

# The TCP/IP Network Layer IPv4 and IPv6

2024

EPFL

# Recap

*communication*

*end-host connectivity*

*across-LANs interconnection*

*within-LAN interconnection*

*point-to-point transmission of bits*

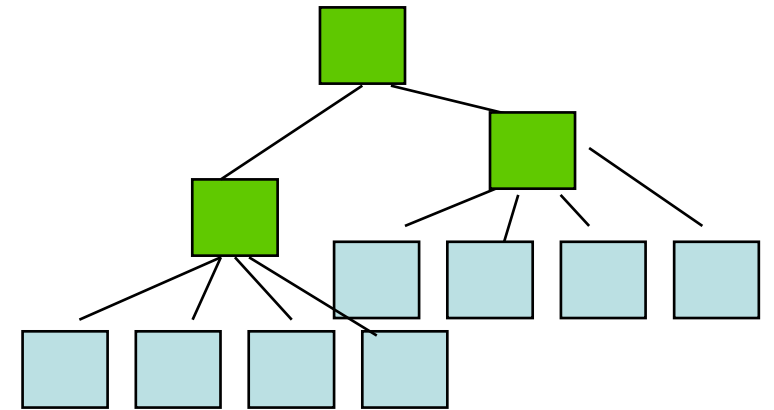
Application

Transport

Network

MAC

Physical

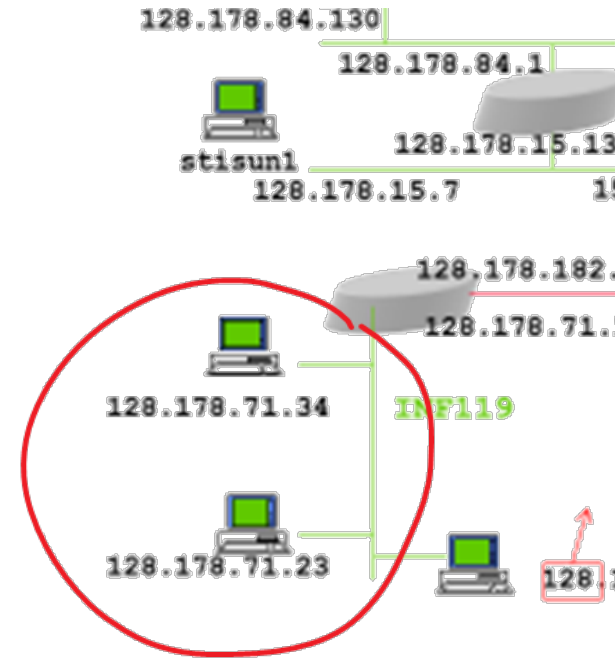


# Recap: most important protocol = IP (Internet Protocol)

- interconnects multiple *local area networks* (LANs)
- uses packet switching
- delivers packets from a source to a destination via a series of routers
- forwards packets from router to router based on *IP addresses*
- offers *no reliable*-delivery guarantees (*best-effort* approach)
  - packets are briefly stored in routers' buffers
  - packets of the same source-destination flow may follow different routes/paths
  - so, packets may be *dropped*, *delayed* or *reordered*

# Recall IP: subnets and subnet masks

- *subnet*  $\leftarrow$  a LAN, i.e. a set of devices:
  - connected at the Data-link layer
  - sharing the same IP-address *prefix*, e.g.: 128.178.71.X
- The prefix is specified using a *subnet mask*  
(= sequence of bits, where 1s indicate fixed positions of the prefix)
  - e.g. for an EPFL IPv4 LAN, the subnet mask is  
1111 1111 1111 1111 1111 1111 0000 0000
  - The size (in bits) of the prefix is not always the same, e.g.:  
ETHZ IPv4 LANs = 26 bits  
EPFL IPv6 LANs = 64 bits
- Various notations for the subnet mask:
  - **dotted, decimal**: e.g., address = 128.178.71.34, mask = 255.255.255.0
  - **"/" (slash)**: e.g. 128.178.71.34/24  
or 2001:620:618:1a6:0a00:20ff:fe78:30f9/64

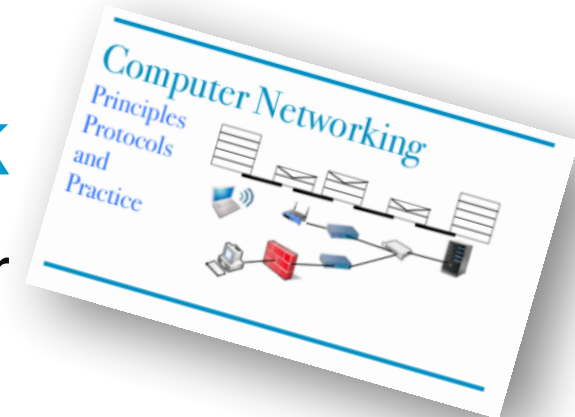


# Contents - Internet Protocol (IP)

1. The 2 main rules of IP Unicast
2. IPv4 addresses
3. IPv6 addresses
4. NATs
5. Host configuration
6. Hop Limit and TTL
7. ARP (connection with MAC layer)

## Textbook

### Chapter 5: The Network Layer



# IP Rule #1 = forward packets according to destination IP prefixes

Recall: goal of IP = interconnect all systems in the world using *IP addresses*

→ every network interface *must* have an IP address

Rule #1 in detail:

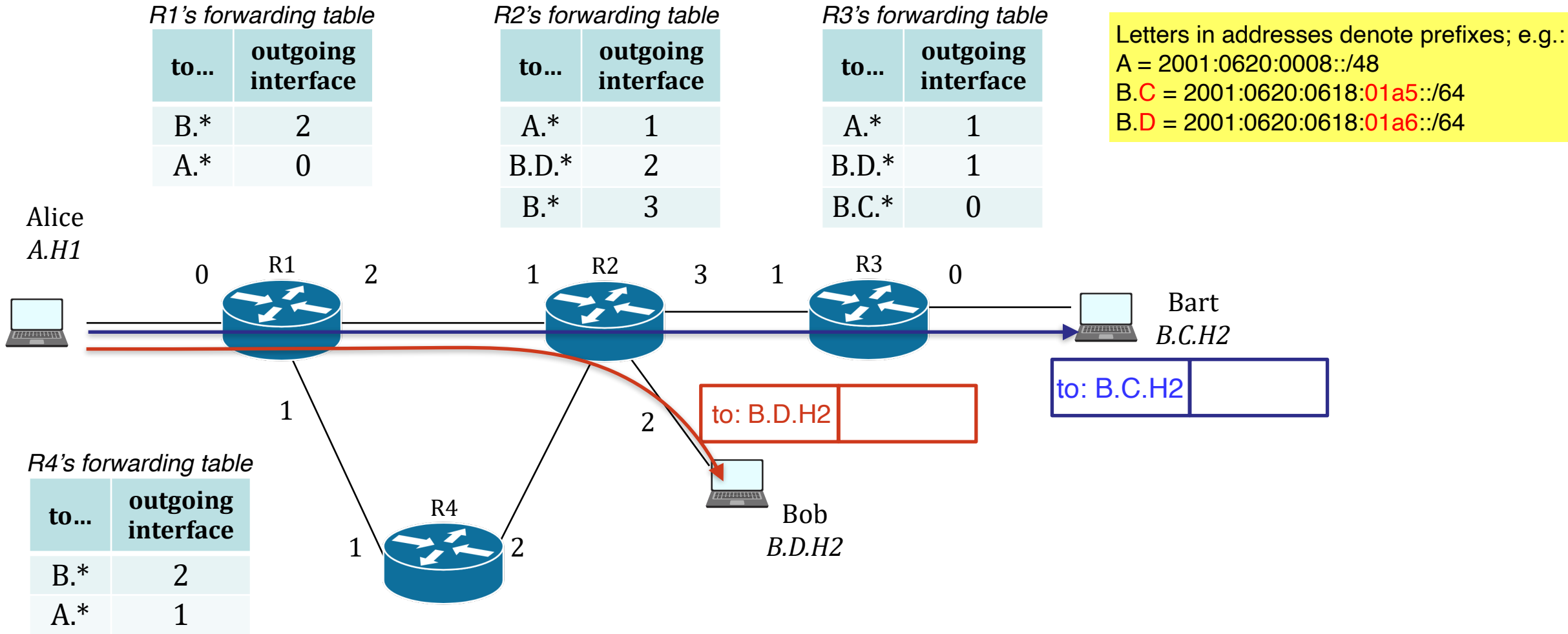
1. assign addresses based on a structure:

- every network interface has an IP address with *prefix* + *suffix*: e.g. 128.178.71.202
- interfaces inside *same subnet* have *same prefix* → *same subnet mask*

2. forward packets according to *longest prefix match*:

- every packet contains the destination IP address in its header
- every system (i.e. **host** = end-system or **router** = intermediate system)
  - has a **forwarding table** (= routing table) and
  - forwards each packet based on the *closest* table entry that matches the destination IP address

# Longest prefix match (= *closest* matching table entry)

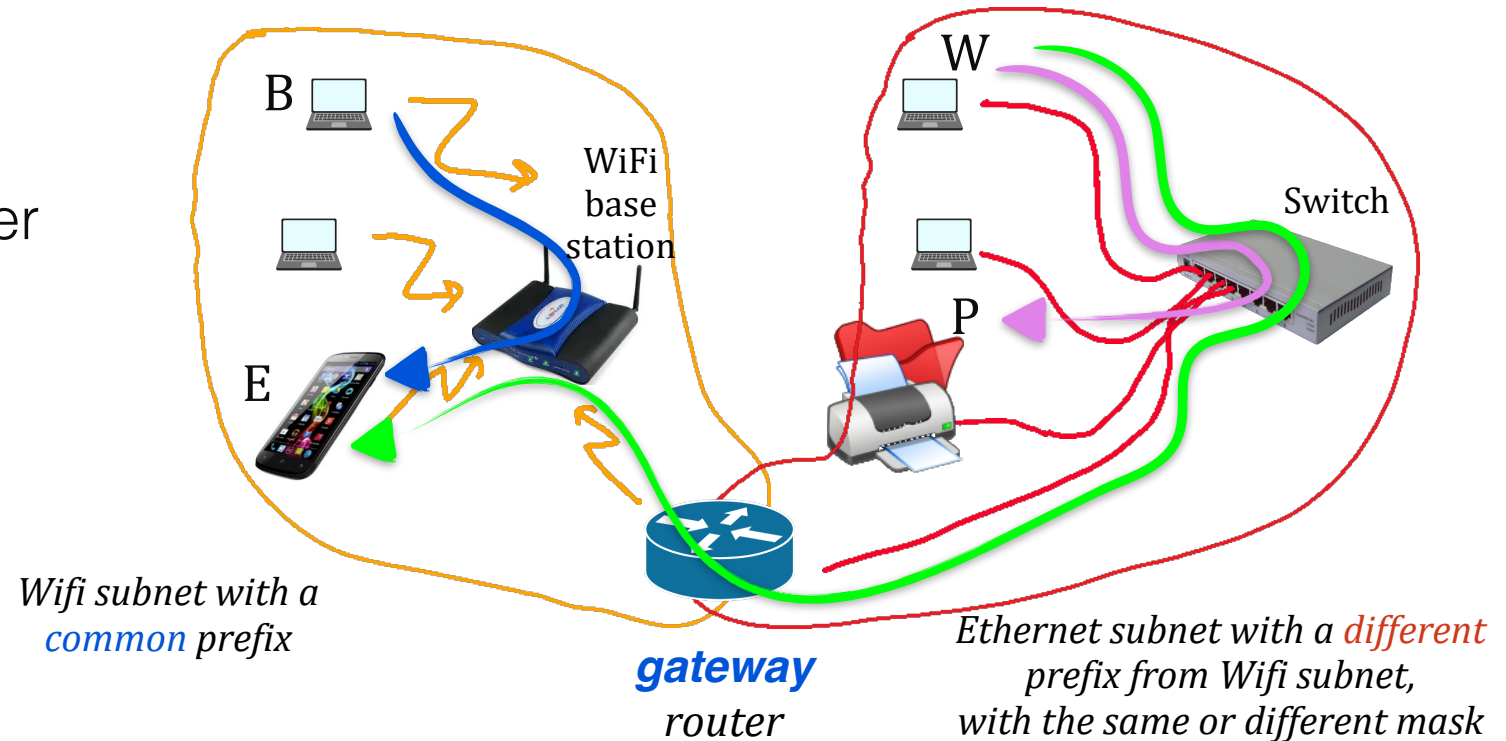


- Benefit: addresses can be *aggregated*, tables can be *compressed*



# IP Rule #2 = routers can only interconnect different LANs/subnets

- Traffic  $B \leftrightarrow E$  and  $W \leftrightarrow P$  does not go through router
- Traffic  $W \leftrightarrow E$  goes through router



IP rule #2 implies:

- *between* LANs/subnets, we use *routers*
- *inside* each subnet, we do *not* —> we use data-link-layer forwarding devices (such as switches, wifi base stations, etc.)

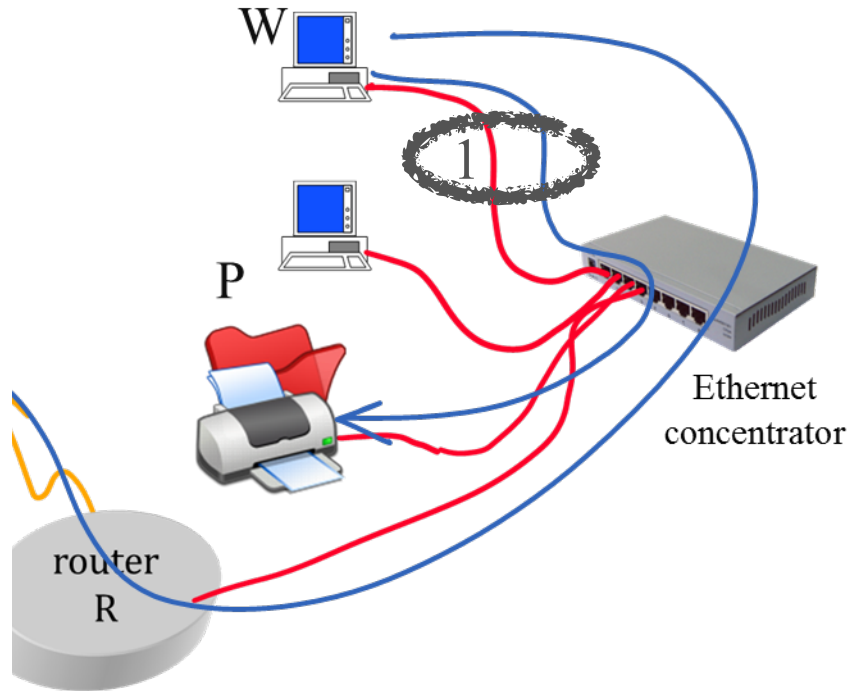


We observe a packet from W to P at 1.  
Which IP destination address do we see ?



Go to [web.speakup.info](http://web.speakup.info) or  
download speakup app

Join room  
46045



- A. The IP address of P
- B. The IP address of an Ethernet interface of the Ethernet concentrator
- C. There is no destination IP address in the packet since communication is inside the subnet and does not go through a router

# Solution

Answer A

The IP address is always present in the IP header if we use TCP/IP, even if communication is inside the same LAN.

## 2. Global/public unicast IPv4 address

- *Uniquely* identifies a network interface in the internet
- 32 bits, usually written in dotted decimal notation

**binary:** 32 bits

example 1: **b**1000 0000 1011 1111 1001 0111 0000 0001

example 2: **b**1000 0001 1100 0000 1100 1000 0000 0010

**dotted decimal:** 4 integers (one integer = 8 bits/ binary digits)

example 1: 128.191.151.1

example 2: 129.192.200.2

**hexadecimal:** 8 hex digits (one hex digit = 4 bits/ binary digits)

example 1: **x**80 bf 97 01

example 2: **x**81 c0 c8 02

# Binary, Decimal and Hexadecimal

Given an integer  $B$  (the basis) any integer can be represented as a string in an alphabet of  $B$  symbols, starting from 0.

	Basis	Alphabet	Example
Decimal	X	$\{0,1, 2,3, 4,5, 6,7, 8,9\}$	200
Binary	II	$\{0,1\}$	1100 1000
Hexadecimal	XVI	$\{0,1, 2,3, 4,5, 6,7, 8,9, a, b, c, d, e, f\}$	c8

Binary  $\longleftrightarrow$  hex is easy: one hex digit ( = nibble) is 4 binary digits

$$c_{hex} = 1100_{bin} \quad 8_{hex} = 1000_{bin} \quad c8_{hex} = 1100\ 1000_{bin}$$

Binary/hex  $\longleftrightarrow$  decimal is best done by a calculator

$$1100\ 1000_{bin} = 128 + 64 + 8 = 200$$

Special Cases to remember

$$f_{hex} = 1111_{bin} = 15_{dec}$$

$$ff_{hex} = 1111\ 1111_{bin} = 255_{dec}$$



# The mask 255.255.254.0 means that the subnet is made of the first ...

- A. 16 bits
- B. 18 bits
- C. 22 bits
- D. 23 bits
- E. 24 bits



Go to [web.speakup.info](https://web.speakup.info) or  
download speakup app

Join room  
46045

# Solution

Answer D

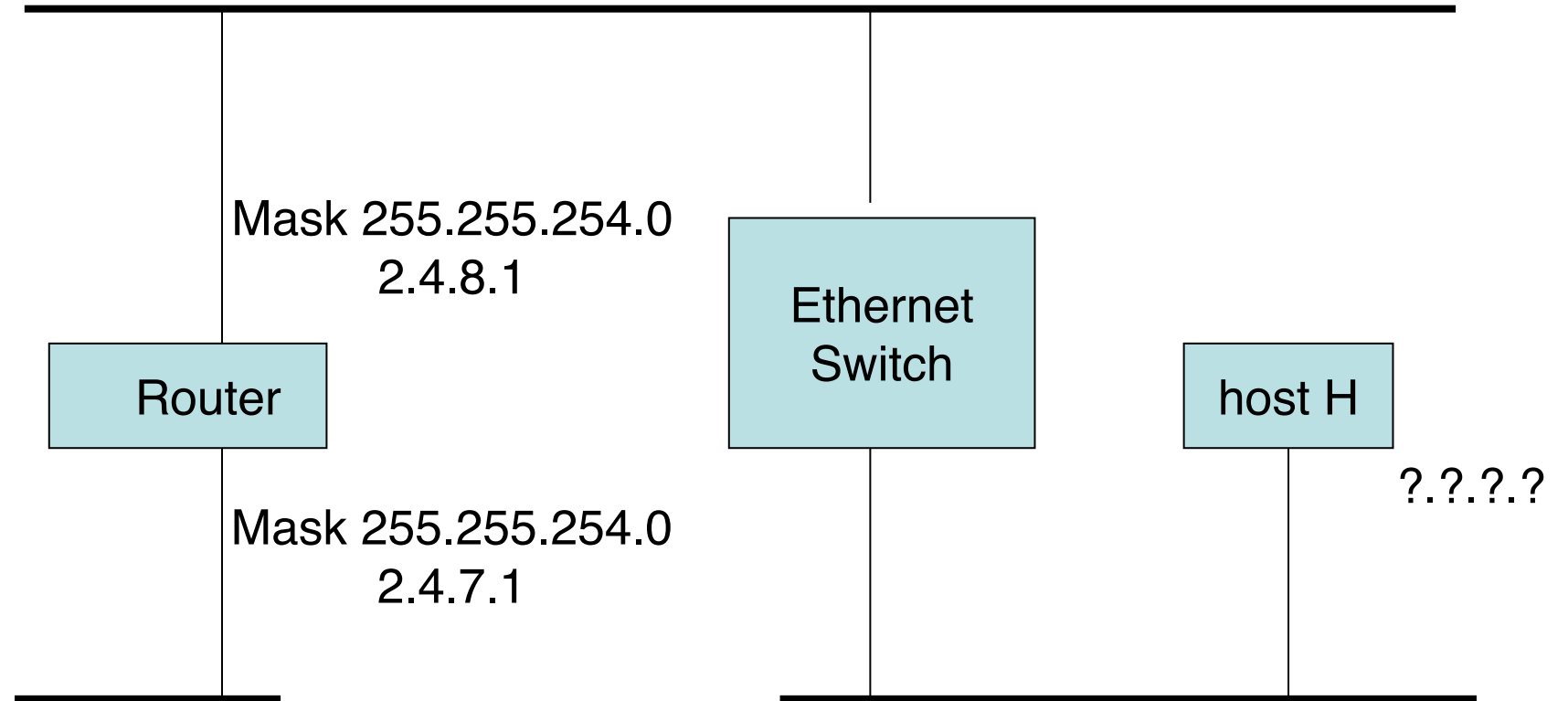
254 = 0b 1111 1110 i.e.

255.255.254.0 = 0b 1111 1111 1111 1111 1111 1110 0000 0000

23 bits equal to 1

# Which address is a valid choice for H ?

- A. only 2.4.8.2
- B. only 2.4.9.1
- C. Both A and B
- D. None





# Solution

Answer C

Router's north interface and H are in the same subnet

So H must have a subnet prefix of 23 bits.

We have:

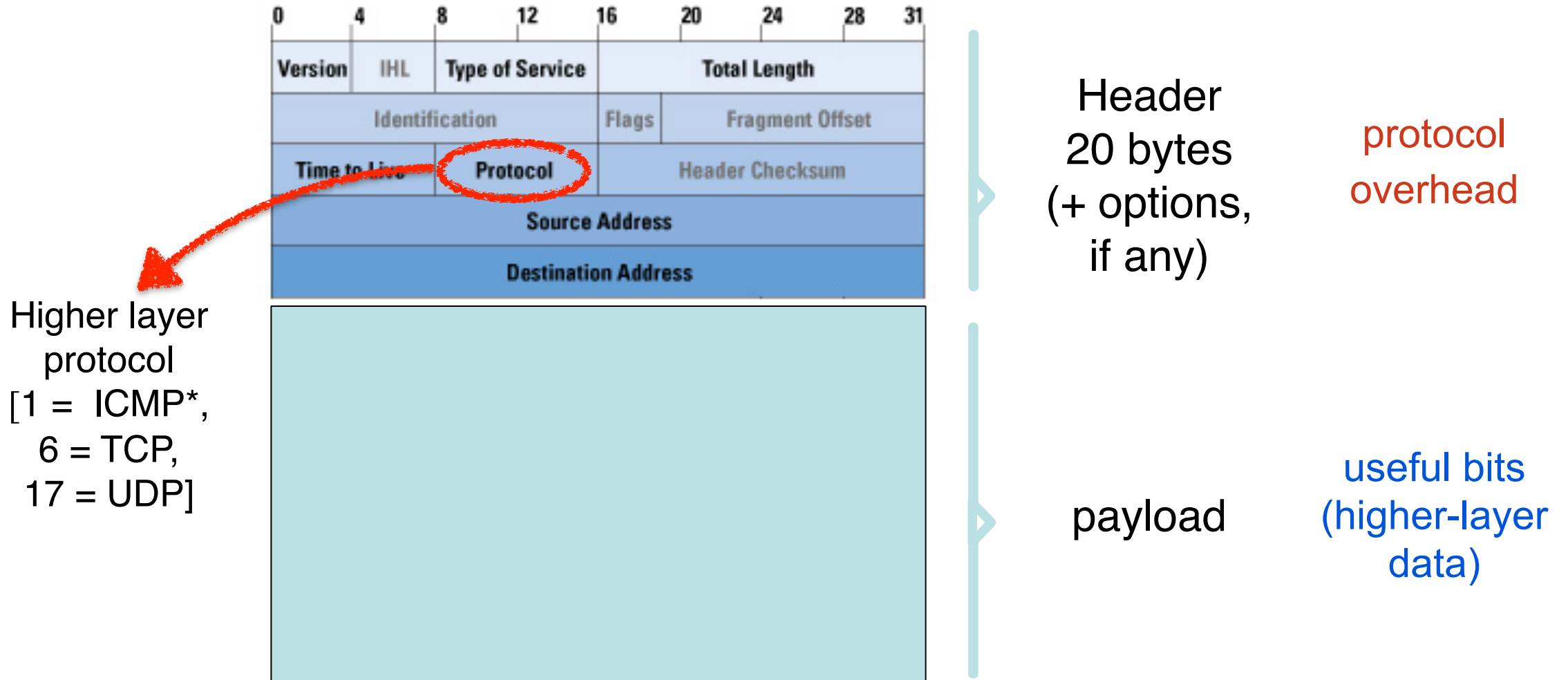
- Router north's subnet prefix:  $2.4.8 / 23 = 0000\ 0010\ 0000\ 0100\ 0000\ 100$
- So A is correct because:  $2.4.8.2 = 0000\ 0010\ 0000\ 0100\ 0000\ 1000\ 0000\ 0010$
- B is also correct because:  $2.4.9.1 = 0000\ 0010\ 0000\ 0100\ 0000\ 1001\ 0000\ 0001$   
and:  $2.4.9 / 23 = 0000\ 0010\ 0000\ 0100\ 0000\ 100$

I.e.: the two prefixes are the *same*:  $2.4.9/23 = 2.4.8/23$  !

# Reserved address blocks

0.0.0.0	<b>absence</b> of address
127/8	<b>loopback</b> addresses (this host, e.g. 127.0.0.1)
10.0.0.0/8, 172.16.0.0/12, 192.168.0.0/16	<b>private</b> addresses (e.g. at home): used by <i>anyone</i> , but <i>not</i> in the public Internet (internet routers drop packets destined to them)
100.64/10	private addresses used only by Internet Service Providers (ISPs)— <b>Carrier Grade NAT addresses</b>
192.88.99/24	<b>IPv6-to-IPv4</b> relay routers
169.254.0.0/16	<b>link local</b> addresses (can be used only by systems in same LAN)
224/4	<b>multicast</b>
240/4	reserved “for experimental/future use” until recently
255.255.255.255/32	link local (LAN) <b>broadcast</b>

# IPv4 Packet Format



\* (ICMP is used to carry error messages at the network layer)

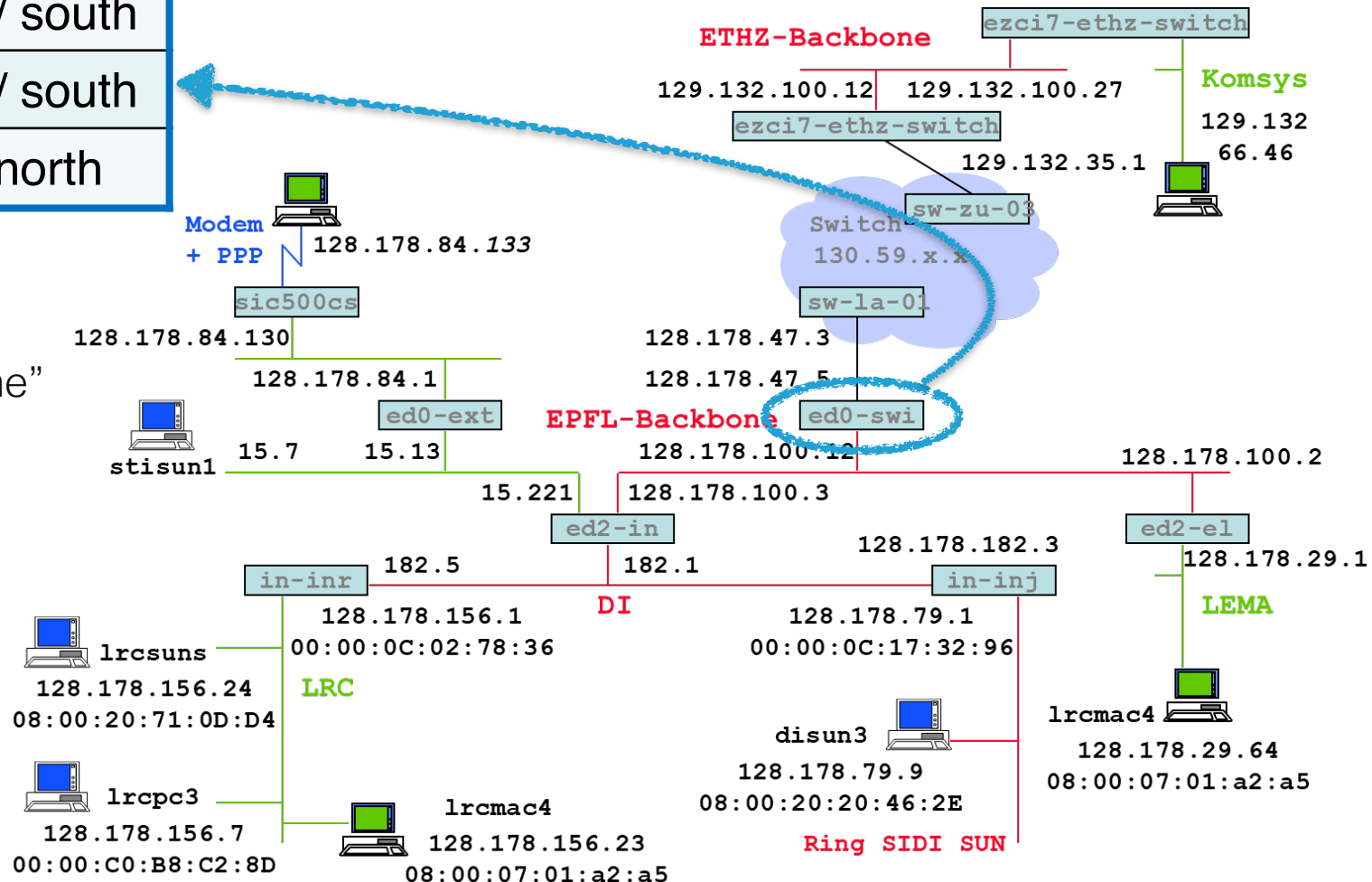
# Forwarding table and longest prefix match: example from EPFL

Destination	Next-Hop / Interface
128.178.29/24	128.178.100.2 / south
128.178/16	128.178.100.3 / south
0/0	128.178.47.3 / north

- destination subnets in *aggregated* form
- next hops identified by both “IP” and “name”
- 0/0 (empty string) = default route

- Longest prefix match means:

```
if packet -> 128.178.*
  if packet -> 128.178.29*
    forward to ed2-e1
  else
    forward to ed2-in
else
  forward to sw-la-01
```



# 3. IPv6

*Why* a new version ?

IPv4 address space is too small (32 bits  $\rightarrow \approx 4 \cdot 10^9$  unique addresses )

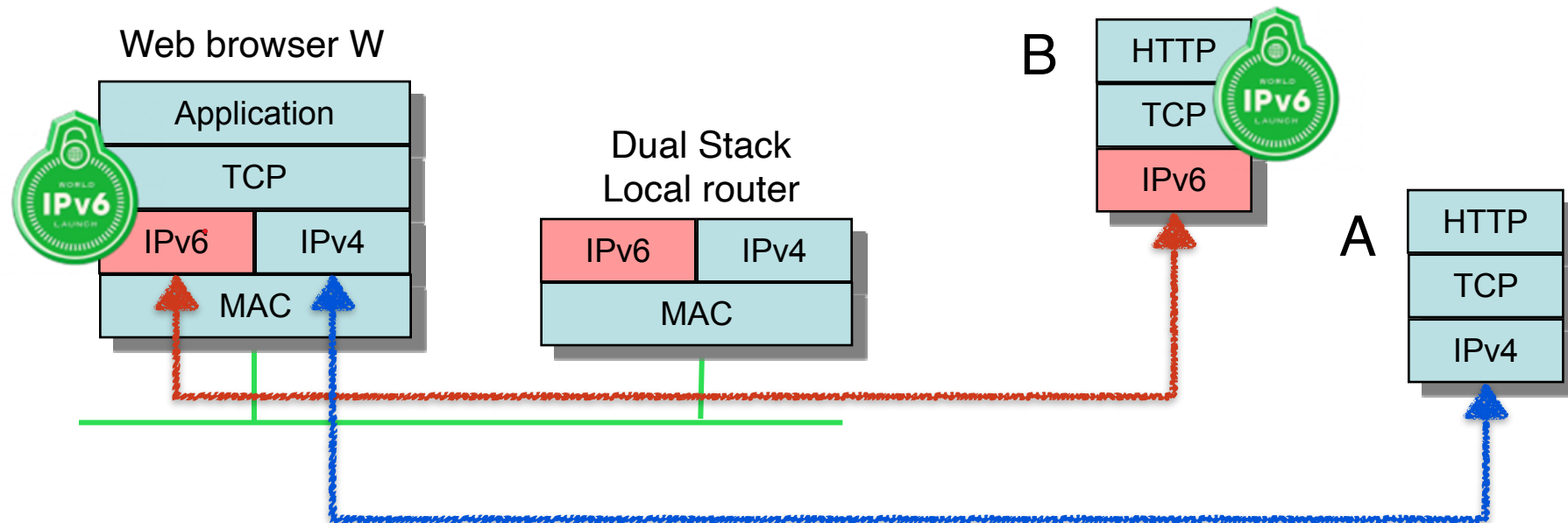
*What* does IPv6 do ?

Redefines packet format with larger addresses of: **128 bits** (  $\approx 3 \cdot 10^{38}$  unique addresses)

Otherwise, it offers essentially the same services as IPv4

*But* IPv6 is *incompatible* with IPv4; routers and hosts must handle separately

A can talk to W, B can talk to W, A and B cannot communicate at the network layer



# IPv6 Addresses and Compression Rules

- We write them in hextets, prefer lower case letters, and separate them by “:” (colon)
  - 1 *hextet* = 1 *piece* = 16 bits = [0-4] hex digits;
  - one IPv6 address uncompressed = 8 hextets
- leading zeroes inside a hextet can be *omitted*
- :: replaces any number of 0s in more than one hextet; can be used *at most once* in address

uncompressed form	compressed form
2002:0000:0000:0000:0000:fff:80b2:0c26	2002::fff:80b2:c26
2001:0620:0618:01a6:0000:20ff:fe78:30f9	2001:620:618:1a6:0:20ff:fe78:30f9

# A few IPv6 global unicast addresses

The block 2000/3 (i.e. 2xxx and 3xxx) is for global/public unicast addresses

2001:620::/32	Switch
2001:620:618::/48	EPFL
2001:620:8::/48	ETHZ
2a02:1200::/27	Swisscom
2001:678::/29	provider independent address
2001::/32	Teredo (tunnels IPv6 in IPv4)
2002::/16	6to4 (tunnels IPv6 in IPv4)

Networks served by a provider use blocks that are *subsets* of the provider's address block, (e.g. check EPFL, ETHZ and SWITCH)



# Reserved address blocks

EPFL  
Private

	::/128	absence of address
	::1/128	loopback address (this host)
	fc00::/7 (i.e. fcxx: and fdxx:) for example: fd24:ec43:12ca:1a6:a00:20ff:fe78:30f9	unique local addresses = <b>private</b> networks (e.g. in EPFL): <i>not</i> to be used in the public Internet
	fe80::/10	link local addresses (used only by systems in same LAN)
	ff00::/8	multicast
	ff02::1:ff00:0/104	solicited node multicast (see NDP later)
	ff02::1/128	link local broadcast
	ff02::2/128	multicast to all link-local routers (in same LAN)

# IPv6 public and private prefixes are structured

an EPFL public address:

2001:620:618:1a6:a00:20ff:fe78:30f9



an EPFL private address:

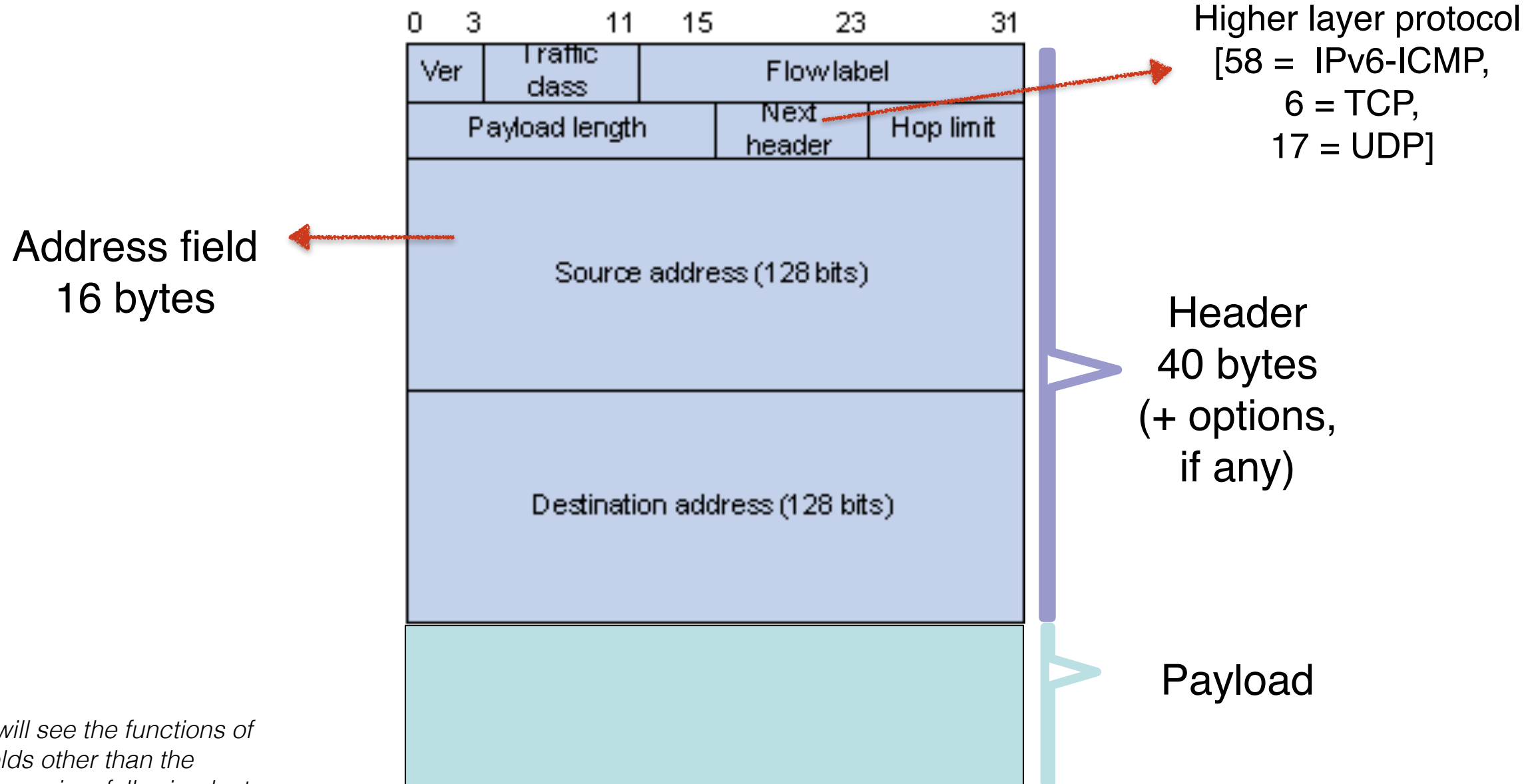
fd24:ec43:12ca:1a6:a00:20ff:fe78:30f9



EPFL private prefix

This is a private address

# IPv6 Packet Format



*\*\*We will see the functions of the fields other than the addresses in a following lecture*

IPv6 Forwarding table: same example from EPFL at ed0-swi

IP address of next hop

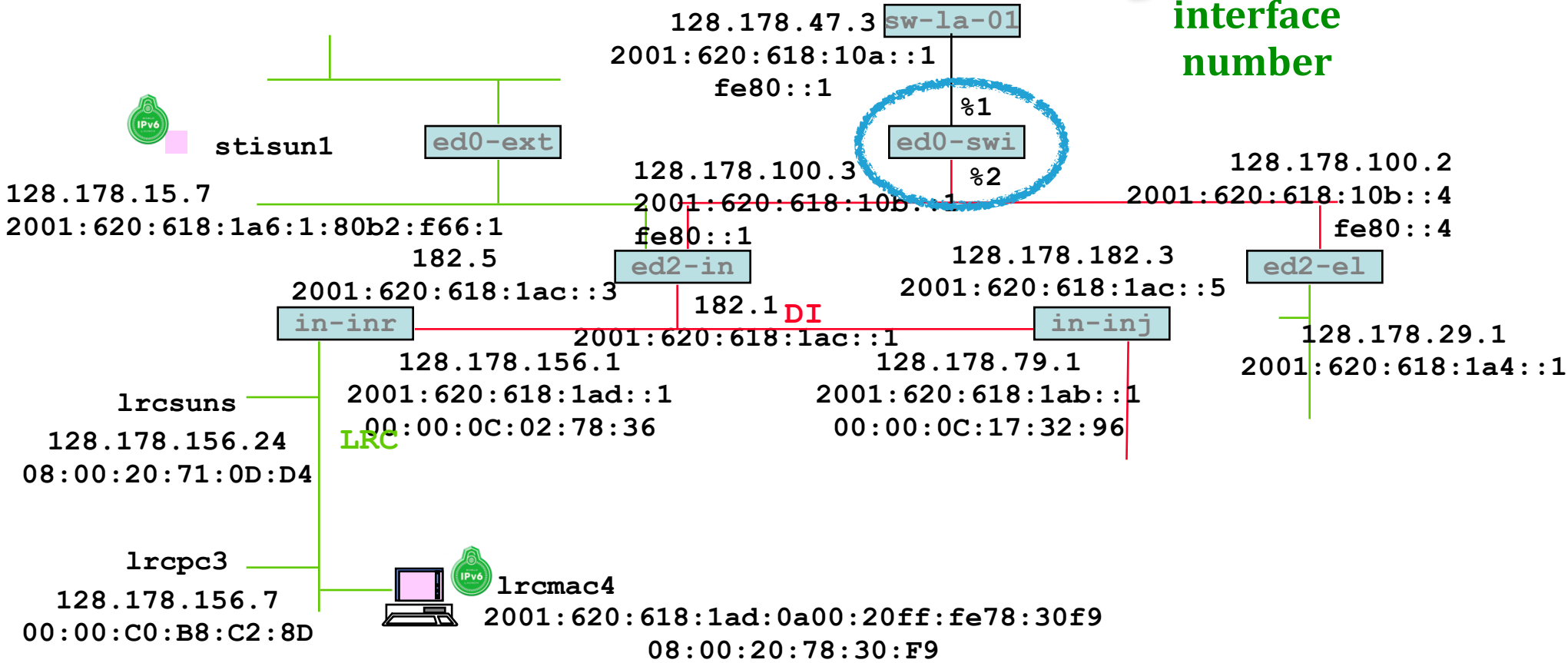
IPv6 forwarding table

Destination	Next-Hop-IP%Interface
2001:620:618:1a4/64	fe80::4%2
2001:620:618/48	fe80::1%2
::/0	fe80::1%1

IPv4 forwarding table

Destination	Next-Hop / Interface
128.178.29/24	128.178.100.2 / south
128.178/16	128.178.100.3 / south
0/0	128.178.47.3 / north

interface number



The dotted decimal notation for  
0102: *ffff* is ...

- A. 1.2.255.255
- B. 16.32.255.255
- C. 228.393.255.255



Go to [web.speakup.info](https://web.speakup.info) or  
download speakup app

Join room  
46045

# Solution

Answer A

Recall the mapping (hex) ff  $\rightarrow$  (decimal) 255

In full, the hexadecimal notation  
«2001::ada:bada» means...

- A. 2001:0ada:bada
- B. 2001:0000:0000:0000:0000:0000:0ada:bada
- C. 2001:0000:0ada:bada
- D. 2001:0000:ada:bada
- E. None of the above



Go to [web.speakup.info](https://web.speakup.info) or  
download speakup app

Join room  
46045



# Solution

Answer B. The convention `::` means as many 0's as required to make the string 128 bits.

Leading zeros are omitted, so that `:bad:` means `:0bad:`