.17
Subnet mask:    11111111.11111111.11111111.11110000    255.255.255.240
                11001001.00010100.11011110.00010000 => Subnet ID: 1
```

The packet leaves the subnet and needs to be routed.

```
Sender:         00001111.11001000.01100011.00010111    15.200.99.23
Subnet mask:    11111111.11000000.00000000.00000000    255.192.0.0
                00001111.11000000.00000000.00000000 => Subnet ID: 3

Receiver:       00001111.11101111.00000001.00000001    15.239.1.1
Subnet mask:    11111111.11000000.00000000.00000000    255.192.0.0
```

Exercise 2.1: Inter-Networking

```
Sender:         11001001.00010100.11011110.00001101    201.20.222.13
Subnet mask:    11111111.11111111.11111111.11110000    255.255.255.240
                11001001.00010100.11011110.00000000 => Subnet ID: 0

Receiver:       11001001.00010100.11011110.00010001    201.20.222.17
Subnet mask:    11111111.11111111.11111111.11110000    255.255.255.240
                11001001.00010100.11011110.00010000 => Subnet ID: 1
```

The packet leaves the subnet and needs to be routed.

```
Sender:         00001111.11001000.01100011.00010111    15.200.99.23
Subnet mask:    11111111.11000000.00000000.00000000    255.192.0.0
                00001111.11000000.00000000.00000000 => Subnet ID: 3

Receiver:       00001111.11101111.00000001.00000001    15.239.1.1
Subnet mask:    11111111.11000000.00000000.00000000    255.192.0.0
                00001111.11000000.00000000.00000000 => Subnet ID: 3
```

Exercise 2.1: Inter-Networking

```
Sender:        11001001.00010100.11011110.00001101    201.20.222.13
Subnet mask:   11111111.11111111.11111111.11110000    255.255.255.240
               11001001.00010100.11011110.00000000 => Subnet ID: 0

Receiver:      11001001.00010100.11011110.00010001    201.20.222.17
Subnet mask:   11111111.11111111.11111111.11110000    255.255.255.240
               11001001.00010100.11011110.00010000 => Subnet ID: 1
```

The packet leaves the subnet and needs to be routed.

```
Sender:        00001111.11001000.01100011.00010111    15.200.99.23
Subnet mask:   11111111.11000000.00000000.00000000    255.192.0.0
               00001111.11000000.00000000.00000000 => Subnet ID: 3

Receiver:      00001111.11101111.00000001.00000001    15.239.1.1
Subnet mask:   11111111.11000000.00000000.00000000    255.192.0.0
               00001111.11000000.00000000.00000000 => Subnet ID: 3
```

The packet does not leave the subnet and can be sent directly on the link layer.

Exercise 2.2: Inter-Networking

```
Kernel IP routing table
Destination     Gateway         Genmask         Flags   MSS Window  irtt Iface
0.0.0.0         10.2.0.1        0.0.0.0         UG        0 0          0 eth0
10.2.0.0        0.0.0.0         255.255.255.0   U         0 0          0 eth1
10.204.0.0      0.0.0.0         255.252.0.0     U         0 0          0 wlan0
10.200.0.0      0.0.0.0         255.248.0.0     U         0 0          0 eth2
172.17.8.15     0.0.0.0         255.255.255.255 UH        0 0          0 eth2
192.168.23.0    0.0.0.0         255.255.255.0   U         0 0          0 wlan1
192.168.42.0    0.0.0.0         255.255.255.240 U         0 0          0 eth3
```

1. `192.168.23.14`
2. `192.168.42.17`
3. `192.168.42.15`
4. `10.2.0.255`
5. `10.207.51.4`
6. `172.17.8.18`
7. `172.17.8.15`
8. `10.202.4.3`
9. `10.216.168.23`

Exercise 2.2: Inter-Networking

```
Kernel IP routing table
Destination     Gateway         Genmask         Flags   MSS Window  irtt Iface
0.0.0.0         10.2.0.1        0.0.0.0         UG        0 0          0 eth0
10.2.0.0        0.0.0.0         255.255.255.0   U         0 0          0 eth1
10.204.0.0      0.0.0.0         255.252.0.0     U         0 0          0 wlan0
10.200.0.0      0.0.0.0         255.248.0.0     U         0 0          0 eth2
172.17.8.15     0.0.0.0         255.255.255.255 UH        0 0          0 eth2
192.168.23.0    0.0.0.0         255.255.255.0   U         0 0          0 wlan1
192.168.42.0    0.0.0.0         255.255.255.240 U         0 0          0 eth3
```

1. 192.168.23.14
   → wlan1
2. 192.168.42.17
3. 192.168.42.15
4. 10.2.0.255
5. 10.207.51.4
6. 172.17.8.18
7. 172.17.8.15
8. 10.202.4.3
9. 10.216.168.23

Exercise 2.2: Inter-Networking

```
Kernel IP routing table
Destination     Gateway         Genmask         Flags   MSS Window  irtt Iface
0.0.0.0         10.2.0.1        0.0.0.0         UG        0 0          0 eth0
10.2.0.0        0.0.0.0         255.255.255.0   U         0 0          0 eth1
10.204.0.0      0.0.0.0         255.252.0.0     U         0 0          0 wlan0
10.200.0.0      0.0.0.0         255.248.0.0     U         0 0          0 eth2
172.17.8.15     0.0.0.0         255.255.255.255 UH        0 0          0 eth2
192.168.23.0    0.0.0.0         255.255.255.0   U         0 0          0 wlan1
192.168.42.0    0.0.0.0         255.255.255.240 U         0 0          0 eth3
```

1. `192.168.23.14`
   → wlan1
2. `192.168.42.17`
   → eth0 → default route
3. `192.168.42.15`
4. `10.2.0.255`
5. `10.207.51.4`
6. `172.17.8.18`
7. `172.17.8.15`
8. `10.202.4.3`
9. `10.216.168.23`

Exercise 2.2: Inter-Networking

```
Kernel IP routing table
Destination     Gateway         Genmask         Flags   MSS Window  irtt Iface
0.0.0.0         10.2.0.1        0.0.0.0         UG        0 0          0 eth0
10.2.0.0        0.0.0.0         255.255.255.0   U         0 0          0 eth1
10.204.0.0      0.0.0.0         255.252.0.0     U         0 0          0 wlan0
10.200.0.0      0.0.0.0         255.248.0.0     U         0 0          0 eth2
172.17.8.15     0.0.0.0         255.255.255.255 UH        0 0          0 eth2
192.168.23.0    0.0.0.0         255.255.255.0   U         0 0          0 wlan1
192.168.42.0    0.0.0.0         255.255.255.240 U         0 0          0 eth3
```

1. `192.168.23.14`
   → wlan1
2. `192.168.42.17`
   → eth0 → default route
3. `192.168.42.15`
   → eth3
4. `10.2.0.255`
5. `10.207.51.4`
6. `172.17.8.18`
7. `172.17.8.15`
8. `10.202.4.3`
9. `10.216.168.23`

Exercise 2.2: Inter-Networking

```
Kernel IP routing table
Destination     Gateway         Genmask         Flags   MSS Window  irtt Iface
0.0.0.0         10.2.0.1        0.0.0.0         UG        0 0          0 eth0
10.2.0.0        0.0.0.0         255.255.255.0   U         0 0          0 eth1
10.204.0.0      0.0.0.0         255.252.0.0     U         0 0          0 wlan0
10.200.0.0      0.0.0.0         255.248.0.0     U         0 0          0 eth2
172.17.8.15     0.0.0.0         255.255.255.255 UH        0 0          0 eth2
192.168.23.0    0.0.0.0         255.255.255.0   U         0 0          0 wlan1
192.168.42.0    0.0.0.0         255.255.255.240 U         0 0          0 eth3
```

1. `192.168.23.14`
   → wlan1
2. `192.168.42.17`
   → eth0 → default route
3. `192.168.42.15`
   → eth3
4. `10.2.0.255`
   → eth1
5. `10.207.51.4`
6. `172.17.8.18`
7. `172.17.8.15`
8. `10.202.4.3`
9. `10.216.168.23`

Exercise 2.2: Inter-Networking

```
Kernel IP routing table
Destination     Gateway         Genmask         Flags   MSS Window  irtt Iface
0.0.0.0         10.2.0.1        0.0.0.0         UG        0 0          0 eth0
10.2.0.0        0.0.0.0         255.255.255.0   U         0 0          0 eth1
10.204.0.0      0.0.0.0         255.252.0.0     U         0 0          0 wlan0
10.200.0.0      0.0.0.0         255.248.0.0     U         0 0          0 eth2
172.17.8.15     0.0.0.0         255.255.255.255 UH        0 0          0 eth2
192.168.23.0    0.0.0.0         255.255.255.0   U         0 0          0 wlan1
192.168.42.0    0.0.0.0         255.255.255.240 U         0 0          0 eth3
```

1. 192.168.23.14
   → wlan1
2. 192.168.42.17
   → eth0 → default route
3. 192.168.42.15
   → eth3
4. 10.2.0.255
   → eth1
5. 10.207.51.4
   → wlan0
6. 172.17.8.18
7. 172.17.8.15
8. 10.202.4.3
9. 10.216.168.23

Exercise 2.2: Inter-Networking

```
Kernel IP routing table
Destination     Gateway         Genmask         Flags   MSS Window  irtt Iface
0.0.0.0         10.2.0.1        0.0.0.0         UG        0 0          0 eth0
10.2.0.0        0.0.0.0         255.255.255.0   U         0 0          0 eth1
10.204.0.0      0.0.0.0         255.252.0.0     U         0 0          0 wlan0
10.200.0.0      0.0.0.0         255.248.0.0     U         0 0          0 eth2
172.17.8.15     0.0.0.0         255.255.255.255 UH        0 0          0 eth2
192.168.23.0    0.0.0.0         255.255.255.0   U         0 0          0 wlan1
192.168.42.0    0.0.0.0         255.255.255.240 U         0 0          0 eth3
```

1. `192.168.23.14`
   → wlan1
2. `192.168.42.17`
   → eth0 → default route
3. `192.168.42.15`
   → eth3
4. `10.2.0.255`
   → eth1
5. `10.207.51.4`
   → wlan0
6. `172.17.8.18`
   → eth0 → default route
7. `172.17.8.15`
8. `10.202.4.3`
9. `10.216.168.23`

Exercise 2.2: Inter-Networking

```
Kernel IP routing table
Destination     Gateway         Genmask         Flags   MSS Window  irtt Iface
0.0.0.0         10.2.0.1        0.0.0.0         UG        0 0          0 eth0
10.2.0.0        0.0.0.0         255.255.255.0   U         0 0          0 eth1
10.204.0.0      0.0.0.0         255.252.0.0     U         0 0          0 wlan0
10.200.0.0      0.0.0.0         255.248.0.0     U         0 0          0 eth2
172.17.8.15     0.0.0.0         255.255.255.255 UH        0 0          0 eth2
192.168.23.0    0.0.0.0         255.255.255.0   U         0 0          0 wlan1
192.168.42.0    0.0.0.0         255.255.255.240 U         0 0          0 eth3
```

1. `192.168.23.14`
   → wlan1
2. `192.168.42.17`
   → eth0 → default route
3. `192.168.42.15`
   → eth3
4. `10.2.0.255`
   → eth1
5. `10.207.51.4`
   → wlan0
6. `172.17.8.18`
   → eth0 → default route
7. `172.17.8.15`
   → eth2
8. `10.202.4.3`
9. `10.216.168.23`

Exercise 2.2: Inter-Networking

```
Kernel IP routing table
Destination     Gateway         Genmask         Flags   MSS Window  irtt Iface
0.0.0.0         10.2.0.1        0.0.0.0         UG        0 0          0 eth0
10.2.0.0        0.0.0.0         255.255.255.0   U         0 0          0 eth1
10.204.0.0      0.0.0.0         255.252.0.0     U         0 0          0 wlan0
10.200.0.0      0.0.0.0         255.248.0.0     U         0 0          0 eth2
172.17.8.15     0.0.0.0         255.255.255.255 UH        0 0          0 eth2
192.168.23.0    0.0.0.0         255.255.255.0   U         0 0          0 wlan1
192.168.42.0    0.0.0.0         255.255.255.240 U         0 0          0 eth3
```

1. `192.168.23.14`
   → wlan1
2. `192.168.42.17`
   → eth0 → default route
3. `192.168.42.15`
   → eth3
4. `10.2.0.255`
   → eth1
5. `10.207.51.4`
   → wlan0
6. `172.17.8.18`
   → eth0 → default route
7. `172.17.8.15`
   → eth2
8. `10.202.4.3`
   → eth2
9. `10.216.168.23`

Exercise 2.2: Inter-Networking

```
Kernel IP routing table
Destination     Gateway         Genmask         Flags   MSS Window  irtt Iface
0.0.0.0         10.2.0.1        0.0.0.0         UG        0 0          0 eth0
10.2.0.0        0.0.0.0         255.255.255.0   U         0 0          0 eth1
10.204.0.0      0.0.0.0         255.252.0.0     U         0 0          0 wlan0
10.200.0.0      0.0.0.0         255.248.0.0     U         0 0          0 eth2
172.17.8.15     0.0.0.0         255.255.255.255 UH        0 0          0 eth2
192.168.23.0    0.0.0.0         255.255.255.0   U         0 0          0 wlan1
192.168.42.0    0.0.0.0         255.255.255.240 U         0 0          0 eth3
```

1. `192.168.23.14`
   → wlan1
2. `192.168.42.17`
   → eth0 → default route
3. `192.168.42.15`
   → eth3
4. `10.2.0.255`
   → eth1
5. `10.207.51.4`
   → wlan0
6. `172.17.8.18`
   → eth0 → default route
7. `172.17.8.15`
   → eth2
8. `10.202.4.3`
   → eth2
9. `10.216.168.23`
   → eth0 → default route

Exercise 3: Subnetting

1. Split into 30 subnets:
   `Network ID: 11000011.00000001.00011111.00000000 = 195.1.31.0`
   `Number of bits for subnet IDs?`
   `Subnet mask:`
   `Number of bits for host IDs?`
   `Number of host IDs per subnet?`

Exercise 3: Subnetting

1. Split into 30 subnets:
   Network ID: 11000011.00000001.00011111.00000000 = 195.1.31.0
   Number of bits for subnet IDs? 30 => 32 $=2^5$ => 5 bits
   Subnet mask:
   Number of bits for host IDs?
   Number of host IDs per subnet?

Exercise 3: Subnetting

1. Split into 30 subnets:
   Network ID: 11000011.00000001.00011111.00000000 = 195.1.31.0
   Number of bits for subnet IDs? 30 => 32 $=2^5$ => 5 bits
   Subnet mask: 11111111.11111111.11111111.11111000
   Number of bits for host IDs?
   Number of host IDs per subnet?

Exercise 3: Subnetting

1. Split into 30 subnets:

   Network ID: 11000011.00000001.00011111.00000000 = 195.1.31.0

   Number of bits for subnet IDs? 30 => 32 $=2^5$ => 5 bits

   Subnet mask: 11111111.11111111.11111111.11111000

   Number of bits for host IDs? 3

   Number of host IDs per subnet?

Exercise 3: Subnetting

1. Split into 30 subnets:
   Network ID: 11000011.00000001.00011111.00000000 = 195.1.31.0
   Number of bits for subnet IDs? 30 => 32 $=2^5$ => 5 bits
   Subnet mask: 11111111.11111111.11111111.11111000
   Number of bits for host IDs? 3
   Number of host IDs per subnet? $2^3 - 2 = 6$
2. Split into 333 subnets:
   Network ID: 00001111.00000000.00000000.00000000 = 15.0.0.0
   Number of bits for subnet IDs?
   Subnet mask:
   Number of bits for host IDs?
   Number of host IDs per subnet?

Exercise 3: Subnetting

1. Split into 30 subnets:
   Network ID: 11000011.00000001.00011111.00000000 = 195.1.31.0
   Number of bits for subnet IDs? 30 => 32 $=2^5$ => 5 bits
   Subnet mask: 11111111.11111111.11111111.11111000
   Number of bits for host IDs? 3
   Number of host IDs per subnet? $2^3 - 2 = 6$
2. Split into 333 subnets:
   Network ID: 00001111.00000000.00000000.00000000 = 15.0.0.0
   Number of bits for subnet IDs? 333 => 512 $=2^9$ => 9 bits
   Subnet mask:
   Number of bits for host IDs?
   Number of host IDs per subnet?

Exercise 3: Subnetting

1. Split into 30 subnets:
   Network ID: 11000011.00000001.00011111.00000000 = 195.1.31.0
   Number of bits for subnet IDs? 30 => 32 $=2^5$ => 5 bits
   Subnet mask: 11111111.11111111.11111111.11111000
   Number of bits for host IDs? 3
   Number of host IDs per subnet? $2^3 - 2 = 6$
2. Split into 333 subnets:
   Network ID: 00001111.00000000.00000000.00000000 = 15.0.0.0
   Number of bits for subnet IDs? 333 => 512 $=2^9$ => 9 bits
   Subnet mask: 11111111.11111111.10000000.00000000
   Number of bits for host IDs?
   Number of host IDs per subnet?

Exercise 3: Subnetting

1. Split into 30 subnets:
   Network ID: 11000011.00000001.00011111.00000000 = 195.1.31.0
   Number of bits for subnet IDs? 30 => 32 $=2^5$ => 5 bits
   Subnet mask: 11111111.11111111.11111111.11111000
   Number of bits for host IDs? 3
   Number of host IDs per subnet? $2^3 - 2 = 6$
2. Split into 333 subnets:
   Network ID: 00001111.00000000.00000000.00000000 = 15.0.0.0
   Number of bits for subnet IDs? 333 => 512 $=2^9$ => 9 bits
   Subnet mask: 11111111.11111111.10000000.00000000
   Number of bits for host IDs? 15
   Number of host IDs per subnet?

Exercise 3: Subnetting

1. Split into 30 subnets:
   Network ID: 11000011.00000001.00011111.00000000 = 195.1.31.0
   Number of bits for subnet IDs? 30 => 32 $=2^5$ => 5 bits
   Subnet mask: 11111111.11111111.11111111.11111000
   Number of bits for host IDs? 3
   Number of host IDs per subnet? $2^3 - 2 = 6$
2. Split into 333 subnets:
   Network ID: 00001111.00000000.00000000.00000000 = 15.0.0.0
   Number of bits for subnet IDs? 333 => 512 $=2^9$ => 9 bits
   Subnet mask: 11111111.11111111.10000000.00000000
   Number of bits for host IDs? 15
   Number of host IDs per subnet? $2^{15} - 2 = 32,766$
3. Split into 20 subnets:
   Network ID: 10111101.00010111.00000000.00000000 = 189.23.0.0
   Number of bits for subnet IDs?
   Subnet mask:
   Number of bits for host IDs?
   Number of host IDs per subnet?

Exercise 3: Subnetting

1. Split into 30 subnets:
   Network ID: 11000011.00000001.00011111.00000000 = 195.1.31.0
   Number of bits for subnet IDs? 30 => 32 $=2^5$ => 5 bits
   Subnet mask: 11111111.11111111.11111111.11111000
   Number of bits for host IDs? 3
   Number of host IDs per subnet? $2^3 - 2 = 6$
2. Split into 333 subnets:
   Network ID: 00001111.00000000.00000000.00000000 = 15.0.0.0
   Number of bits for subnet IDs? 333 => 512 $=2^9$ => 9 bits
   Subnet mask: 11111111.11111111.10000000.00000000
   Number of bits for host IDs? 15
   Number of host IDs per subnet? $2^{15} - 2 = 32,766$
3. Split into 20 subnets:
   Network ID: 10111101.00010111.00000000.00000000 = 189.23.0.0
   Number of bits for subnet IDs? 20 => 32 $=2^5$ => 5 bits
   Subnet mask:
   Number of bits for host IDs?
   Number of host IDs per subnet?

Exercise 3: Subnetting

1. Split into 30 subnets:
   Network ID: 11000011.00000001.00011111.00000000 = 195.1.31.0
   Number of bits for subnet IDs? 30 => 32 $=2^5$ => 5 bits
   Subnet mask: 11111111.11111111.11111111.11111000
   Number of bits for host IDs? 3
   Number of host IDs per subnet? $2^3 - 2 = 6$
2. Split into 333 subnets:
   Network ID: 00001111.00000000.00000000.00000000 = 15.0.0.0
   Number of bits for subnet IDs? 333 => 512 $=2^9$ => 9 bits
   Subnet mask: 11111111.11111111.10000000.00000000
   Number of bits for host IDs? 15
   Number of host IDs per subnet? $2^{15} - 2 = 32,766$
3. Split into 20 subnets:
   Network ID: 10111101.00010111.00000000.00000000 = 189.23.0.0
   Number of bits for subnet IDs? 20 => 32 $=2^5$ => 5 bits
   Subnet mask: 11111111.11111111.11111000.00000000
   Number of bits for host IDs?
   Number of host IDs per subnet?

Exercise 3: Subnetting

1. Split into 30 subnets:
   Network ID: 11000011.00000001.00011111.00000000 = 195.1.31.0
   Number of bits for subnet IDs? 30 => 32 $=2^5$ => 5 bits
   Subnet mask: 11111111.11111111.11111111.11111000
   Number of bits for host IDs? 3
   Number of host IDs per subnet? $2^3 - 2 = 6$
2. Split into 333 subnets:
   Network ID: 00001111.00000000.00000000.00000000 = 15.0.0.0
   Number of bits for subnet IDs? 333 => 512 $=2^9$ => 9 bits
   Subnet mask: 11111111.11111111.10000000.00000000
   Number of bits for host IDs? 15
   Number of host IDs per subnet? $2^{15} - 2 = 32,766$
3. Split into 20 subnets:
   Network ID: 10111101.00010111.00000000.00000000 = 189.23.0.0
   Number of bits for subnet IDs? 20 => 32 $=2^5$ => 5 bits
   Subnet mask: 11111111.11111111.11111000.00000000
   Number of bits for host IDs? 11
   Number of host IDs per subnet?

Exercise 3: Subnetting

1. Split into 30 subnets:
   Network ID: 11000011.00000001.00011111.00000000 = 195.1.31.0
   Number of bits for subnet IDs? 30 => 32 $=2^5$ => 5 bits
   Subnet mask: 11111111.11111111.11111111.11111000
   Number of bits for host IDs? 3
   Number of host IDs per subnet? $2^3 - 2 = 6$
2. Split into 333 subnets:
   Network ID: 00001111.00000000.00000000.00000000 = 15.0.0.0
   Number of bits for subnet IDs? 333 => 512 $=2^9$ => 9 bits
   Subnet mask: 11111111.11111111.10000000.00000000
   Number of bits for host IDs? 15
   Number of host IDs per subnet? $2^{15} - 2 = 32,766$
3. Split into 20 subnets:
   Network ID: 10111101.00010111.00000000.00000000 = 189.23.0.0
   Number of bits for subnet IDs? 20 => 32 $=2^5$ => 5 bits
   Subnet mask: 11111111.11111111.11111000.00000000
   Number of bits for host IDs? 11
   Number of host IDs per subnet? $2^{11} - 2 = 2,046$

Exercise 3: Subnetting

4. Each subnet should have 17 hosts:
   `Network ID: 11000011.00000011.10000000.00000000 = 195.3.128.0`
   `Number of bits for host IDs?`
   `Number of bits for subnet IDs?`
   `Number of possible subnets?`
   `Subnet mask:`

Exercise 3: Subnetting

4. Each subnet should have 17 hosts:

   Network ID: 11000011.00000011.10000000.00000000 = 195.3.128.0

   Number of bits for host IDs? 17 + 2 => 32 = 2^5 => 5 bits

   Number of bits for subnet IDs?

   Number of possible subnets?

   Subnet mask:

Exercise 3: Subnetting

4. Each subnet should have 17 hosts:

   Network ID: 11000011.00000011.10000000.00000000 = 195.3.128.0

   Number of bits for host IDs? 17 + 2 => 32 = 2^5 => 5 bits

   Number of bits for subnet IDs? 3

   Number of possible subnets?

   Subnet mask:

Exercise 3: Subnetting

4. Each subnet should have 17 hosts:

   `Network ID: 11000011.00000011.10000000.00000000 = 195.3.128.0`

   `Number of bits for host IDs?` `17 + 2 => 32 =` 2^5 `=> 5 bits`

   `Number of bits for subnet IDs?` `3`

   `Number of possible subnets?` $2^3 = 8$

   `Subnet mask:`

Exercise 3: Subnetting

4. Each subnet should have 17 hosts:
   Network ID: 11000011.00000011.10000000.00000000 = 195.3.128.0
   Number of bits for host IDs? 17 + 2 => 32 = 2^5 => 5 bits
   Number of bits for subnet IDs? 3
   Number of possible subnets? $2^3 = 8$
   Subnet mask: 11111111.11111111.11111111.11100000 = 255.255.255.224
5. Each subnet should have 10 hosts:
   Network ID: 10000001.00001111.00000000.00000000 = 129.15.0.0
   Number of bits for host IDs?
   Number of bits for subnet IDs?
   Number of possible subnets?
   Subnet mask:

Exercise 3: Subnetting

4. Each subnet should have 17 hosts:
   Network ID: 11000011.00000011.10000000.00000000 = 195.3.128.0
   Number of bits for host IDs? 17 + 2 => 32 = 2^5 => 5 bits
   Number of bits for subnet IDs? 3
   Number of possible subnets? $2^3 = 8$
   Subnet mask: 11111111.11111111.11111111.11100000 = 255.255.255.224
5. Each subnet should have 10 hosts:
   Network ID: 10000001.00001111.00000000.00000000 = 129.15.0.0
   Number of bits for host IDs? 10 + 2=> 16 = 2^4 => 4 bits
   Number of bits for subnet IDs?
   Number of possible subnets?
   Subnet mask:

Exercise 3: Subnetting

4. Each subnet should have 17 hosts:
   Network ID: 11000011.00000011.10000000.00000000 = 195.3.128.0
   Number of bits for host IDs? 17 + 2 => 32 = 2^5 => 5 bits
   Number of bits for subnet IDs? 3
   Number of possible subnets? $2^3 = 8$
   Subnet mask: 11111111.11111111.11111111.11100000 = 255.255.255.224
5. Each subnet should have 10 hosts:
   Network ID: 10000001.00001111.00000000.00000000 = 129.15.0.0
   Number of bits for host IDs? 10 + 2=> 16 = 2^4 => 4 bits
   Number of bits for subnet IDs? 12
   Number of possible subnets?
   Subnet mask:

Exercise 3: Subnetting

4. Each subnet should have 17 hosts:
   Network ID: 11000011.00000011.10000000.00000000 = 195.3.128.0
   Number of bits for host IDs? 17 + 2 => 32 = 2^5 => 5 bits
   Number of bits for subnet IDs? 3
   Number of possible subnets? $2^3 = 8$
   Subnet mask: 11111111.11111111.11111111.11100000 = 255.255.255.224
5. Each subnet should have 10 hosts:
   Network ID: 10000001.00001111.00000000.00000000 = 129.15.0.0
   Number of bits for host IDs? 10 + 2=> 16 = 2^4 => 4 bits
   Number of bits for subnet IDs? 12
   Number of possible subnets? $2^{12} = 4096$
   Subnet mask:

Exercise 3: Subnetting

4. Each subnet should have 17 hosts:
   Network ID: 11000011.00000011.10000000.00000000 = 195.3.128.0
   Number of bits for host IDs? 17 + 2 => 32 = 2^5 => 5 bits
   Number of bits for subnet IDs? 3
   Number of possible subnets? $2^3 = 8$
   Subnet mask: 11111111.11111111.11111111.11100000 = 255.255.255.224
5. Each subnet should have 10 hosts:
   Network ID: 10000001.00001111.00000000.00000000 = 129.15.0.0
   Number of bits for host IDs? 10 + 2=> 16 = 2^4 => 4 bits
   Number of bits for subnet IDs? 12
   Number of possible subnets? $2^{12} = 4096$
   Subnet mask: 11111111.11111111.11111111.11110000 = 255.255.255.240

Exercise 4: Checksums in IP Packets

Calculate the checksum for each IP header:

- `4500 0034 4C22 4000 F706 ???? C163 9055 0A00 008B`
- `4500 0034 671E 4000 4006 ???? 0A00 008b C163 9055`
- `4500 00F2 0000 4000 4011 ???? 0A00 008b 0A00 00FF`

Verify the checksum of each IP header:

- `4500 0034 02FD 4000 3606 276C 6CA0 A330 0A00 008B`
- `4500 00E7 02FC 4000 3606 37BC 6CA0 A330 0A00 008B`
- `4500 0034 A9D5 4000 4006 814E 0A00 008B adC2 4613`

Exercise 4: Checksums in IP Packets

Calculate the checksum for each IP header:

- `4500 0034 4C22 4000 F706 DB5D C163 9055 0A00 008B`
- `4500 0034 671E 4000 4006 ???? 0A00 008b C163 9055`
- `4500 00F2 0000 4000 4011 ???? 0A00 008b 0A00 00FF`

Verify the checksum of each IP header:

- `4500 0034 02FD 4000 3606 276C 6CA0 A330 0A00 008B`
- `4500 00E7 02FC 4000 3606 37BC 6CA0 A330 0A00 008B`
- `4500 0034 A9D5 4000 4006 814E 0A00 008B adC2 4613`

Exercise 4: Checksums in IP Packets

Calculate the checksum for each IP header:

- `4500 0034 4C22 4000 F706 DB5D C163 9055 0A00 008B`
- `4500 0034 671E 4000 4006 7762 0A00 008b C163 9055`
- `4500 00F2 0000 4000 4011 ???? 0A00 008b 0A00 00FF`

Verify the checksum of each IP header:

- `4500 0034 02FD 4000 3606 276C 6CA0 A330 0A00 008B`
- `4500 00E7 02FC 4000 3606 37BC 6CA0 A330 0A00 008B`
- `4500 0034 A9D5 4000 4006 814E 0A00 008B adC2 4613`

Exercise 4: Checksums in IP Packets

Calculate the checksum for each IP header:

- `4500 0034 4C22 4000 F706 DB5D C163 9055 0A00 008B`
- `4500 0034 671E 4000 4006 7762 0A00 008b C163 9055`
- `4500 00F2 0000 4000 4011 2472 0A00 008b 0A00 00FF`

Verify the checksum of each IP header:

- `4500 0034 02FD 4000 3606 276C 6CA0 A330 0A00 008B`
- `4500 00E7 02FC 4000 3606 37BC 6CA0 A330 0A00 008B`
- `4500 0034 A9D5 4000 4006 814E 0A00 008B adC2 4613`

Exercise 4: Checksums in IP Packets

Calculate the checksum for each IP header:

- `4500 0034 4C22 4000 F706 DB5D C163 9055 0A00 008B`
- `4500 0034 671E 4000 4006 7762 0A00 008b C163 9055`
- `4500 00F2 0000 4000 4011 2472 0A00 008b 0A00 00FF`

Verify the checksum of each IP header:

- `4500 0034 02FD 4000 3606 276C 6CA0 A330 0A00 008B`

→ Correct

- `4500 00E7 02FC 4000 3606 37BC 6CA0 A330 0A00 008B`

- `4500 0034 A9D5 4000 4006 814E 0A00 008B adC2 4613`

Exercise 4: Checksums in IP Packets

Calculate the checksum for each IP header:

- `4500 0034 4C22 4000 F706 DB5D C163 9055 0A00 008B`
- `4500 0034 671E 4000 4006 7762 0A00 008b C163 9055`
- `4500 00F2 0000 4000 4011 2472 0A00 008b 0A00 00FF`

Verify the checksum of each IP header:

- `4500 0034 02FD 4000 3606 276C 6CA0 A330 0A00 008B`

→ Correct

- `4500 00E7 02FC 4000 3606 37BC 6CA0 A330 0A00 008B`

→ Wrong! Correct is: 26BA

- `4500 0034 A9D5 4000 4006 814E 0A00 008B adC2 4613`

Exercise 4: Checksums in IP Packets

Calculate the checksum for each IP header:

- `4500 0034 4C22 4000 F706 DB5D C163 9055 0A00 008B`
- `4500 0034 671E 4000 4006 7762 0A00 008b C163 9055`
- `4500 00F2 0000 4000 4011 2472 0A00 008b 0A00 00FF`

Verify the checksum of each IP header:

- `4500 0034 02FD 4000 3606 276C 6CA0 A330 0A00 008B`

→ Correct

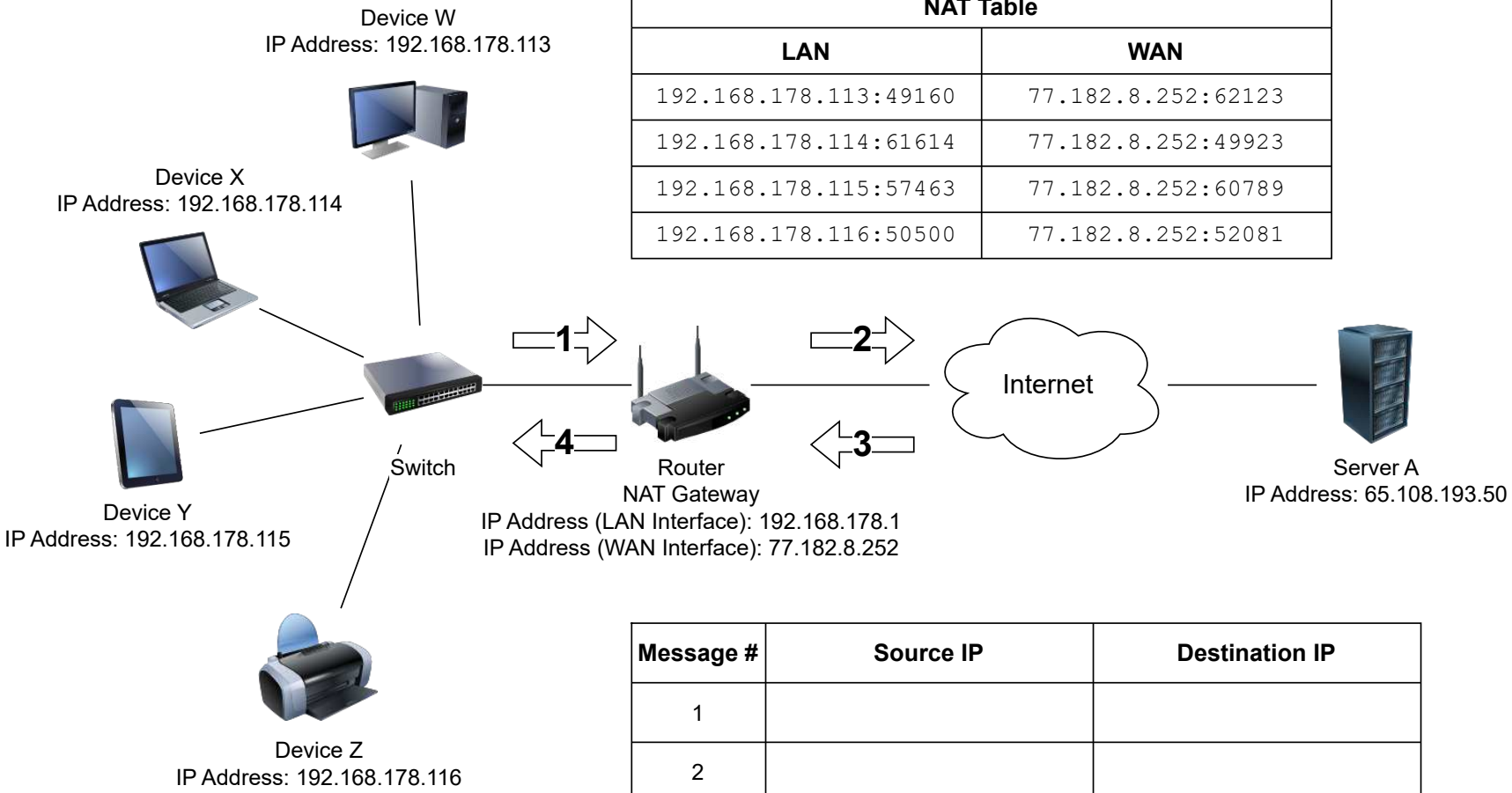
- `4500 00E7 02FC 4000 3606 37BC 6CA0 A330 0A00 008B`

→ Wrong! Correct is: 26BA

- `4500 0034 A9D5 4000 4006 814E 0A00 008B adC2 4613`

→ Wrong! Correct is: 928E

Exercise 5: Network Address Translation



NAT Table	
LAN	**WAN**
192.168.178.113:49160	77.182.8.252:62123
192.168.178.114:61614	77.182.8.252:49923
192.168.178.115:57463	77.182.8.252:60789
192.168.178.116:50500	77.182.8.252:52081

Message #	Source IP	Destination IP
1		
2		
3		
4		

Exercise 6: Address Types and Spaces

1. Name the three private IPv4 address spaces.
2. What is the prefix for a link-local address in IPv4 and IPv6 networks?
3. Which of the following IPv4 addresses are multicast addresses?
4. How can an IPv6 anycast address be distinguished from a unicast or a multicast address?
5. Which IPv6 address can you use in order to *ping* all stations in a local network?
6. What type of address is given with `fd04:2342:0815:1:6770:37ca:7a5c:f408/64`? What is its purpose?
7. What type of address is given with `ff02::1:ff5c:f408`? What is its purpose?

Exercise 6: Address Types and Spaces

1. Name the three private IPv4 address spaces.

→ `10.0.0.0/8, 172.16.0.0/12, and 192.168.0.0/16`

2. What is the prefix for a link-local address in IPv4 and IPv6 networks?

3. Which of the following IPv4 addresses are multicast addresses?

4. How can an IPv6 anycast address be distinguished from a unicast or a multicast address?

5. Which IPv6 address can you use in order to *ping* all stations in a local network?

6. What type of address is given with `fd04:2342:0815:1:6770:37ca:7a5c:f408/64`? What is its purpose?

7. What type of address is given with `ff02::1:ff5c:f408`? What is its purpose?

Exercise 6: Address Types and Spaces

1. Name the three private IPv4 address spaces.

→ `10.0.0.0/8, 172.16.0.0/12, and 192.168.0.0/16`

2. What is the prefix for a link-local address in IPv4 and IPv6 networks?

→ `169.254.0.0/16 and fe80::/10`

3. Which of the following IPv4 addresses are multicast addresses?

4. How can an IPv6 anycast address be distinguished from a unicast or a multicast address?

5. Which IPv6 address can you use in order to *ping* all stations in a local network?

6. What type of address is given with `fd04:2342:0815:1:6770:37ca:7a5c:f408/64`? What is its purpose?

7. What type of address is given with `ff02::1:ff5c:f408`? What is its purpose?

Exercise 6: Address Types and Spaces

1. Name the three private IPv4 address spaces.

→ `10.0.0.0/8, 172.16.0.0/12, and 192.168.0.0/16`

2. What is the prefix for a link-local address in IPv4 and IPv6 networks?

→ `169.254.0.0/16 and fe80::/10`

3. Which of the following IPv4 addresses are multicast addresses?
   - 224.1.2.3
   - 234.23.23.23

4. How can an IPv6 anycast address be distinguished from a unicast or a multicast address?

5. Which IPv6 address can you use in order to *ping* all stations in a local network?

6. What type of address is given with `fd04:2342:0815:1:6770:37ca:7a5c:f408/64`? What is its purpose?

7. What type of address is given with `ff02::1:ff5c:f408`? What is its purpose?

Exercise 6: Address Types and Spaces

1. Name the three private IPv4 address spaces.

→ `10.0.0.0/8, 172.16.0.0/12, and 192.168.0.0/16`

2. What is the prefix for a link-local address in IPv4 and IPv6 networks?

→ `169.254.0.0/16 and fe80::/10`

3. Which of the following IPv4 addresses are multicast addresses?
   - 224.1.2.3
   - 234.23.23.23

4. How can an IPv6 anycast address be distinguished from a unicast or a multicast address?

→ As soon as a unicast address is assigned to more than one interface it becomes an anycast address.

5. Which IPv6 address can you use in order to *ping* all stations in a local network?

6. What type of address is given with `fd04:2342:0815:1:6770:37ca:7a5c:f408/64`? What is its purpose?

7. What type of address is given with `ff02::1:ff5c:f408`? What is its purpose?

Exercise 6: Address Types and Spaces

1. Name the three private IPv4 address spaces.

→ `10.0.0.0/8, 172.16.0.0/12, and 192.168.0.0/16`

2. What is the prefix for a link-local address in IPv4 and IPv6 networks?

→ `169.254.0.0/16 and fe80::/10`

3. Which of the following IPv4 addresses are multicast addresses?
   - 224.1.2.3
   - 234.23.23.23

4. How can an IPv6 anycast address be distinguished from a unicast or a multicast address?

→ As soon as a unicast address is assigned to more than one interface it becomes an anycast address.

5. Which IPv6 address can you use in order to *ping* all stations in a local network?

→ Using the *all nodes* multicast addresses: (`ff02::1` and `ff05::1`).

6. What type of address is given with `fd04:2342:0815:1:6770:37ca:7a5c:f408/64`? What is its purpose?

7. What type of address is given with `ff02::1:ff5c:f408`? What is its purpose?

Exercise 6: Address Types and Spaces

1. Name the three private IPv4 address spaces.

→ `10.0.0.0/8, 172.16.0.0/12, and 192.168.0.0/16`

2. What is the prefix for a link-local address in IPv4 and IPv6 networks?

→ `169.254.0.0/16 and fe80::/10`

3. Which of the following IPv4 addresses are multicast addresses?
   - 224.1.2.3
   - 234.23.23.23

4. How can an IPv6 anycast address be distinguished from a unicast or a multicast address?

→ As soon as a unicast address is assigned to more than one interface it becomes an anycast address.

5. Which IPv6 address can you use in order to *ping* all stations in a local network?

→ Using the *all nodes* multicast addresses: (`ff02::1` and `ff05::1`).

6. What type of address is given with `fd04:2342:0815:1:6770:37ca:7a5c:f408/64`? What is its purpose?

→ This is a unique local address (ULA) which serves a similar purpose as private address in IPv4.

7. What type of address is given with `ff02::1:ff5c:f408`? What is its purpose?

Exercise 6: Address Types and Spaces

1. Name the three private IPv4 address spaces.

→ `10.0.0.0/8, 172.16.0.0/12, and 192.168.0.0/16`

2. What is the prefix for a link-local address in IPv4 and IPv6 networks?

→ `169.254.0.0/16 and fe80::/10`

3. Which of the following IPv4 addresses are multicast addresses?
   - 224.1.2.3
   - 234.23.23.23

4. How can an IPv6 anycast address be distinguished from a unicast or a multicast address?

→ As soon as a unicast address is assigned to more than one interface it becomes an anycast address.

5. Which IPv6 address can you use in order to *ping* all stations in a local network?

→ Using the *all nodes* multicast addresses: (`ff02::1` and `ff05::1`).

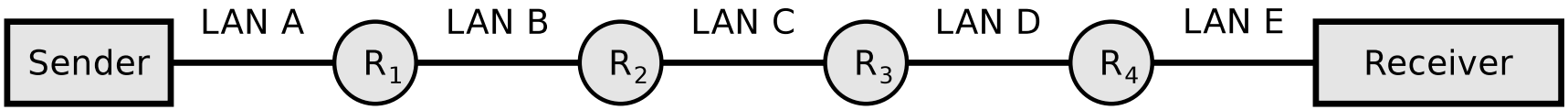
6. What type of address is given with `fd04:2342:0815:1:6770:37ca:7a5c:f408/64`? What is its purpose?

→ This is a unique local address (ULA) which serves a similar purpose as private address in IPv4.

7. What type of address is given with `ff02::1:ff5c:f408`? What is its purpose?

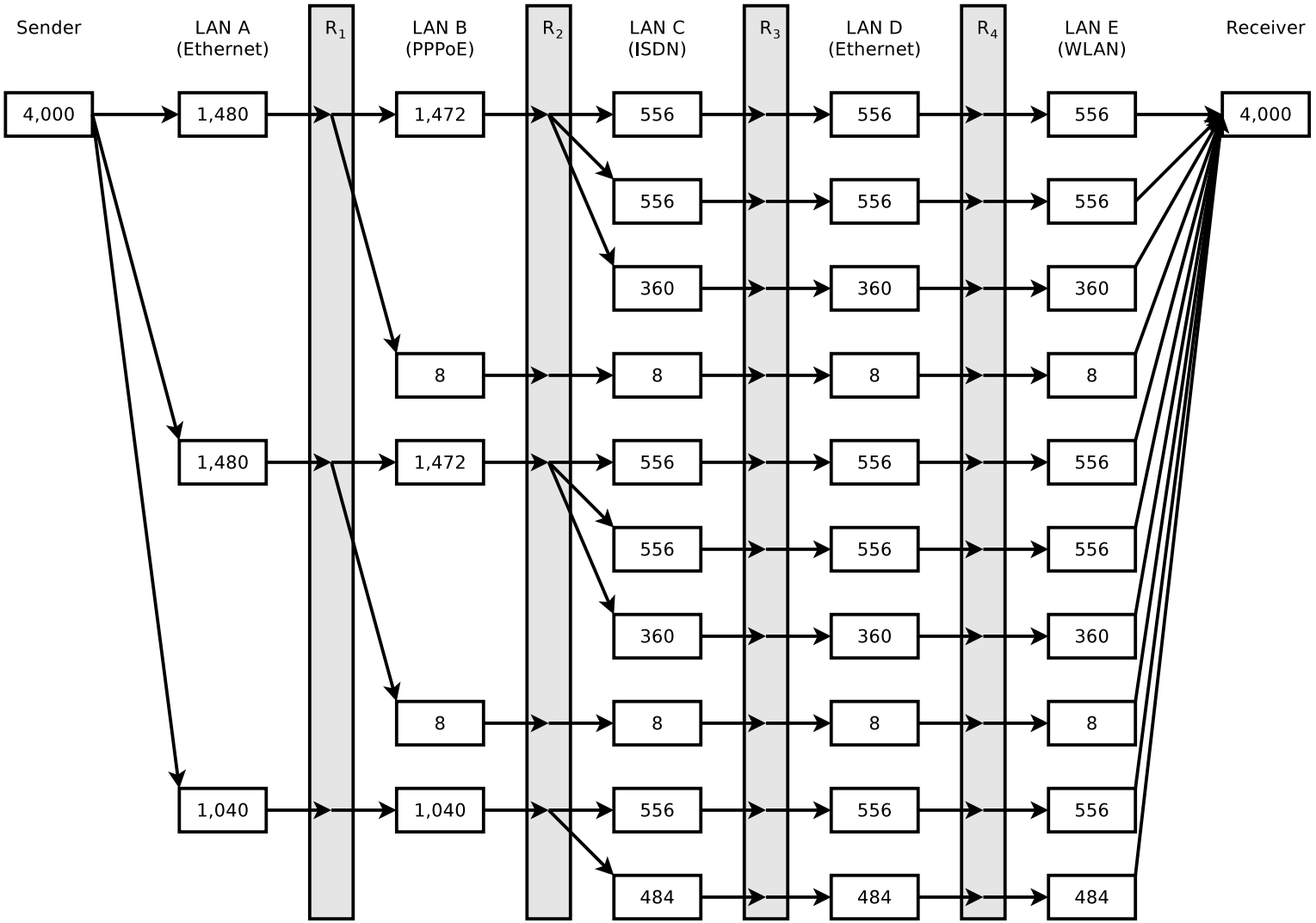
→ This is a solicited node multicast address which is used for NDP.

Exercise 7: Fragmenting IP Packets



	LAN A	**LAN B**	**LAN C**	**LAN D**	**LAN E**
Network technology	Ethernet	PPPoE	ISDN	Ethernet	WLAN
MTU [bytes]	1,500	1,492	576	1,400	2,312
IP-Header [bytes]	20	20	20	20	20
maximum payload [bytes]	1,480	1,472	556	1,380	2,292

Exercise 7: Fragmenting IP Packets



Exercise 8.1: IPv6 Address Representation

Simplify these IPv6 addresses:

- `1080:0000:0000:0000:0007:0700:0003:316b`
- `2001:0db8:0000:0000:f065:00ff:0000:03ec`
- `2001:0db8:3c4d:0016:0000:0000:2a3f:2a4d`
- `2001:0c60:f0a1:0000:0000:0000:0000:0001`
- `2111:00ab:0000:0004:0000:0000:0000:1234`

Exercise 8.1: IPv6 Address Representation

Simplify these IPv6 addresses:

- `1080:0000:0000:0000:0007:0700:0003:316b`

  Solution: `1080::7:700:3:316b`
- `2001:0db8:0000:0000:f065:00ff:0000:03ec`
- `2001:0db8:3c4d:0016:0000:0000:2a3f:2a4d`
- `2001:0c60:f0a1:0000:0000:0000:0000:0001`
- `2111:00ab:0000:0004:0000:0000:0000:1234`

Exercise 8.1: IPv6 Address Representation

Simplify these IPv6 addresses:

- `1080:0000:0000:0000:0007:0700:0003:316b`

  Solution: `1080::7:700:3:316b`
- `2001:0db8:0000:0000:f065:00ff:0000:03ec`

  Solution: `2001:db8::f065:ff:0:3ec`
- `2001:0db8:3c4d:0016:0000:0000:2a3f:2a4d`
- `2001:0c60:f0a1:0000:0000:0000:0000:0001`
- `2111:00ab:0000:0004:0000:0000:0000:1234`

Exercise 8.1: IPv6 Address Representation

Simplify these IPv6 addresses:

- `1080:0000:0000:0000:0007:0700:0003:316b`

  Solution: `1080::7:700:3:316b`
- `2001:0db8:0000:0000:f065:00ff:0000:03ec`

  Solution: `2001:db8::f065:ff:0:3ec`
- `2001:0db8:3c4d:0016:0000:0000:2a3f:2a4d`

  Solution: `2001:db8:3c4d:16::2a3f:2a4d`
- `2001:0c60:f0a1:0000:0000:0000:0000:0001`

- `2111:00ab:0000:0004:0000:0000:0000:1234`

Exercise 8.1: IPv6 Address Representation

Simplify these IPv6 addresses:

- `1080:0000:0000:0000:0007:0700:0003:316b`

  Solution: `1080::7:700:3:316b`
- `2001:0db8:0000:0000:f065:00ff:0000:03ec`

  Solution: `2001:db8::f065:ff:0:3ec`
- `2001:0db8:3c4d:0016:0000:0000:2a3f:2a4d`

  Solution: `2001:db8:3c4d:16::2a3f:2a4d`
- `2001:0c60:f0a1:0000:0000:0000:0000:0001`

  Solution: `2001:c60:f0a1::1`
- `2111:00ab:0000:0004:0000:0000:0000:1234`

Exercise 8.1: IPv6 Address Representation

Simplify these IPv6 addresses:

- `1080:0000:0000:0000:0007:0700:0003:316b`

  Solution: `1080::7:700:3:316b`
- `2001:0db8:0000:0000:f065:00ff:0000:03ec`

  Solution: `2001:db8::f065:ff:0:3ec`
- `2001:0db8:3c4d:0016:0000:0000:2a3f:2a4d`

  Solution: `2001:db8:3c4d:16::2a3f:2a4d`
- `2001:0c60:f0a1:0000:0000:0000:0000:0001`

  Solution: `2001:c60:f0a1::1`
- `2111:00ab:0000:0004:0000:0000:0000:1234`

  Solution: `2111:ab:0:4::1234`

Exercise 8.2: IPv6 Address Representation

Provide all positions of these simplified IPv6 addresses:

- `2001::2:0:0:1`
- `2001:db8:0:c::1c`
- `1080::9956:0:0:234`
- `2001:638:208:ef34::91ff:0:5424`
- `2001:0:85a4::4a1e:370:7112`

Exercise 8.2: IPv6 Address Representation

Provide all positions of these simplified IPv6 addresses:

- `2001::2:0:0:1`

  Solution: `2001:0000:0000:0000:0002:0000:0000:0001`
- `2001:db8:0:c::1c`
- `1080::9956:0:0:234`
- `2001:638:208:ef34::91ff:0:5424`
- `2001:0:85a4::4a1e:370:7112`

Exercise 8.2: IPv6 Address Representation

Provide all positions of these simplified IPv6 addresses:

- `2001::2:0:0:1`

  Solution: `2001:0000:0000:0000:0002:0000:0000:0001`
- `2001:db8:0:c::1c`

  Solution: `2001:0db8:0000:000c:0000:0000:0000:001c`
- `1080::9956:0:0:234`
- `2001:638:208:ef34::91ff:0:5424`
- `2001:0:85a4::4a1e:370:7112`

Exercise 8.2: IPv6 Address Representation

Provide all positions of these simplified IPv6 addresses:

- `2001::2:0:0:1`

  Solution: `2001:0000:0000:0000:0002:0000:0000:0001`

- `2001:db8:0:c::1c`

  Solution: `2001:0db8:0000:000c:0000:0000:0000:001c`

- `1080::9956:0:0:234`

  Solution: `1080:0000:0000:0000:9956:0000:0000:0234`

- `2001:638:208:ef34::91ff:0:5424`

- `2001:0:85a4::4a1e:370:7112`

Exercise 8.2: IPv6 Address Representation

Provide all positions of these simplified IPv6 addresses:

- `2001::2:0:0:1`

  Solution: `2001:0000:0000:0000:0002:0000:0000:0001`
- `2001:db8:0:c::1c`

  Solution: `2001:0db8:0000:000c:0000:0000:0000:001c`
- `1080::9956:0:0:234`

  Solution: `1080:0000:0000:0000:9956:0000:0000:0234`
- `2001:638:208:ef34::91ff:0:5424`

  Solution: `2001:0638:0208:ef34:0000:91ff:0000:5424`
- `2001:0:85a4::4a1e:370:7112`

Exercise 8.2: IPv6 Address Representation

Provide all positions of these simplified IPv6 addresses:

- `2001::2:0:0:1`

  Solution: `2001:0000:0000:0000:0002:0000:0000:0001`
- `2001:db8:0:c::1c`

  Solution: `2001:0db8:0000:000c:0000:0000:0000:001c`
- `1080::9956:0:0:234`

  Solution: `1080:0000:0000:0000:9956:0000:0000:0234`
- `2001:638:208:ef34::91ff:0:5424`

  Solution: `2001:0638:0208:ef34:0000:91ff:0000:5424`
- `2001:0:85a4::4a1e:370:7112`

  Solution: `2001:0000:85a4:0000:0000:4a1e:0370:7112`

Exercise 9.1: Do some research

1. The transition from IPv4 to IPv6 may indicate that one IP version number has been skipped. What happened to **IPv5**?

Exercise 9.1: Do some research

1. The transition from IPv4 to IPv6 may indicate that one IP version number has been skipped. What happened to **IPv5**?

   The protocol to be transported on the network layer using an IP header with the version set to 5 is the *Internet Stream Protocol*. It defines a family of experimental protocols which were never introduced for public use. It is specified in RFCs 1190 and 1819 and some concepts were adopted for ATM or MPLS.

Exercise 9.2: Do some research

2. Explain the meaning of the fields `Flags`, `MSS`, `Window`, and `irtt` in the forwarding table as shown in task 2.

Exercise 9.2: Do some research

2. Explain the meaning of the fields `Flags`, `MSS`, `Window`, and `irtt` in the forwarding table as shown in task 2.

Flags :

- U route is up
- H target is a host
- G use gateway
- R reinstate route for dynamic routing
- D dynamically installed by daemon or redirect
- M modified from routing daemon or redirect
- A installed by addrconf
- C cache entry
- ! reject route

MSS Default maximum segment size for TCP connections over this route.

Window Default window size for TCP connections over this route.

irtt Initial RTT (Round Trip Time). The kernel uses this to guess about the best TCP protocol parameters without waiting on (possibly slow) answers.

Exercise 9.3: Do some research

3. In IPv6 different scopes are defined. Figure out which of the originally defined scopes has been declared as deprecated (and why).

Exercise 9.3: Do some research

3. In IPv6 different scopes are defined. Figure out which of the originally defined scopes has been declared as deprecated (and why).

   The site-local scope has been deprecated in RFC 3879 for various reasons. One of the reasons has been that the term was not well-defined.

Exercise 9.3: Do some research

3. In IPv6 different scopes are defined. Figure out which of the originally defined scopes has been declared as deprecated (and why).

   The site-local scope has been deprecated in RFC 3879 for various reasons. One of the reasons has been that the term was not well-defined.

4. Describe the purpose of the following address blocks:
   - 192.0.2.0/24
   - 198.51.100.0/24
   - 203.0.113.0/24

Exercise 9.3: Do some research

3. In IPv6 different scopes are defined. Figure out which of the originally defined scopes has been declared as deprecated (and why).

   The site-local scope has been deprecated in RFC 3879 for various reasons. One of the reasons has been that the term was not well-defined.

4. Describe the purpose of the following address blocks:
   - 192.0.2.0/24
   - 198.51.100.0/24
   - 203.0.113.0/24

   Documentation purposes, e.g., example IP addresses or ranges.