



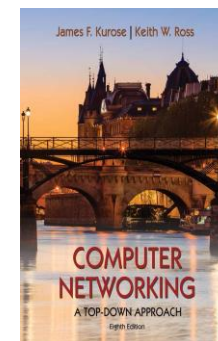
AinShams University
Faculty of Engineering

CSE 351: Computer Networks

Section 8

Eng. Noha Wahdan

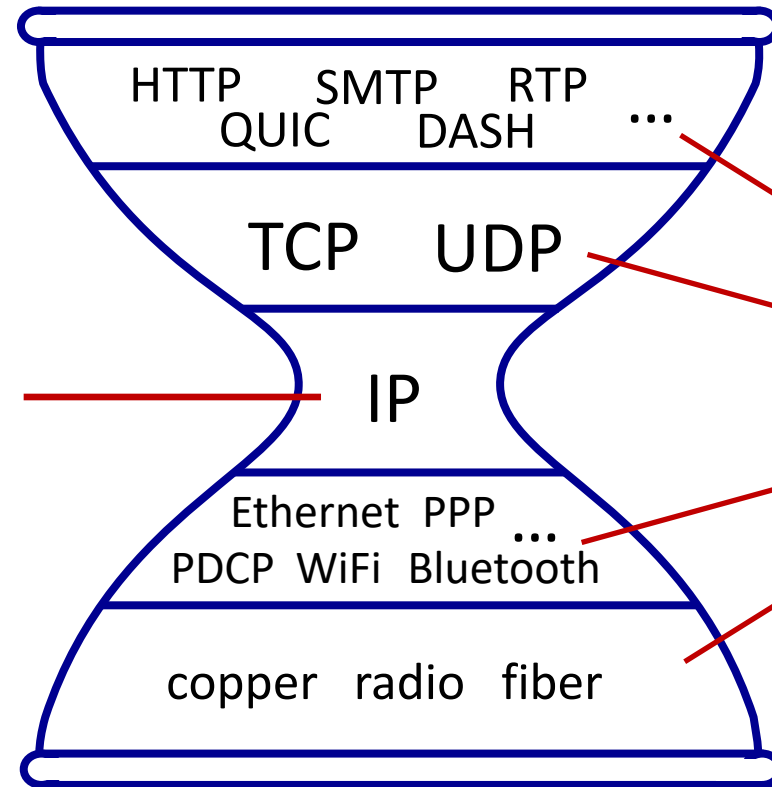
Class textbook:
*Computer Networking: A Top-Down
Approach (8th ed.)*
J.F. Kurose, K.W. Ross
Pearson, 2020
http://gaia.cs.umass.edu/kurose_ross



The IP hourglass

Internet's "thin waist":

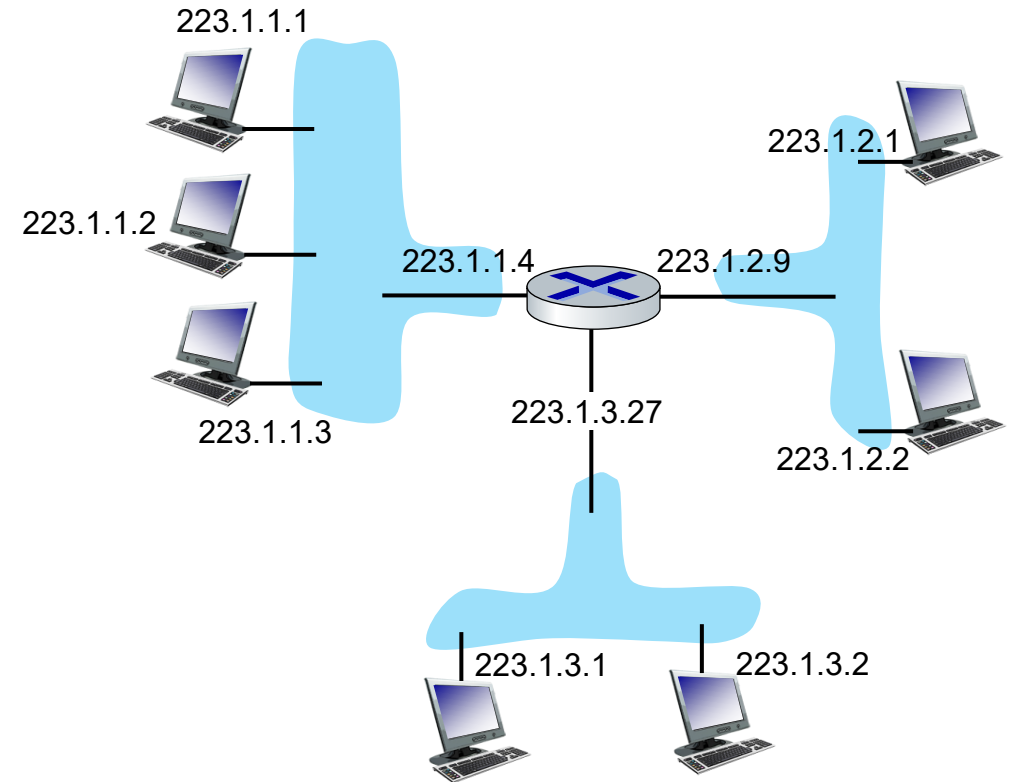
- *one* network layer protocol: IP
- *must* be implemented by every (billions) of Internet-connected devices



many protocols
in physical, link,
transport, and
application
layers

IP addressing: introduction

- **IP address:** 32-bit identifier associated with each host or router *interface*
- **interface:** connection between host/router and physical link
 - router's typically have multiple interfaces
 - host typically has one or two interfaces (e.g., wired Ethernet, wireless 802.11)



dotted-decimal IP address notation:

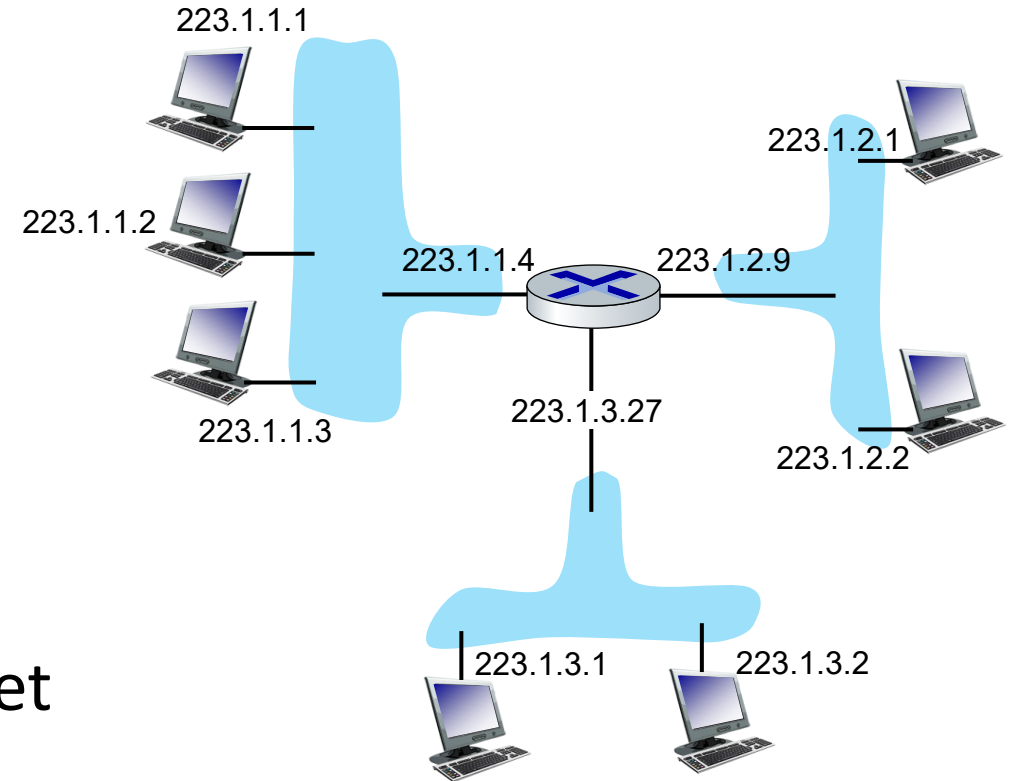
223.1.1.1 = 11011111 00000001 00000001 00000001

223 1 1 1

Network Layer: 4-3

Subnets

- *What's a subnet ?*
 - device interfaces that can physically reach each other **without passing through an intervening router**
- IP addresses have structure:
 - **subnet part**: devices in same subnet have common high order bits
 - **host part**: **remaining** low order bits

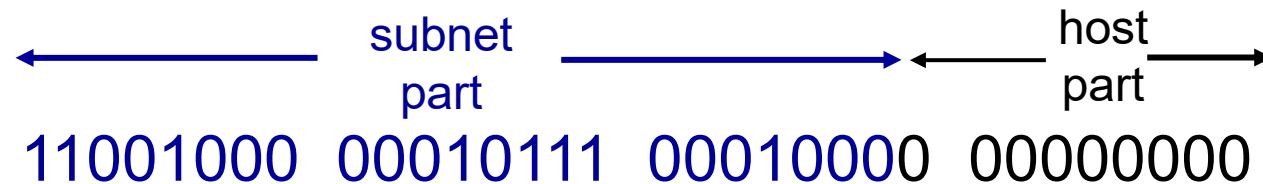


network consisting of 3 subnets

IP addressing: Classless Addressing

CIDR: Classless InterDomain Routing (pronounced “cider”)

- subnet portion of address of arbitrary length
- address format: **a.b.c.d/x**, where x is # bits in subnet portion of address



Slashed notation: 200.23.16.0/23

Subnet mask: 11111111 11111111 11111110 00000000
255.255.254.0

Summary of some Special IP Addresses

| Subnet part | Host part | Type of address |
|-------------|-----------|---|
| Network | All zeros | Network Address (Identifies the network itself) |
| Network | All ones | Network Broadcast Address |
| All ones | | Limited broadcast on the local network |
| All zeros | | This host |

Note: You can obtain the above division either using the **slashed notation** **OR** using the **subnet mask**, as shown in the previous slide.

IP addresses: how to get one?

That's actually **two** questions:

1. Q: How does a *network* get IP address for itself (network part of address)
2. Q: How does a *host* get IP address within its network (host part of address)?

IP addressing:

How does an ISP get block of addresses?

ICANN: Internet Corporation for Assigned Names and Numbers

<http://www.icann.org/>

- allocates IP addresses
- manages DNS

How does network get subnet part of IP address?

From its *provider ISP's* address space

How does a host get IP address ?

Using *DHCP*

Are there enough 32-bit IP addresses?

ICANN allocated last chunk of IPv4 addresses to RRs in 2011

- NAT helps IPv4 address space exhaustion
- IPv6 has 128-bit address space

IP addresses: how to get one?

Q: how does *network* get subnet part of IP address?

A: gets allocated portion of its provider ISP's address space

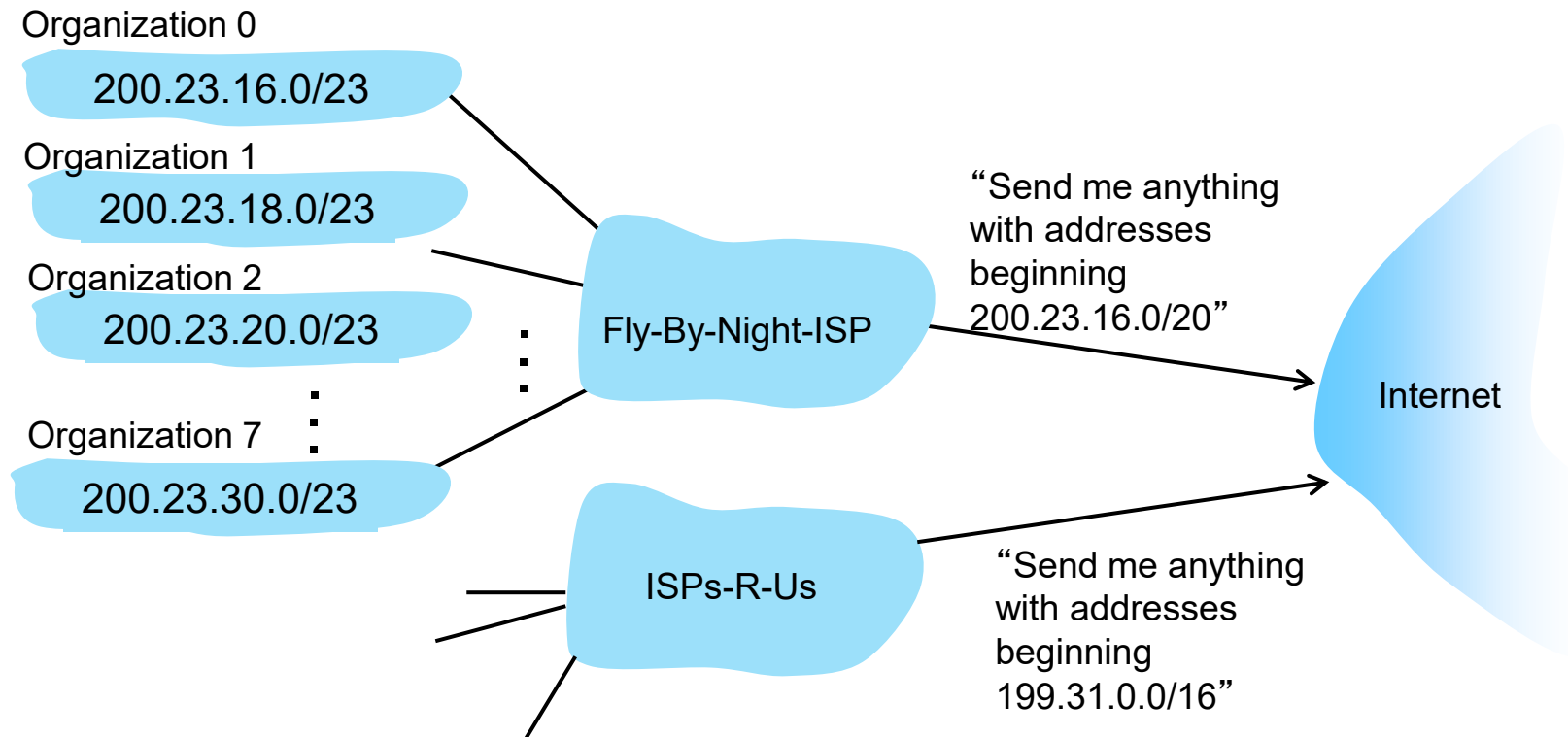
ISP's block 11001000 00010111 00010000 00000000 200.23.16.0/20

ISP can then allocate out its address space in 8 blocks:

| | | | |
|----------------|-----------------------------------|----------|----------------|
| Organization 0 | <u>11001000 00010111 00010000</u> | 00000000 | 200.23.16.0/23 |
| Organization 1 | <u>11001000 00010111 00010010</u> | 00000000 | 200.23.18.0/23 |
| Organization 2 | <u>11001000 00010111 00010100</u> | 00000000 | 200.23.20.0/23 |
| ... | | | |
| Organization 7 | <u>11001000 00010111 00011110</u> | 00000000 | 200.23.30.0/23 |

Hierarchical addressing: route aggregation

hierarchical addressing allows efficient advertisement of routing information:



DHCP: Dynamic Host Configuration Protocol

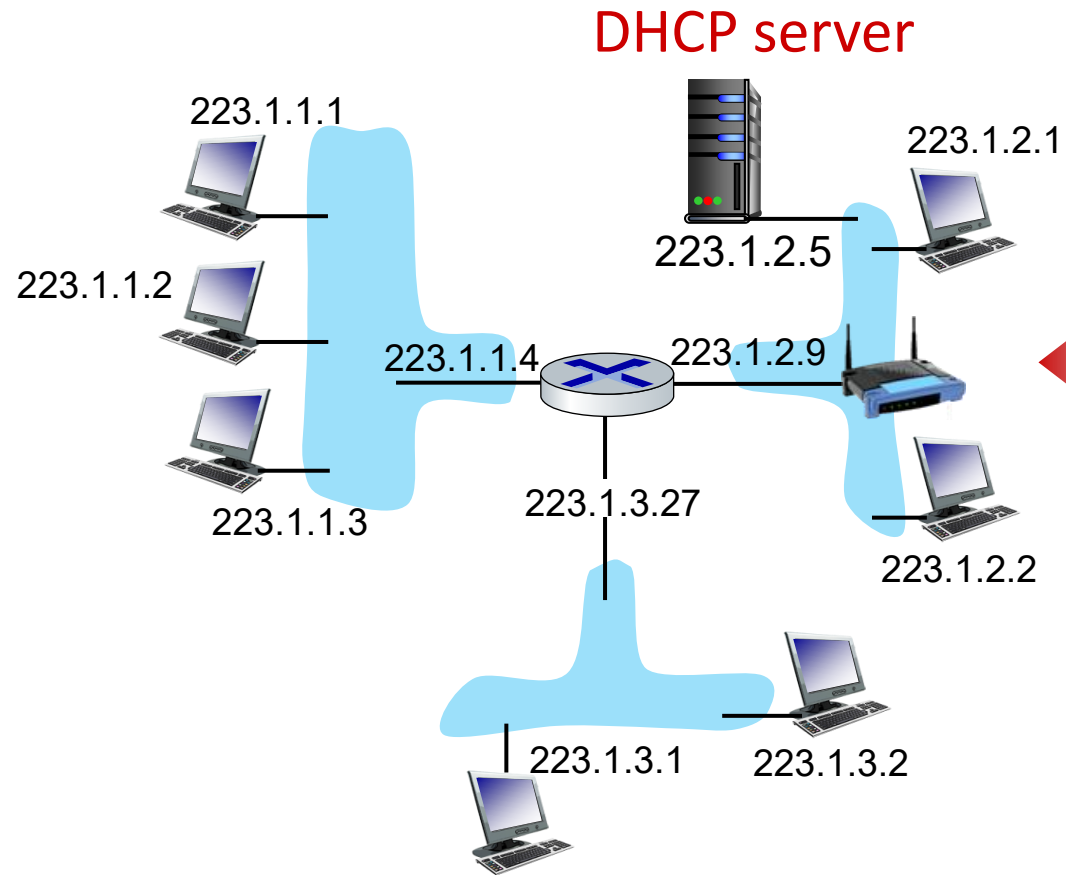
goal: host *dynamically* obtains IP address from network server when it “joins” network

- can renew its lease on address in use
- allows reuse of addresses (only hold address while connected/on)
- support for mobile users who join/leave network

DHCP overview:

- host broadcasts **DHCP discover** msg [optional]
- DHCP server responds with **DHCP offer** msg [optional]
- host requests IP address: **DHCP request** msg
- DHCP server sends address: **DHCP ack** msg

DHCP client-server scenario



Typically, DHCP server will be co-located in router, serving all subnets to which router is attached



arriving **DHCP client** needs address in this network

DHCP client-server scenario

DHCP server: 223.1.2.5



DHCP discover

src : 0.0.0.0, 68
dest.: 255.255.255.255, 67
yiaddr: 0.0.0.0
transaction ID: 654

Arriving client



DHCP offer

src: 223.1.2.5, 67
dest: 255.255.255.255, 68
yiaddr: 223.1.2.4
transaction ID: 654
lifetime: 3600 secs

DHCP request

src: 0.0.0.0, 68
dest.: 255.255.255.255, 67
yiaddr: 223.1.2.4
transaction ID: 655
lifetime: 3600 secs

DHCP ACK

src: 223.1.2.5, 67
dest: 255.255.255.255, 68
yiaddr: 223.1.2.4
transaction ID: 655
lifetime: 3600 secs

The two steps above can be skipped “if a client remembers and wishes to reuse a previously allocated network address” [RFC 2131]

DHCP: more than IP addresses

DHCP can return more than just allocated IP address on subnet:

- IP address of first-hop router for client
- name and IP address of DNS sever
- network mask (indicating network versus host portion of address)

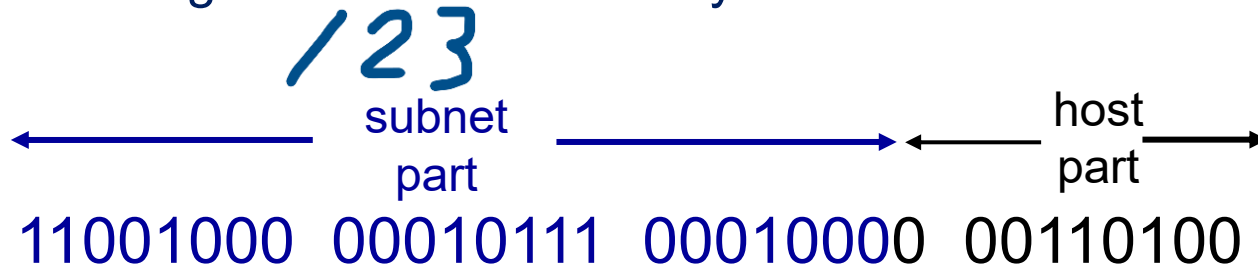
Example 1:

Q: Given the host address 200.23.16.52/23, what is the network address of the subnet that it belongs to? What is broadcast address for this subnet?

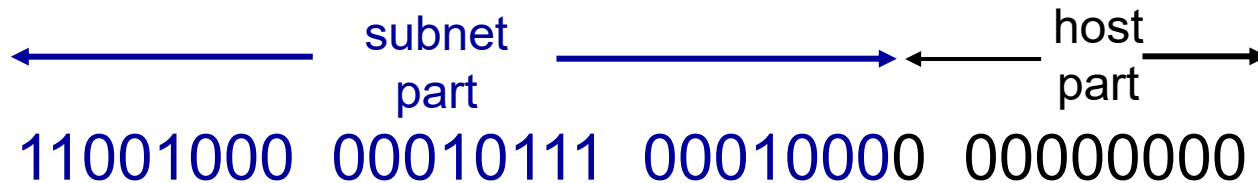
First of all, **always** convert given IP address from dotted decimal notation address to binary to avoid possible confusion.

Solve in binary, then convert to dotted decimal notation if needed.

A: The given address in binary is



- The **Network address** will be:



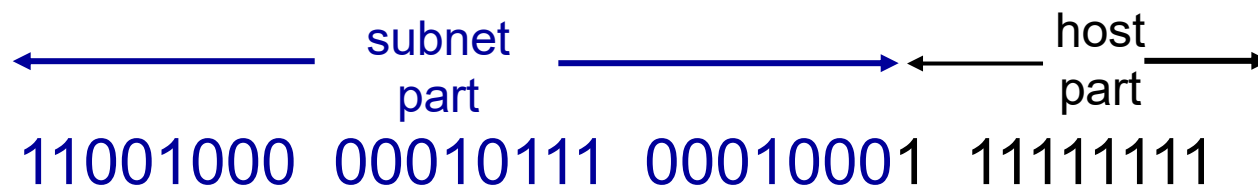
Slashed notation: 200.23.16.0/23

Example 1 (Cont'd):

Q: Given the host address 200.23.16.52/23, what the network address of the subnet that it belongs to? What is broadcast address for this subnet?



- The **Broadcast address** will be:



Slashed notation: 200.23.17.255/23

Example 1 (Cont'd):

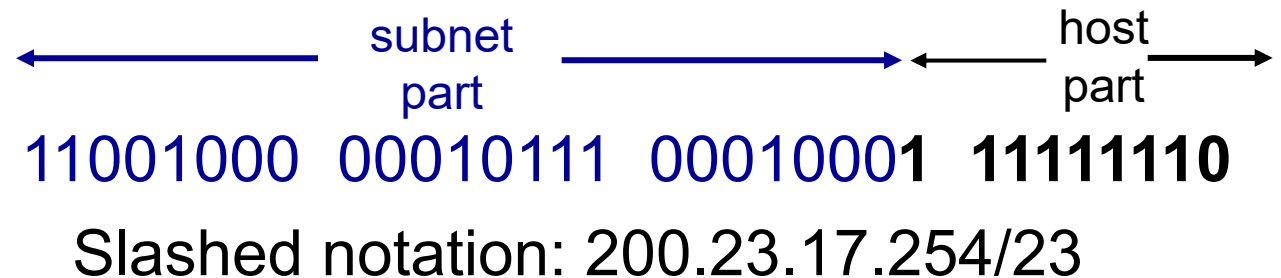
Q: Given the host address 200.23.16.52/23, what the network address of the subnet that it belongs to? What is broadcast address for this subnet?

- A: The host addresses in this subnet will be all addresses in between them. This means that all of them will share the same network address in the subnet part, varying the host part.

First host address will be



Last host address will be



Example 2:

- Consider a subnet with prefix **128.119.40.128/26**.
Give an example of one IP address that can be assigned to this network.

Any IP address in between the range:

from **128.119.40.128**

Network address

1000 0000.0111 0111.0010 1000.1000 0000

to **128.119.40.191**

Broadcast address

1000 0000.0111 0111.0010 1000.1011 1111

First host address is: 1000 0000.0111 0111.0010 1000.1000 0001

Last host address is: 1000 0000.0111 0111.0010 1000.1011 1110

Example 3:

- Suppose an ISP owns the block of addresses of the form **128.119.40.64/26**.

Suppose it wants to create four subnets from this block, with each block having the same number of IP addresses.

What are the prefixes (of form a.b.c.d/x) for the four subnets?

128.119.40.64/26 ->> 1000 0000.0111 0111.0010 1000.0100 0000

Four equal size subnets:

128.119.40.64/28, 1000 0000.0111 0111.0010 1000.0100 0000

128.119.40.80/28, 1000 0000.0111 0111.0010 1000.0101 0000

128.119.40.96/28, 1000 0000.0111 0111.0010 1000.0110 0000

128.119.40.112/28 1000 0000.0111 0111.0010 1000.0111 0000

Note: Each of the four subnets can have $2^4-2 = 14$ host addresses

Sheet 5: Question 28

A network on the Internet has a subnet mask of 255.255.240.0. What is the maximum number of hosts it can handle?

1) Convert the subnet to binary (Each decimal number is converted to 8 binary bits):

11111111. 11111111. 11110000.00000000

2) Count the number of zeros in the subnet mask, as it represents the host portion----->> 12 bits

3) Number of possible host addresses = $2^{12} = 4096$ addresses.

4) Number of possible hosts = $4096 - 2 = 4094$ hosts.

Sheet 5: Question 30

A large number of consecutive IP addresses are available starting at 198.16.0.0. Suppose that four organizations, A, B, C, and D, request 4000, 2000, 4000, and 8000 addresses, respectively, and in that order. For each of these, give the first IP address assigned, the last IP address assigned, and the mask in the w.x.y.z/s notation.

First of all, IP address ranges will be allocated in blocks of power of 2. So, the four organizations will be allocated IPs as

A:4096 (2^{12})

B:2048(2^{11})

C:4096(2^{12})

D:8192(2^{13}).

Sheet 5: Question 30

A large number of consecutive IP addresses are available starting at 198.16.0.0. Suppose that four organizations, A, B, C, and D, request 4000, 2000, 4000, and 8000 addresses, respectively, and in that order. For each of these, give the first IP address assigned, the last IP address assigned, and the mask in the w.x.y.z/s notation.

Remaining unused IPs are wasted. IPs will be allocated to the organizations contiguously.

This means that A needs 12 bits for host portion, B needs 11 bits for host portion, C needs 12 bits for host portion, and D needs 13 bits for host portion.

Sheet 5: Question 30

| | 1 ST IP ADDRESS | LAST IP ADDRESS |
|---|---|--|
| A | 11000110.00010000.00000000.00000000 198.16.0.0 | 11000110.00010000.00001111.11111111 198.16.15.255 |

Sheet 5: Question 30

| | 1 ST IP ADDRESS | LAST IP ADDRESS |
|----------|--|--|
| A | 11000110.00010000.00000000.00000000 198.16.0.0 | 11000110.00010000.00001111.11111111 198.16.15.255 |
| B | 11000110.00010000.00010000.00000000 198.16.16.0 | 11000110.00010000.00010111.11111111 198.16.23.255 |

Sheet 5: Question 30

| | 1 ST IP ADDRESS | LAST IP ADDRESS |
|----------|--|--|
| A | 11000110.00010000.00000000.00000000 198.16.0.0 | 11000110.00010000.00001111.11111111 198.16.15.255 |
| B | 11000110.00010000.00010000.00000000 198.16.16.0 | 11000110.00010000.00010111.11111111 198.16.23.255 |
| C | 11000110.00010000.00100000.00000000 198.16.32.0 | 11000110.00010000.00101111.11111111 198.16.47.255 |

Sheet 5: Question 30

| | 1 ST IP ADDRESS | LAST IP ADDRESS |
|----------|--|--|
| A | 11000110.00010000.00000000.00000000 198.16.0.0 | 11000110.00010000.00001111.11111111 198.16.15.255 |
| B | 11000110.00010000.00010000.00000000 198.16.16.0 | 11000110.00010000.00010111.11111111 198.16.23.255 |
| C | 11000110.00010000.00100000.00000000 198.16.32.0 | 11000110.00010000.00101111.11111111 198.16.47.255 |
| D | 11000110.00010000.01000000.00000000 198.16.64.0 | 11000110.00010000.01011111.11111111 198.16.95.255 |

Sheet 5: Question 30

A needs 12 bits for host portion, B needs 11 bits for host portion, C needs 12 bits for host portion, and D needs 13 bits for host portion.

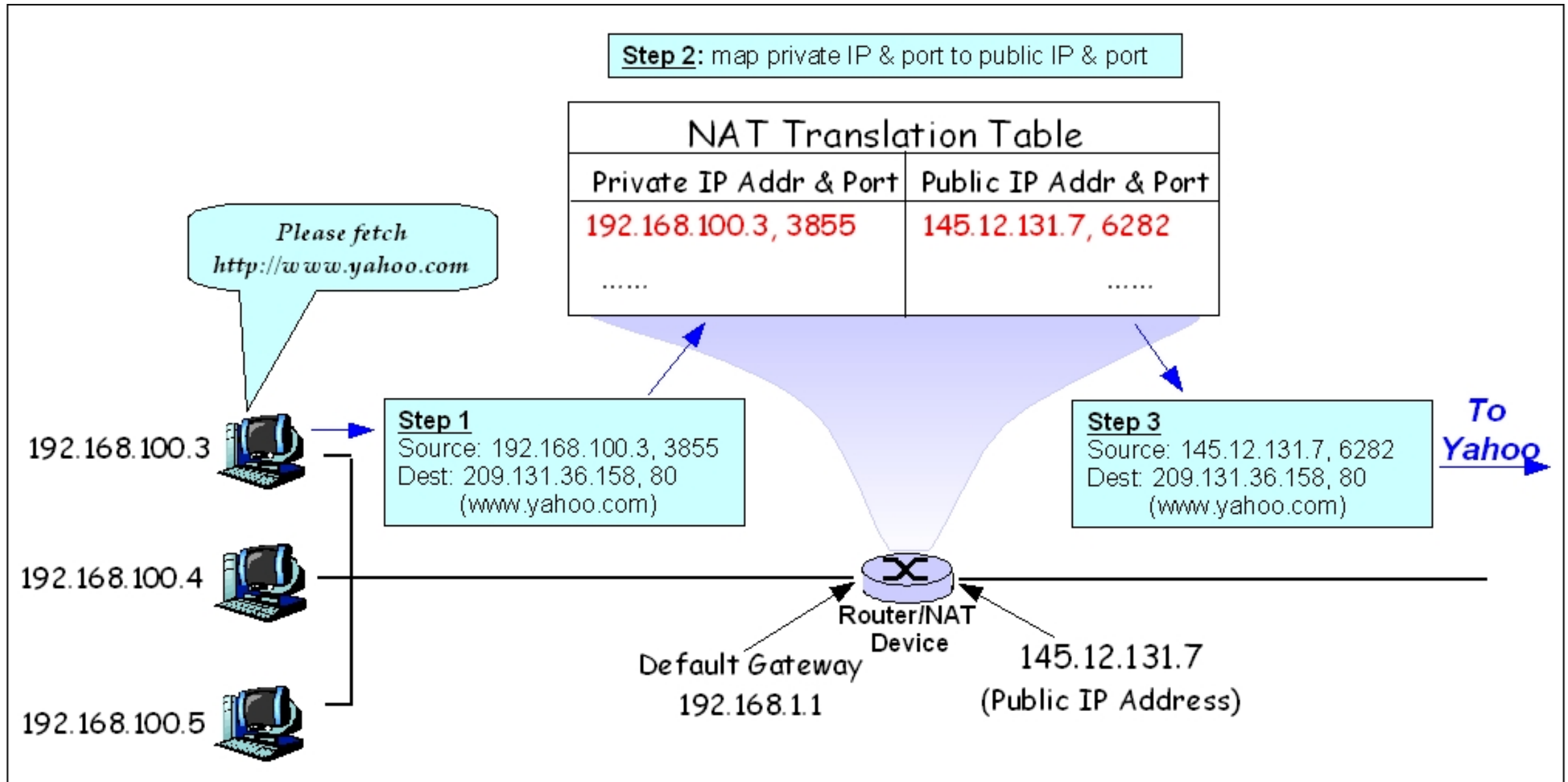
A's network mask will be 198.16.0.0/20.

B's network mask will be 198.16.16.0/21.

C's network mask will be 198.16.32.0/20.

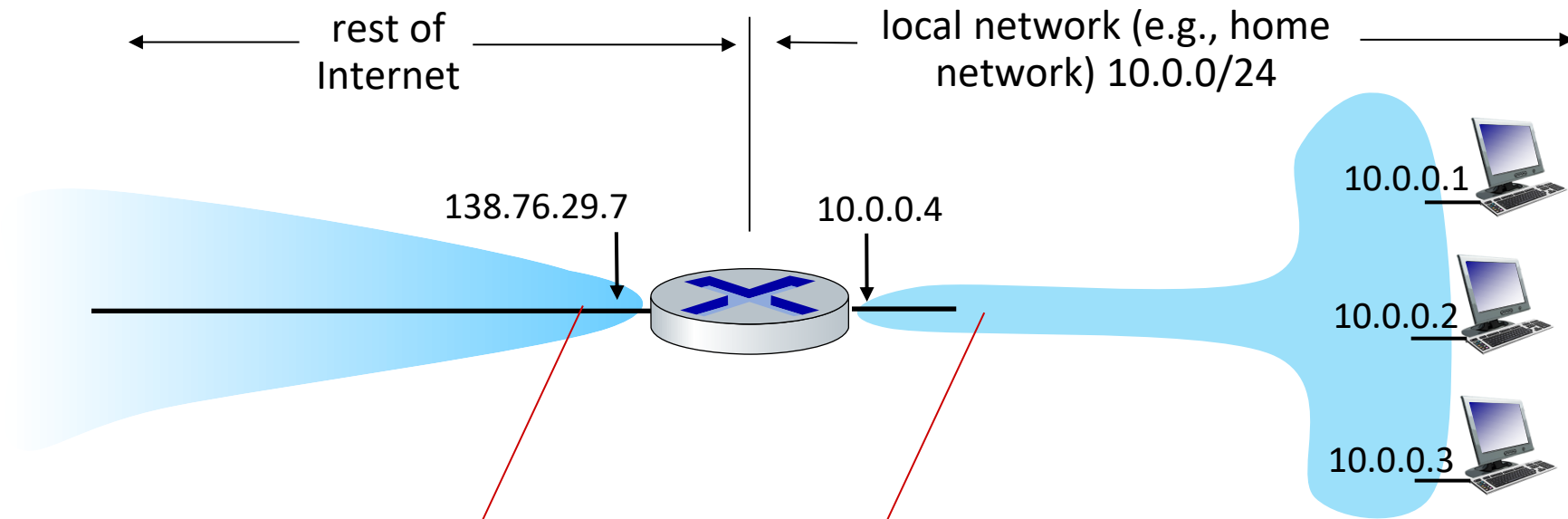
D's network mask will be 198.16.64.0/19.

NAT: network address translation



NAT: network address translation

NAT: all devices in local network share just **one** IPv4 address as far as outside world is concerned

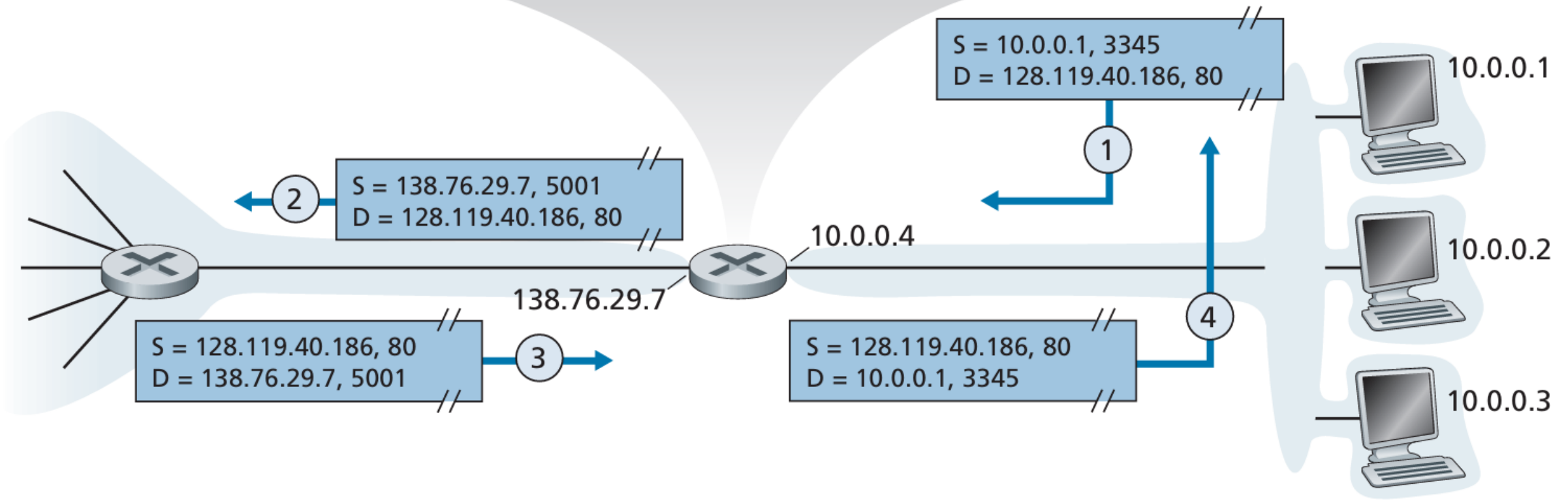


all datagrams *leaving* local network have *same* source NAT IP address: 138.76.29.7, but *different* source port numbers

datagrams with source or destination in this network have 10.0.0/24 address for source, destination (as usual)

NAT: network address translation

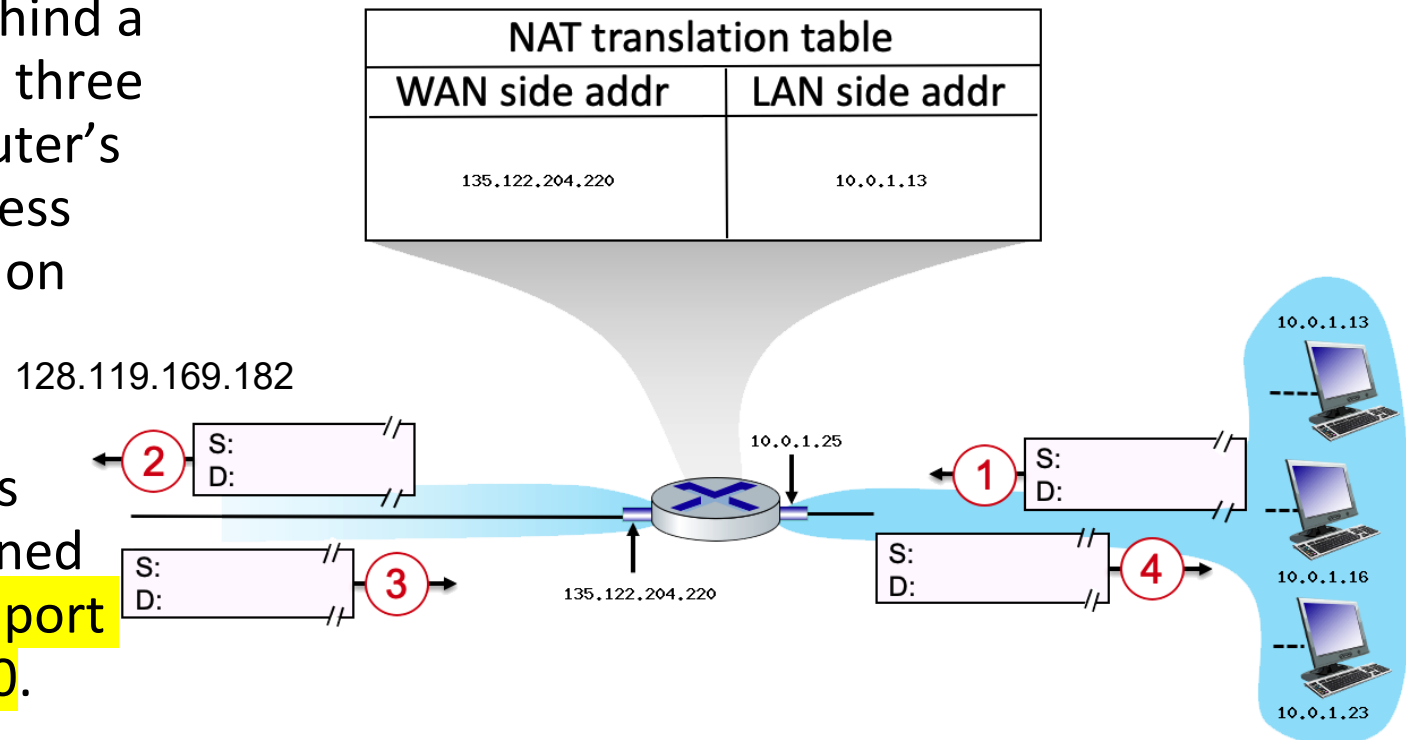
| NAT translation table | |
|-----------------------|----------------|
| WAN side | LAN side |
| 138.76.29.7, 5001 | 10.0.0.1, 3345 |
| ... | ... |



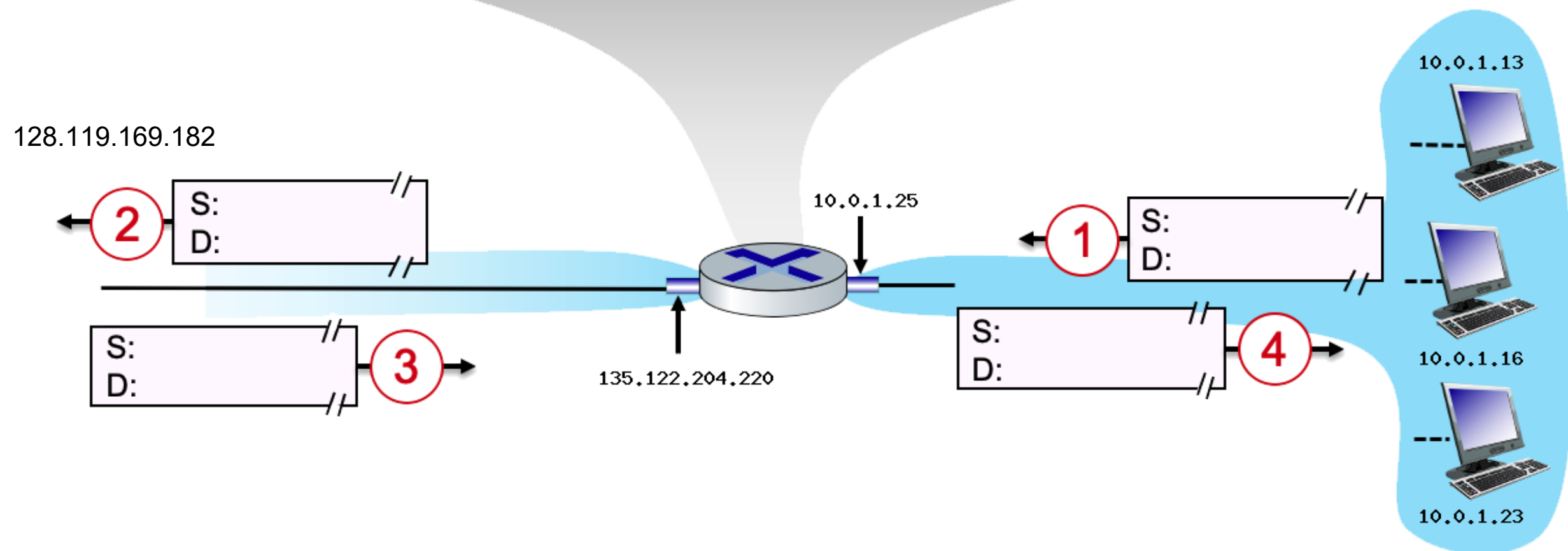
NAT: network address translation

Three hosts are in a local network behind a NAT'd router that sits between these three hosts and the larger Internet. The router's interface on the LAN side has IP address **10.0.1.25**, while the router's address on the Internet side has IP address **135.122.204.220**.

Suppose that the host with IP address **10.0.1.13** sends an IP datagram destined to host **128.119.169.182**. The **source port is 3344**, and the **destination port is 80**.

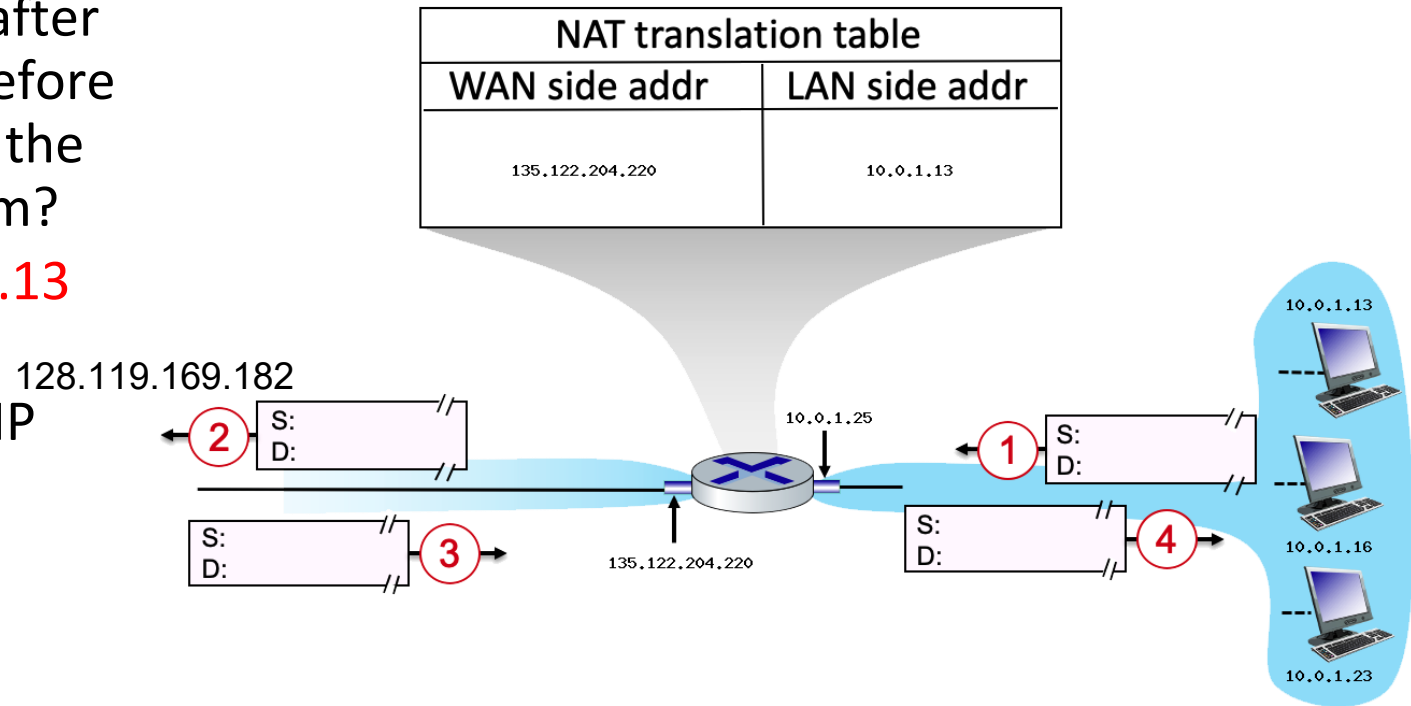


| NAT translation table | |
|-----------------------|---------------|
| WAN side addr | LAN side addr |
| 135.122.204.220 | 10.0.1.13 |



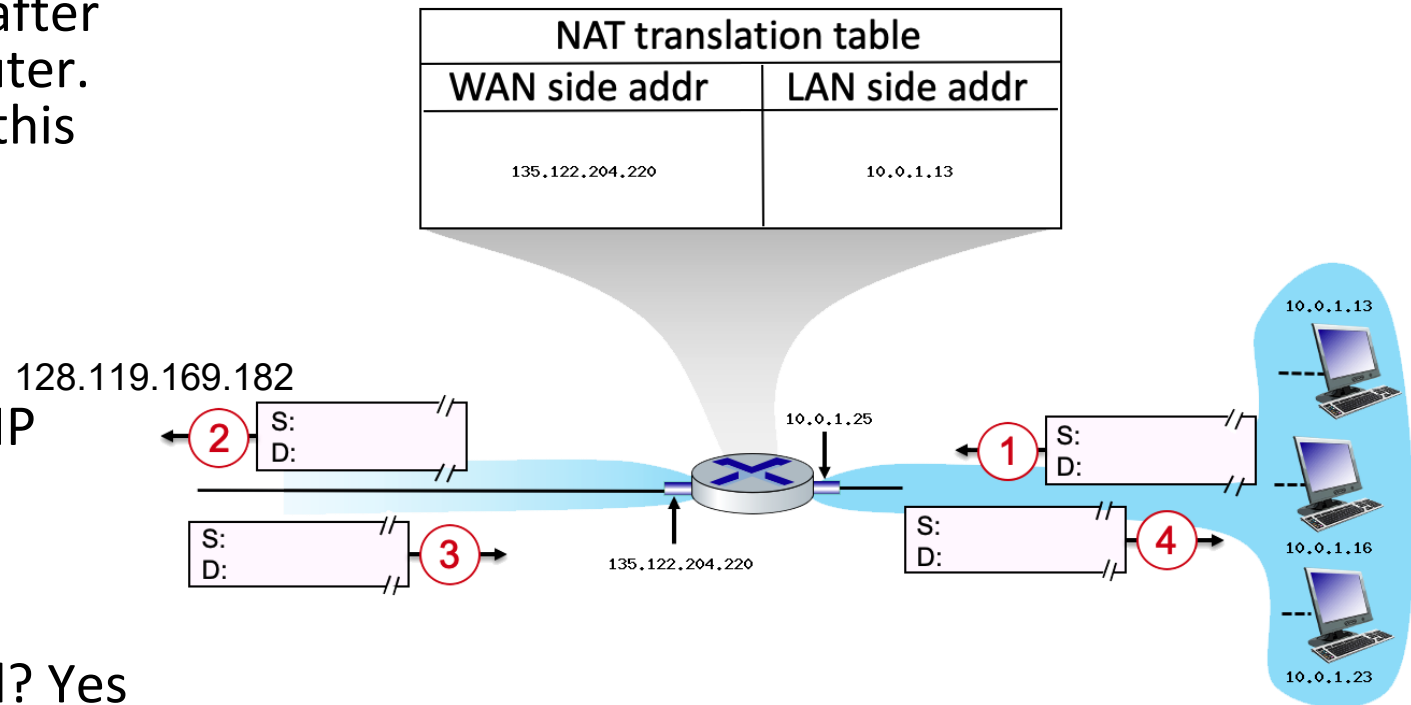
NAT: network address translation

- Consider the datagram at step 1, after it has been sent by the host but before it has reached the router. What is the source IP address for this datagram?
- The local host's IP, which is 10.0.1.13
- At step 1, what is the destination IP address?
- The remote machine's IP, which is 128.119.169.182



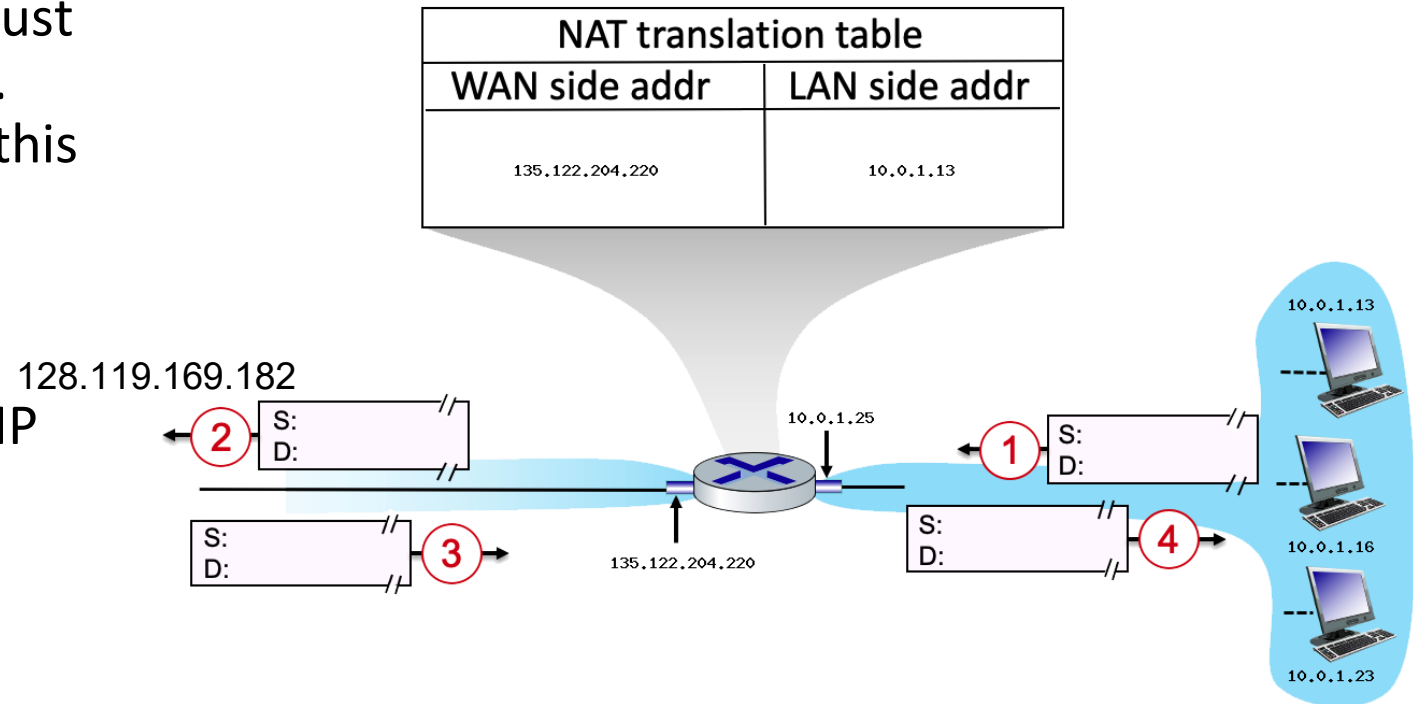
NAT: network address translation

- Consider the datagram at step 2, after it has been transmitted by the router. What is the source IP address for this datagram?
- The router's public IP, which is 135.122.204.220
- At step 2, what is the destination IP address for this datagram?
- The remote machine's IP, which is 128.119.169.182
- Will the source port have changed? Yes or No.
- Yes, the NAT will change the source port.



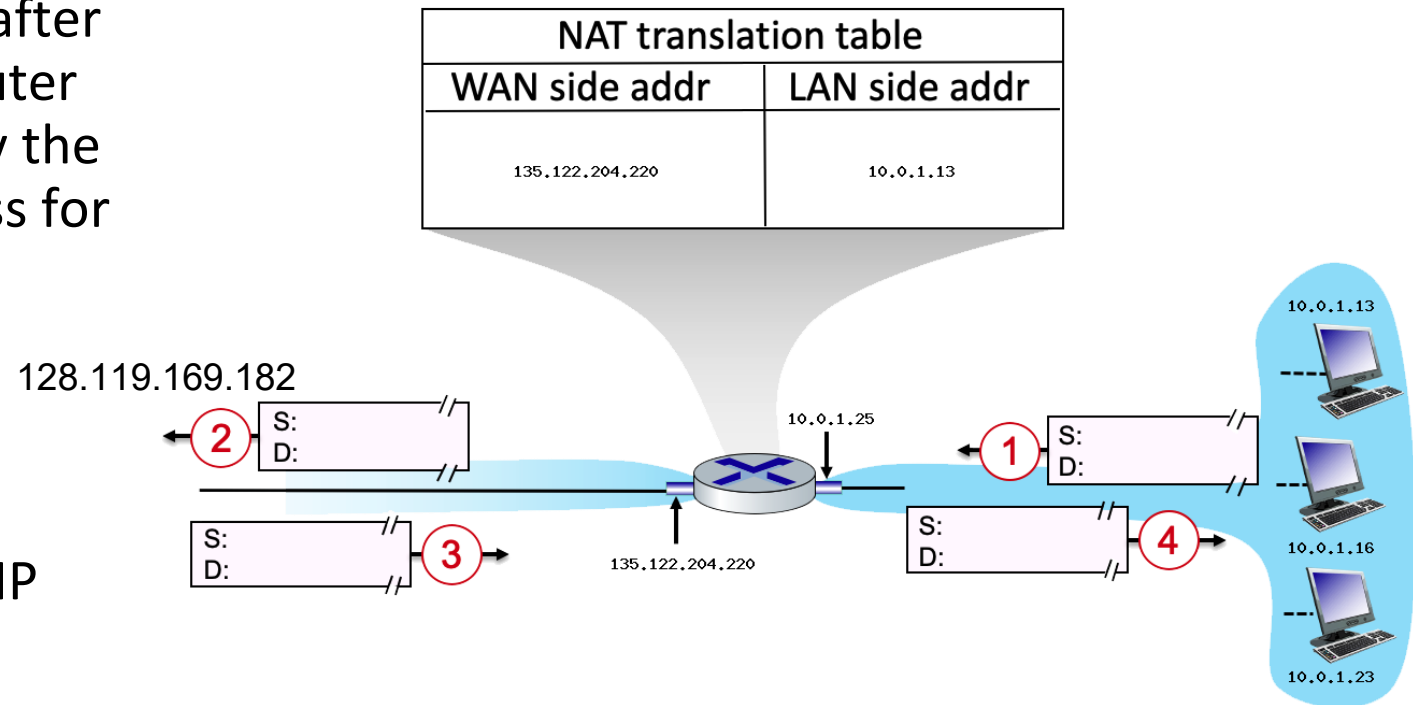
NAT: network address translation

- Consider the datagram at step 3, just before it is received by the router.
What is the source IP address for this datagram?
- The remote machine's IP, which is 128.119.169.182
- At step 3, what is the destination IP address for this datagram?
- The router's public IP, which is 135.122.204.220



NAT: network address translation

- Consider the datagram at step 4, after it has been transmitted by the router but before it has been received by the host. What is the source IP address for this datagram?
- The remote machine's IP, which is 128.119.169.182
- At step 4, what is the destination IP address for this datagram?
- The local host's IP, which is 10.0.1.13



NAT: network address translation

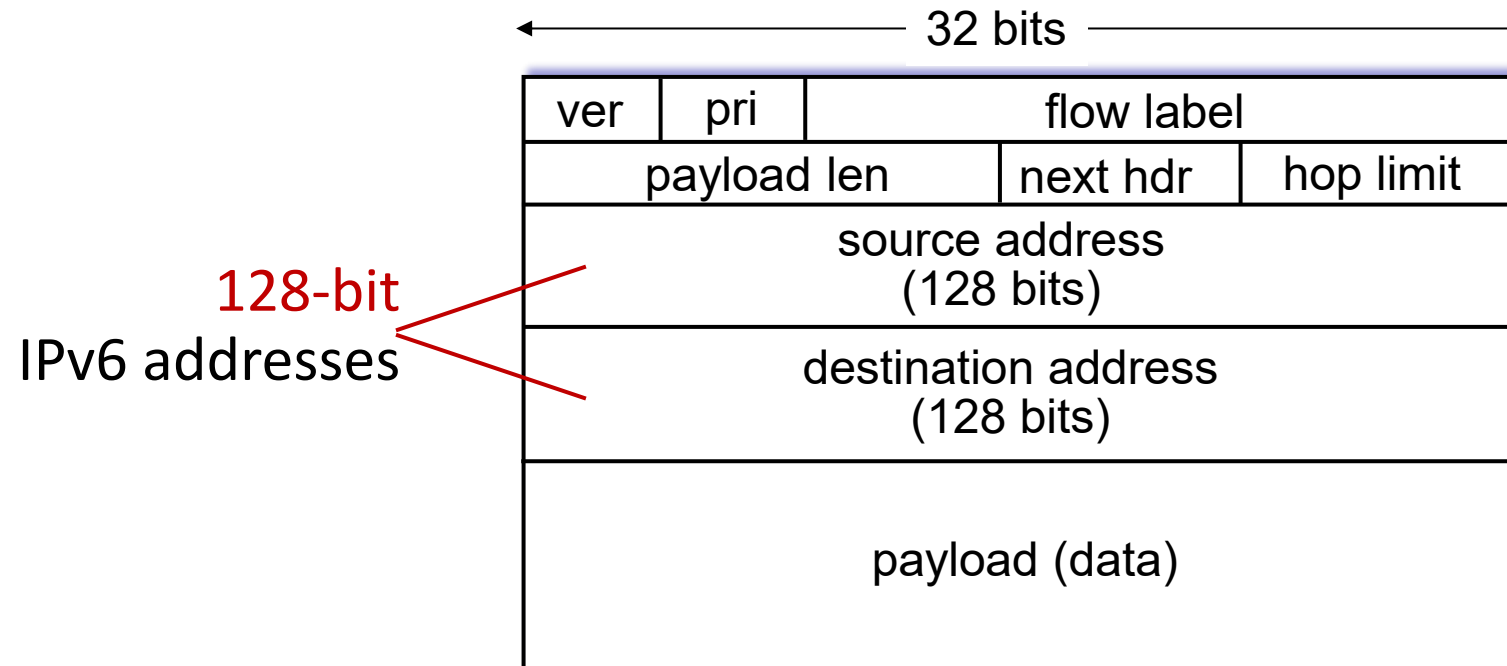
implementation: NAT router must (transparently):

- **outgoing datagrams: replace** (source IP address, port #) of every outgoing datagram to (NAT IP address, new port #)
 - remote clients/servers will respond using (NAT IP address, new port #) as destination address
- **remember (in NAT translation table)** every (source IP address, port #) to (NAT IP address, new port #) translation pair
- **incoming datagrams: replace** (NAT IP address, new port #) in destination fields of every incoming datagram with corresponding (source IP address, port #) stored in NAT table

IPv6: motivation

- **initial motivation:** 32-bit IPv4 address space would be completely allocated
- additional motivation:
 - speed processing/forwarding: 40-byte fixed length header
 - enable different network-layer treatment of “flows”

IPv6 datagram format



What's missing (compared with IPv4):

- no checksum (to speed processing at routers)
- no fragmentation/reassembly
- no options (available as upper-layer, next-header protocol at router)

Sheet 5: Question 40

IPv6 uses 16-byte addresses. If a block of 1 million addresses is allocated every picosecond, how long will the addresses last?

With 16×8 bit addresses → There are 2^{128} possible addresses

If the allocation rate is $\frac{1 \times 10^6 \text{ addresses}}{1 \times 10^{-12} \text{ seconds}} = \frac{10^{18} \text{ addresses}}{\text{sec}}$

The addresses will last for $\frac{2^{128}}{10^{18}} = \frac{3.4 \times 10^{38}}{10^{18}}$
 $= 3.4 \times 10^{20} \text{ seconds} = 10^{13} \text{ years}$

Thanks