Introduction to Computer Networks

CS640

Efficient Addressing

https://pages.cs.wisc.edu/~mgliu/CS640/S26/index.html

Ming Liu
mgliu@cs.wisc.edu

Outline

- Last
  - IP Introduction
- Today
  - Efficient Addressing
- Announcements
  - Lab2 due on 03/03/2026

Recap

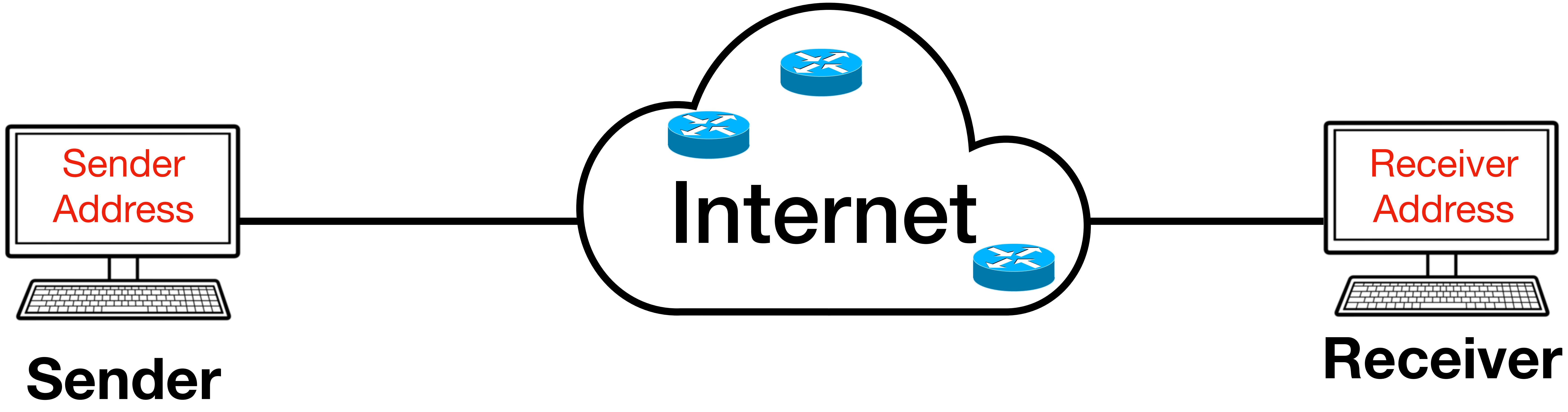
- Key Questions
  - Why is IP so powerful?

- Terminology
  - Best-effort host-to-host service model
  - IP header and its fields
  - ICMP
  - Router and routing

Why do we care about IP addressing?

IP — A Single Logical Network

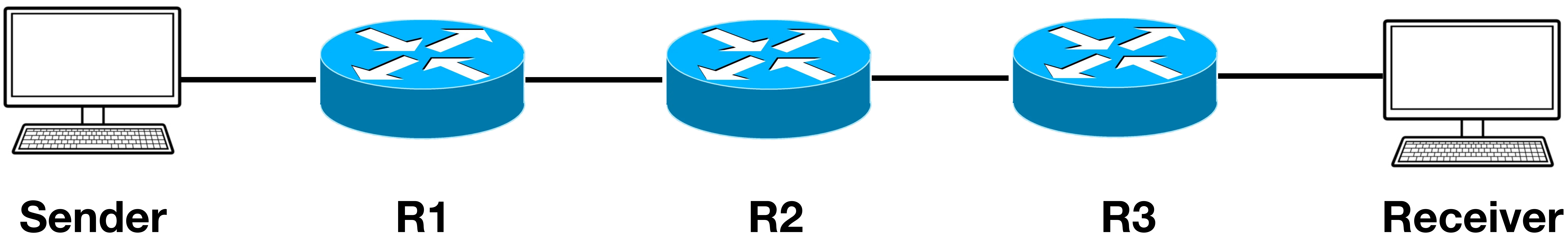
- IP addresses differentiate different communication entities
- IP addresses determine how data is transmitted
- IP addresses impact the network scalability



How do we allocate IP addresses?

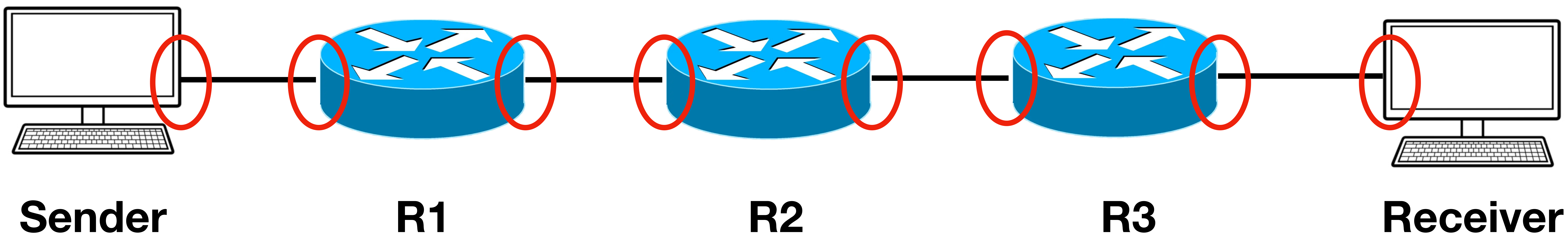
IP Address Owner

- Interface: the communication port connects hosts and links
  - A host might contain one or several interfaces
  - A router encloses multiple interfaces



IP Address Owner

- Interface: the communication port connects hosts and links
  - A host might contain one or several interfaces
  - A router encloses multiple interfaces
- An IP address is owned by an interface
  - A host might contain one or several IP addresses
  - A router encloses multiple IP addresses



IP Address Owner

- Interface: the communication port connects hosts and links
  - A host might contain one or several interfaces
  - A router encloses multiple interfaces
- An IP address is owned by an interface
  - A host might contain one or several IP addresses

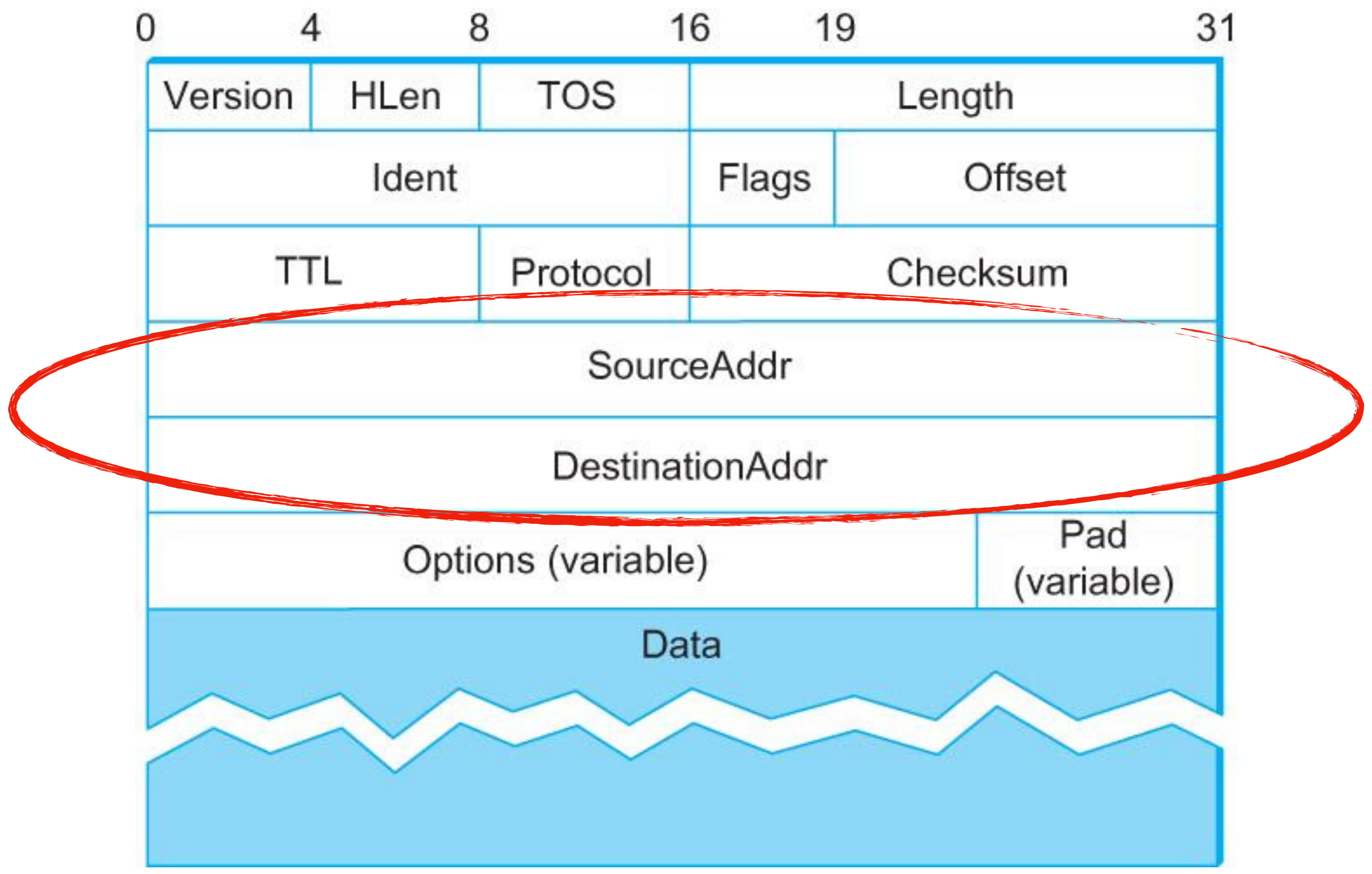
Hosts use IP addresses to differentiate communication entities.
Routers use IP addresses to route communication traffic.

IP address allocation must consider:

- The scale of your network => The number of hosts
- The efficiency of address lookup in the router => Hosts from the same network should have address locality

The Classful Addressing Scheme

- Hierarchical addresses
  - ICANN (Internet Corporation for Assigned Names and Numbers)
- Divide the 32-bit address space into network and host
  - <Network, Host>



Dotted-decimal Notation

- An IPv4 address is 32 bits long (4 bytes) => 2^32 addresses
  - Each byte of the address is written in its decimal form, separated by a period (dot) from other bytes in the address

11000001 . 00100000 . 11011000 . 00001001

Dotted-decimal Notation

- An IPv4 address is 32 bits long (4 bytes) => 2^32 addresses
  - Each byte of the address is written in its decimal form, separated by a period (dot) from other bytes in the address

11000001 . 00100000 . 11011000 . 00001001

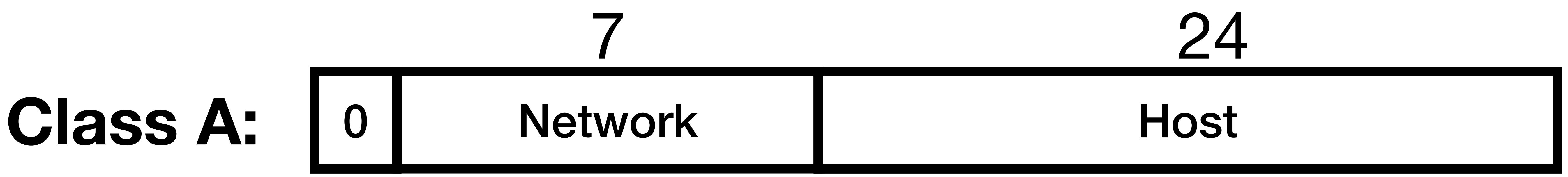
↓

193 . 32 . 216 . 9

How do we partition the 32-bit range to network and hosts?

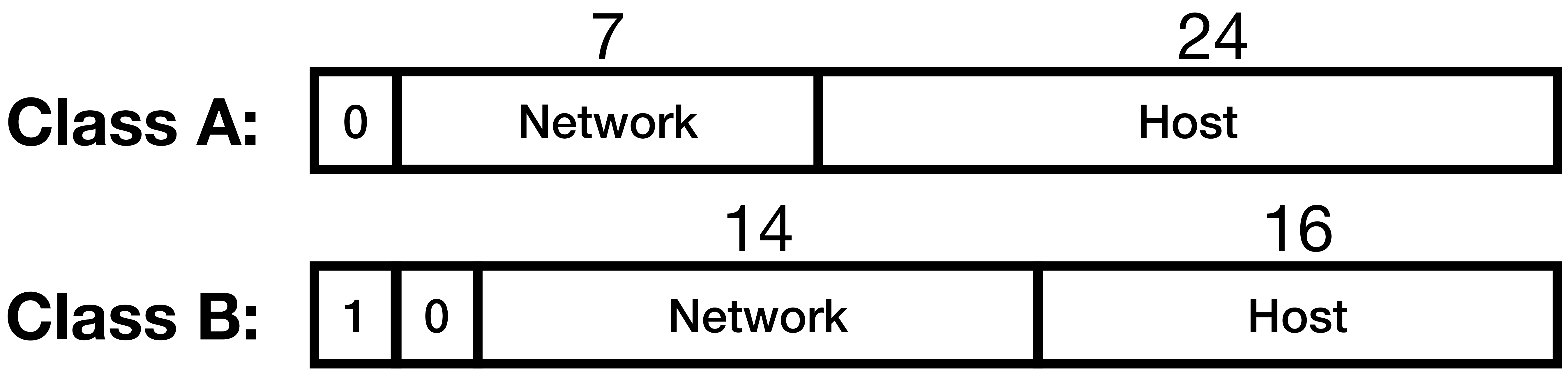
Five IP Classes

- Class A address
  - 1.0.0.0 to 127.0.0.0



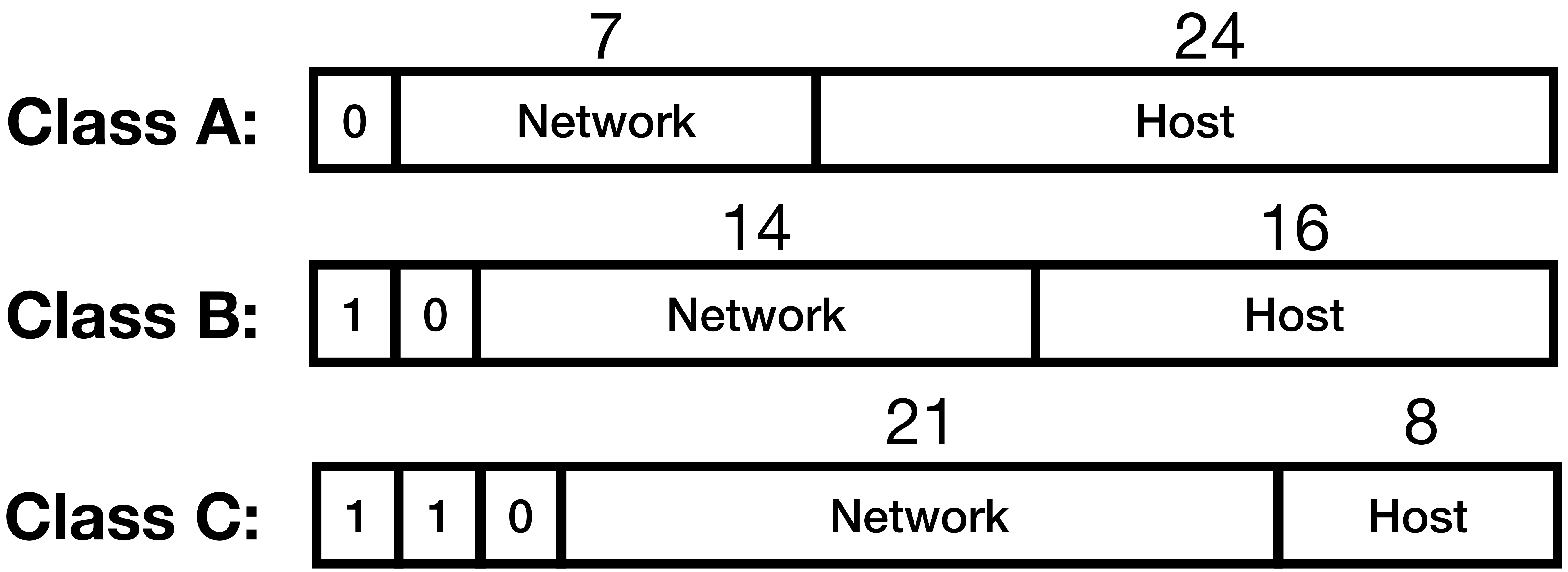
Five IP Classes

- Class B address
  - 128.0.0.0 to 191.255.0.0.0



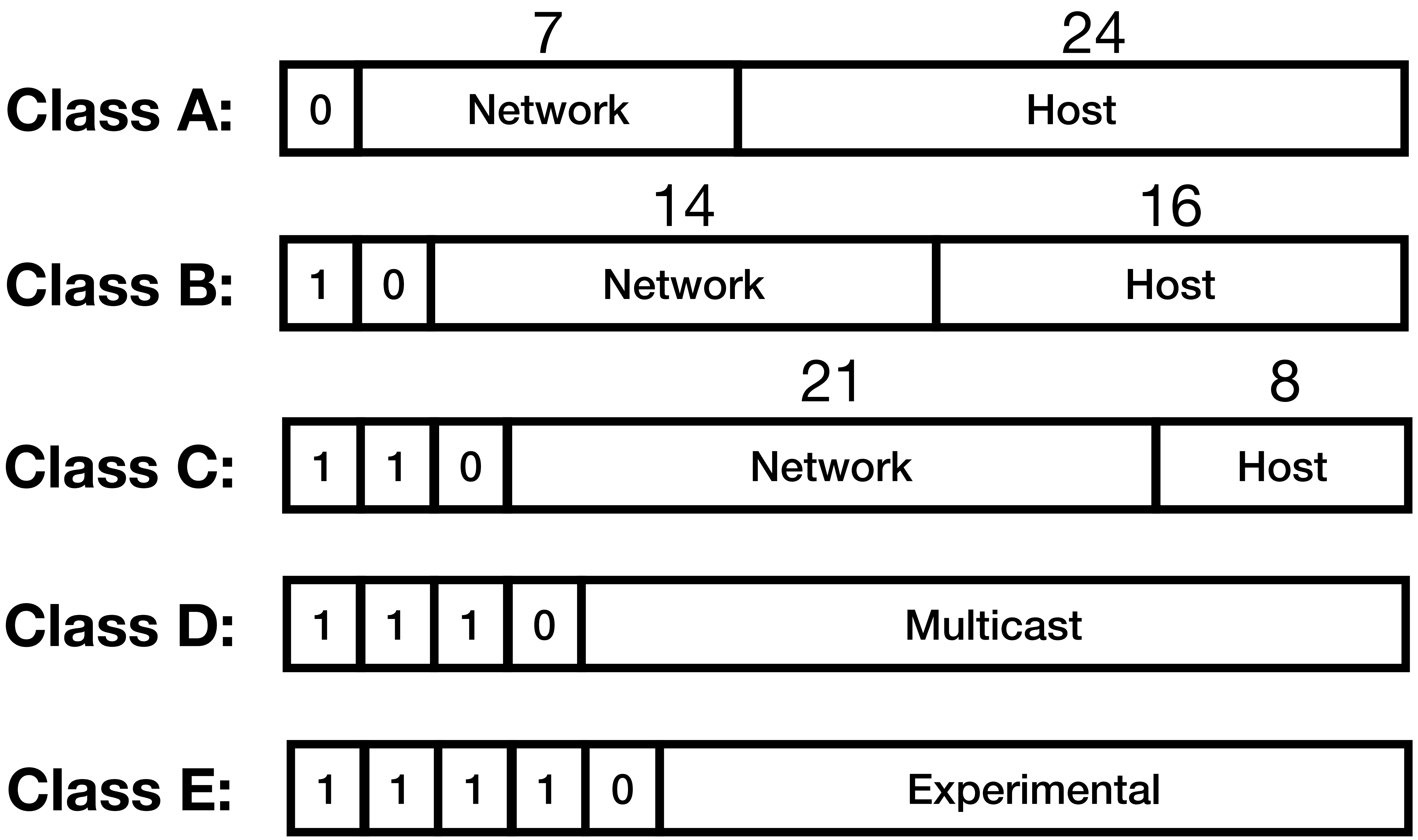
Five IP Classes

- Class C address
  - 192.0.0.0 to 223.255.255.0



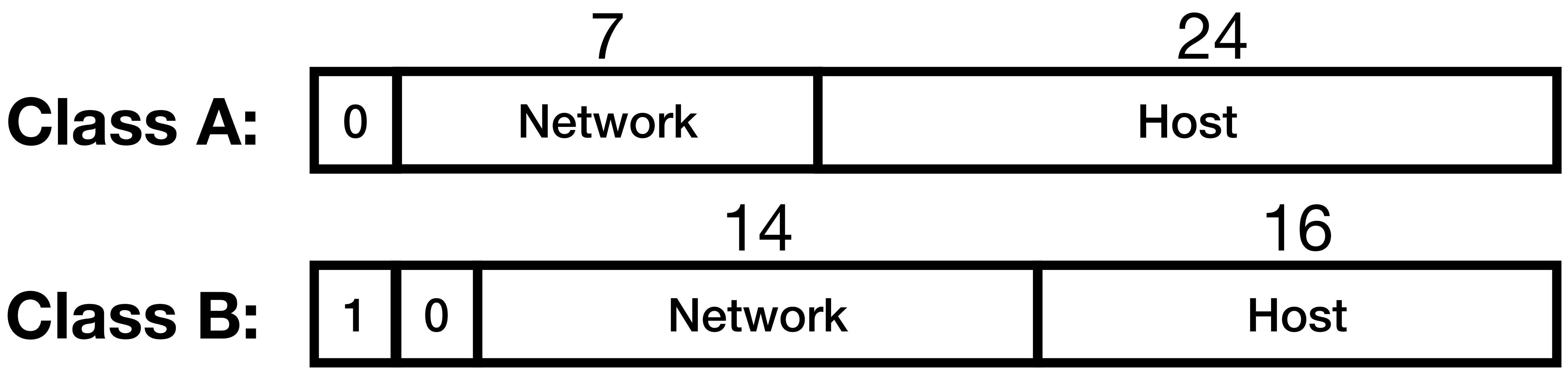
Five IP Classes

- Class D: reserved for multicasting
- Class E: used for experimental research and development



Five IP Classes

- Class D: reserved for multicasting
- Class E: used for experimental research and development



ICANN assigns the network number (or IP address range) for an organization, but not for individual hosts!

As an IT operator, which IP classes should I apply for?

As an IT operator, which IP classes should I apply for?

- The problem becomes increasingly challenging when you consider upgrading

Classful Addressing is Inefficient

- Issue #1: address waste
  - Class C with 2 hosts (2/255 = 0.78% efficiency)
  - Class B with 256 hosts (256/65535 = 0.39% efficiency)

Classful Addressing is Inefficient

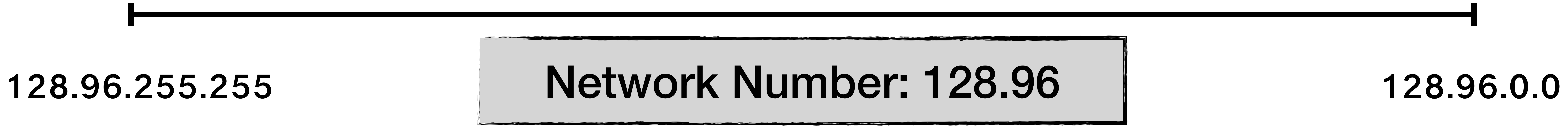
- Issue #1: address waste
  - Class C with 2 hosts (2/255 = 0.78% efficiency)
  - Class B with 256 hosts (256/65535 = 0.39% efficiency)

- Issue #2: still too many networks (discussed later)
  - Routing tables become expensive and cannot scale
  - Route propagation protocols do not scale

How can we improve the address usage?

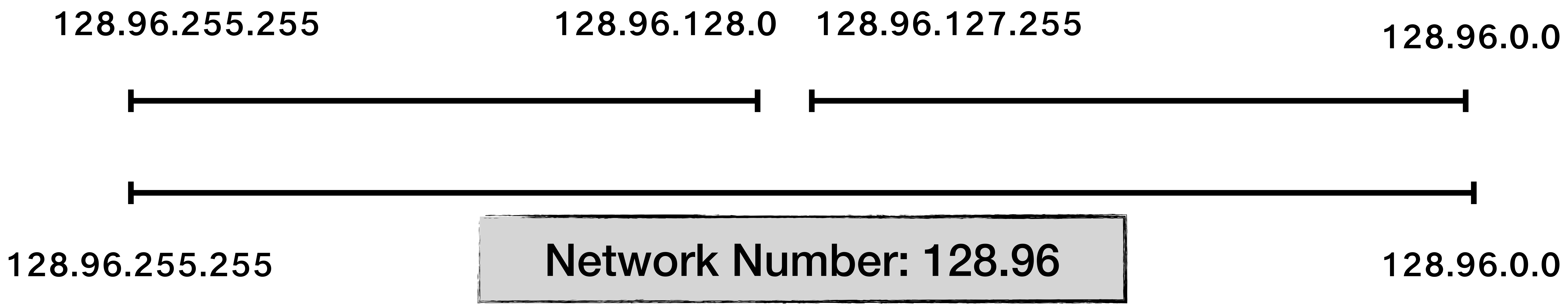
Idea: Flexible Address Partitioning

- Class B address: 128.96.X.X



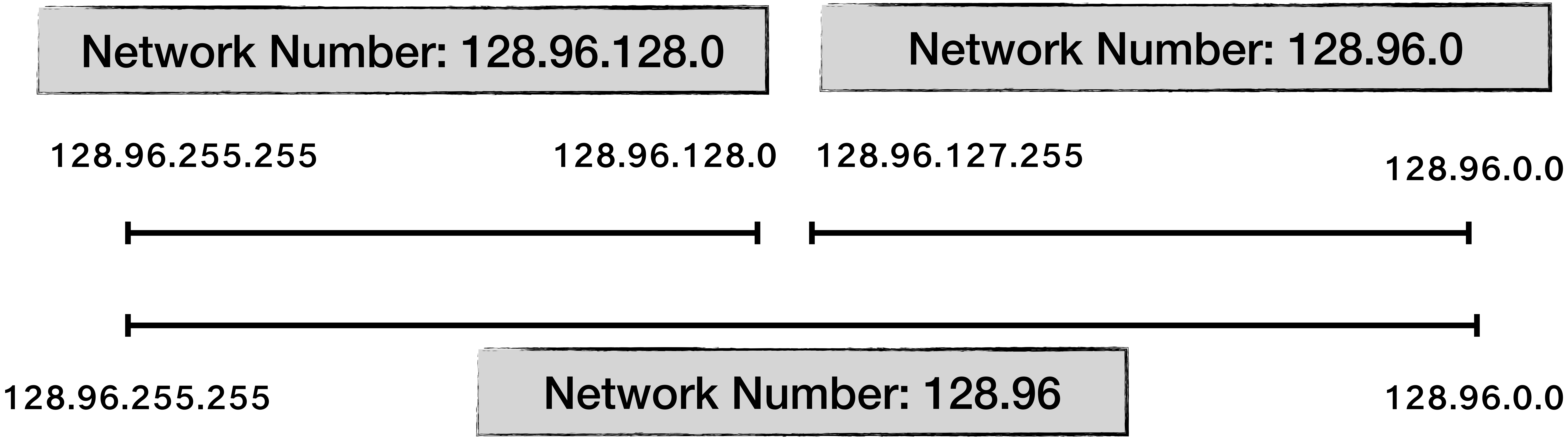
Idea: Flexible Address Partitioning

- Class B address: 128.96.X.X



Idea: Flexible Address Partitioning

- Class B address: 128.96.X.X



#1: Subnetting

- Idea: add another addressing hierarchy (RFC 950)
  - Partition one class address further via masks

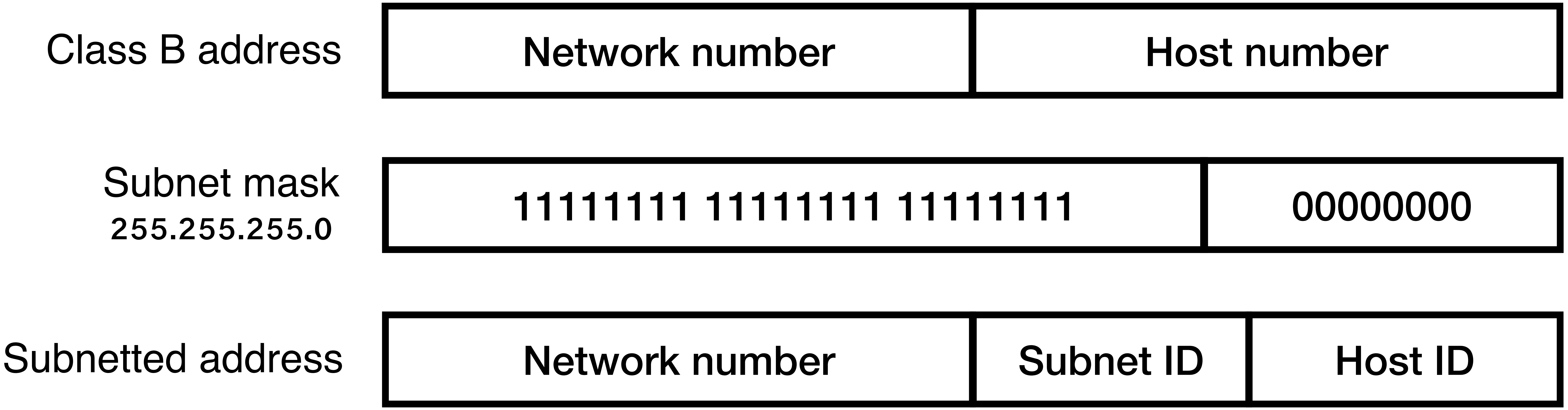
#1: Subnetting

- Idea: add another addressing hierarchy (RFC 950)
  - Partition one class address further via masks

- The subnet mask makes the network number flexible
  - E.g., 255.255.255.0 or /24 => Leftmost bits are 1s, followed by 0s

#1: Subnetting

- Idea: add another addressing hierarchy (RFC 950)
  - Partition one class address further via masks
- The subnet mask makes the network number flexible
  - E.g., 255.255.255.0 or /24 => Leftmost bits are 1s, followed by 0s



A Subnet Example

- Get a class B address: 128.96.X.X

A Subnet Example

- Get a class B address: 128.96.X.X

- Suppose the subnet mask is: 255.255.0.0
  - 1 subnet, each holding 2^16 hosts

A Subnet Example

- Get a class B address: 128.96.X.X

- Suppose the subnet mask is: 255.255.0.0
  - 1 subnet, each holding 2^16 hosts
- Suppose the subnet mask is: 255.255.128.0
  - 2 subnets, each holding 2^15 hosts

A Subnet Example

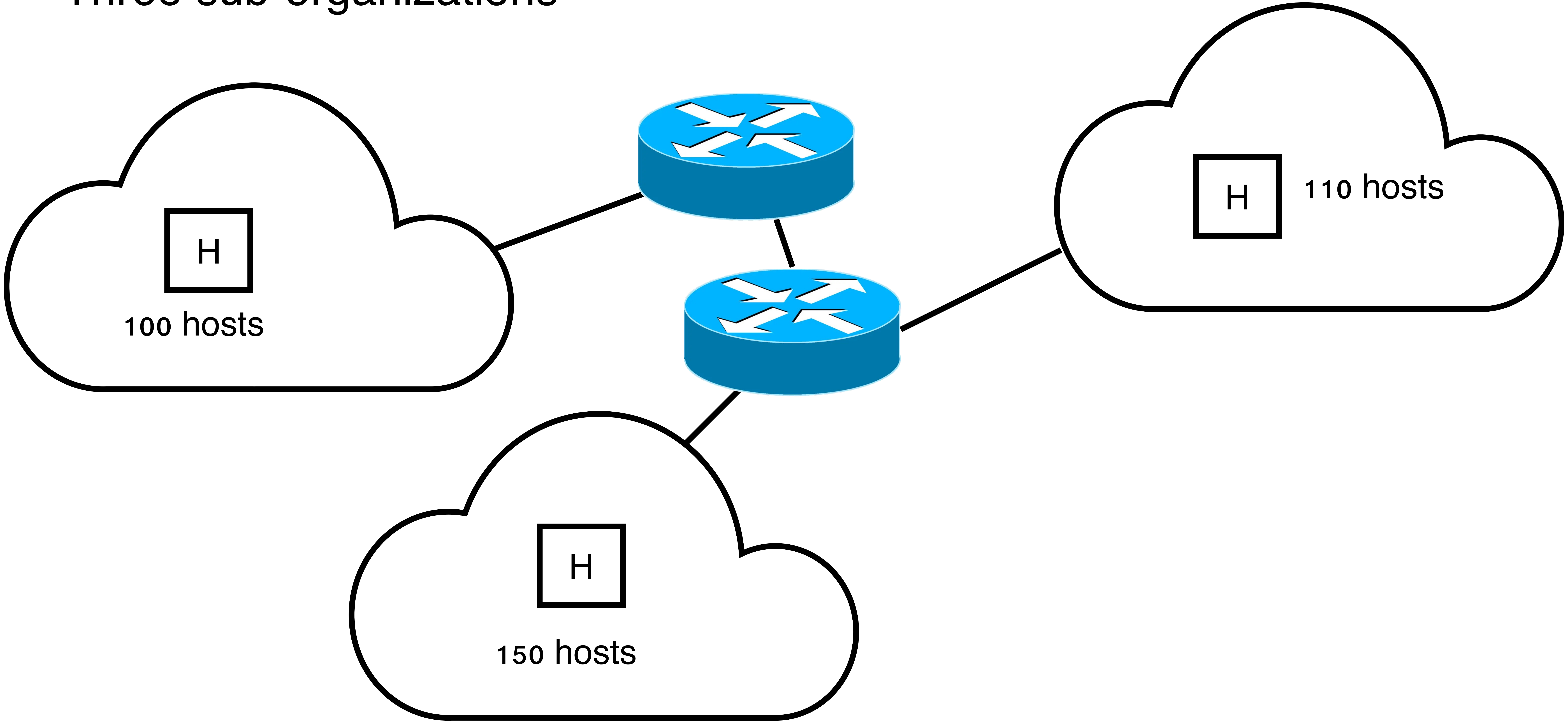
- Get a class B address: 128.96.X.X

- Suppose the subnet mask is: 255.255.0.0
    - 1 subnet, each holding 2^16 hosts
- Suppose the subnet mask is: 255.255.128.0
    - 2 subnets, each holding 2^15 hosts

- Suppose the subnet mask is: 255.255.192.0
    - 4 subnets, each holding 2^14 hosts

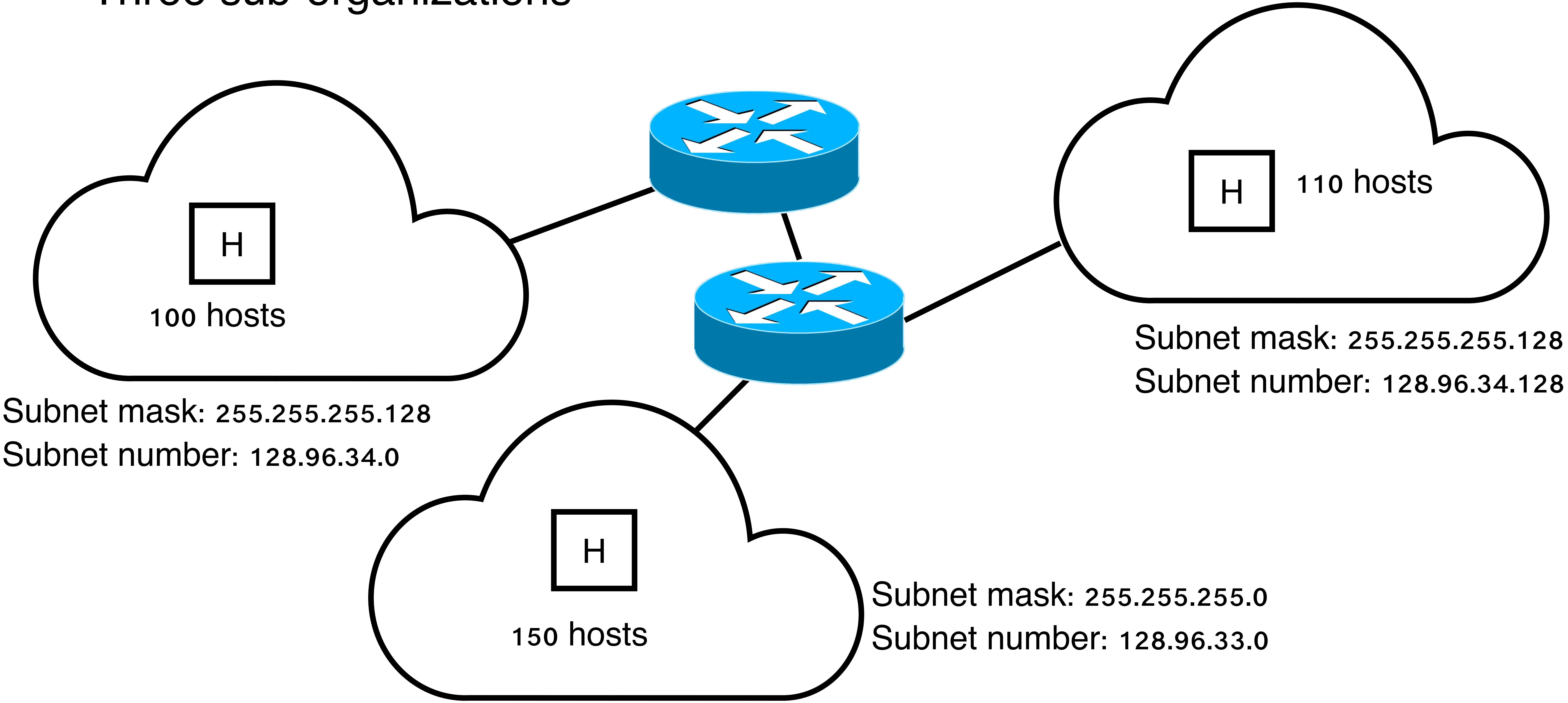
A Subnet Example

- Get a class B address: 128.96.X.X
  - Three sub-organizations



A Subnet Example

- Get a class B address: 128.96.X.X
  - Three sub-organizations



Can We Do Arbitrary Bit Mask Numbers?

- Yes. Chapter 2.2 (RFC 950)
  - It doesn’t care about contiguous bits

```
To support subnets, it is necessary to store one more 32-bit
quantity, called my_ip_mask.  This is a bit-mask with bits set in
the fields corresponding to the IP network number, and additional
bits set corresponding to the subnet number field.

The code then becomes:

   IF bitwise_and(dg.ip_dest, my_ip_mask)
                             = bitwise_and(my_ip_addr, my_ip_mask)
       THEN
           send_dg_locally(dg, dg.ip_dest)
       ELSE
           send_dg_locally(dg,
                  gateway_to(bitwise_and(dg.ip_dest, my_ip_mask)))
```

Can We Do Arbitrary Bit Mask Numbers?

- Yes. Chapter 2.2 (RFC 950)
  - It doesn’t care about contiguous bits

- But not preferred due to computing efficiency
  - AND is more straightforward

- Bit masks: a sequence of N 1 bits followed by a sequence of M 0 bits, where N + M = 32
  - If the subnet number is S (like /24 or slash-24), this means that hosts on the subnet S have IP addresses whose first N bits are 1

Is subnetting enough to solve the address efficiency issue?

Partially Solved, But

- Class B network numbers are highly costly (not everyone needs)
  - Lots of class C addresses are not being used

- The backbone routing tables grow significantly
  - Lots of small networks
  - Route calculation and management require high computing overheads

#2: Supernetting

- Idea: Enable network number to be any length
  - Route aggregation
- Collapse multiple addresses assigned to a single entity
  - Assign a block of continuous network numbers to nearby networks

The CIDR Address Assignement

- CIDR = Classless Interdomain Routing
  - Breaks rigid boundaries between address classes
  - Combine subnetting and supernetting

- An organization can apply class C addresses and merge
  - E.g., 192.4.16.xx to 192.4.31.xx enables a 20-bit network number

CIDR Address

- A CIDR address block: a.b.c.d/x
  - The x most significant bits of an address constitute the network portion
  - Referred to as the prefix
  - 128.211.168.0/21 => 128.211.168.0 — 182.211.175.255

- All possible CIDR masks can be generated
  - /8, /16, /24 refer to traditional class A, B, and C class addresses

- Broadcast address of a CIDR address block
  - All host bits are “1”

An Exercise

A tech startup is assigned the private block 192.168.10.0/24. They need to divide this address space for four distinct departments with different host requirements. Allocate the subnets starting from the largest requirement to the smallest.

- Engineering: 100 hosts
- Marketing: 50 hosts
- Sales: 25 hosts
- Support: 12 hosts

An Exercise

A tech startup is assigned the private block 192.168.10.0/24. They need to divide this address space for four distinct departments with different host requirements. Allocate the subnets starting from the largest requirement to the smallest.

- Engineering: 100 hosts
- Marketing: 50 hosts
- Sales: 25 hosts
- Support: 12 hosts

Solution:

- Engineering (100 hosts): Needs a /25 (128 addresses)
  - Subnet: 192.168.10.0/25
  - Range: .1 to .126 (Broadcast: .127)
- Marketing (50 hosts): Needs a /26 (64 addresses)
  - Subnet: 192.168.10.128/26
  - Range: .129 to .190 (Broadcast: .191)
- Sales (25 hosts): Needs a /27 (32 addresses)
  - Subnet: 192.168.10.192/27
  - Range: .193 to .222 (Broadcast: .223)
- Support (12 hosts): Needs a /28 (16 addresses)
  - Subnet: 192.168.10.224/28
  - Range: .225 to .238 (Broadcast: .239)

How do we assign IP addresses to hosts?

Technique #1: Manual Configure

- Manually choose an IP address based on the network
  - Static allocation
  - `ip a add 192.168.1.100/255.255.255.0 dev eth0`

Technique #1: Manual Configure

- Manually choose an IP address based on the network
  - Static allocation
  - `ip a add 192.168.1.100/255.255.255.0 dev eth0`

- Drawbacks:
  - Lots of work to configure all the hosts in a large network
  - Easy to make mistakes

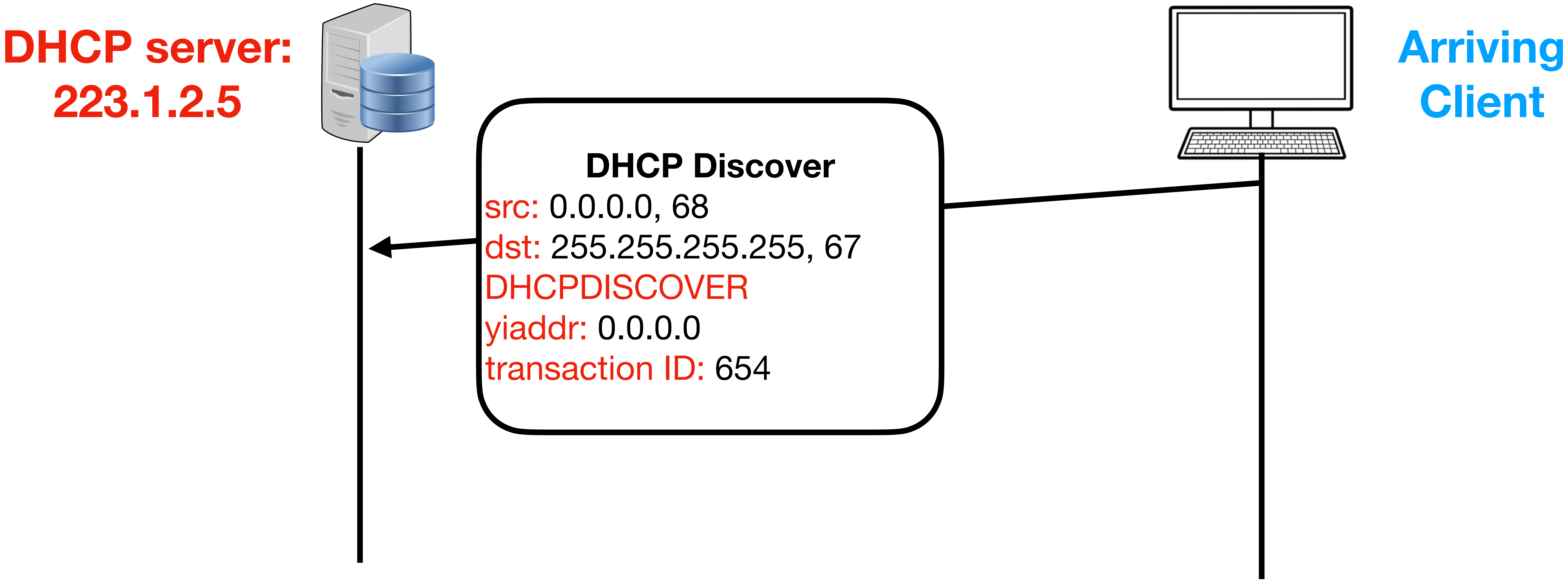
Technique #2: DHCP

- Dynamic Host Configuration Protocol
  - A dedicated service for assigning IP for each administrative domain

- A DHCP server maintains a pool of available addresses
  - Maintain the mapping between hosts and IP addresses
  - Each IP is associated with a lease to ensure flexibility
  - Leases are periodically refreshed

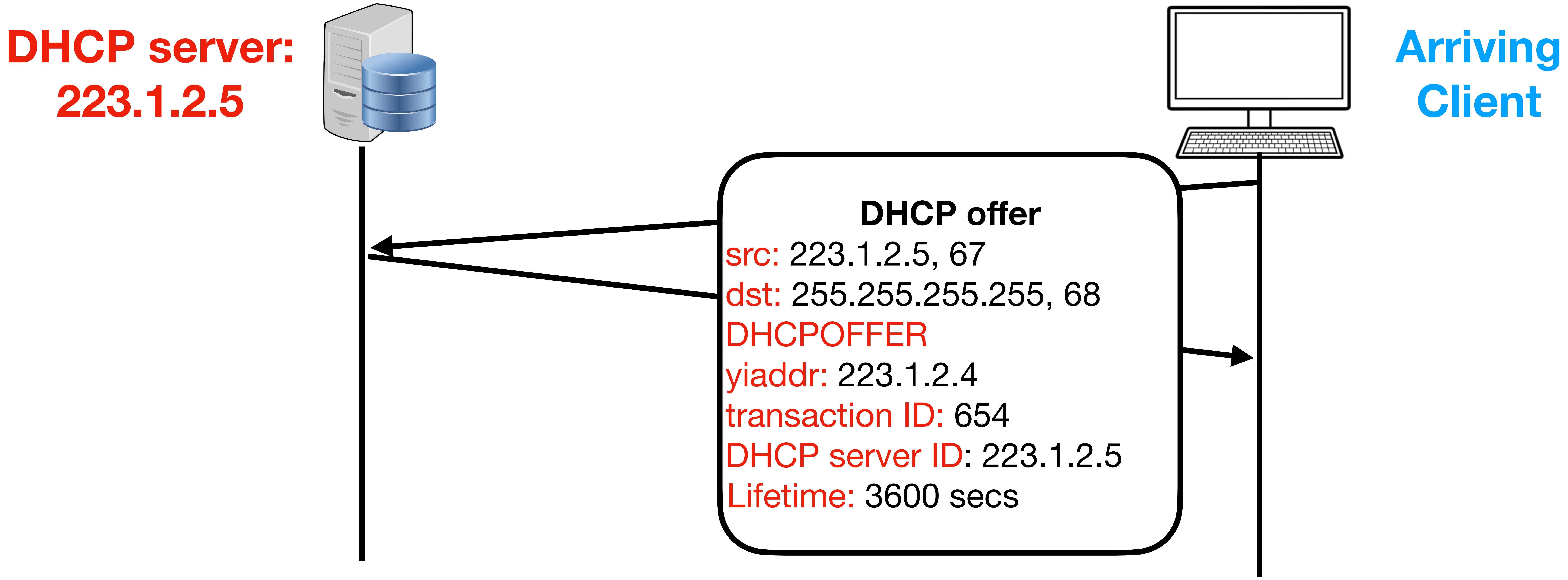
How DHCP Works

- #1: DHCP Discover
  - An arriving client sends a special IP broadcast message to the network



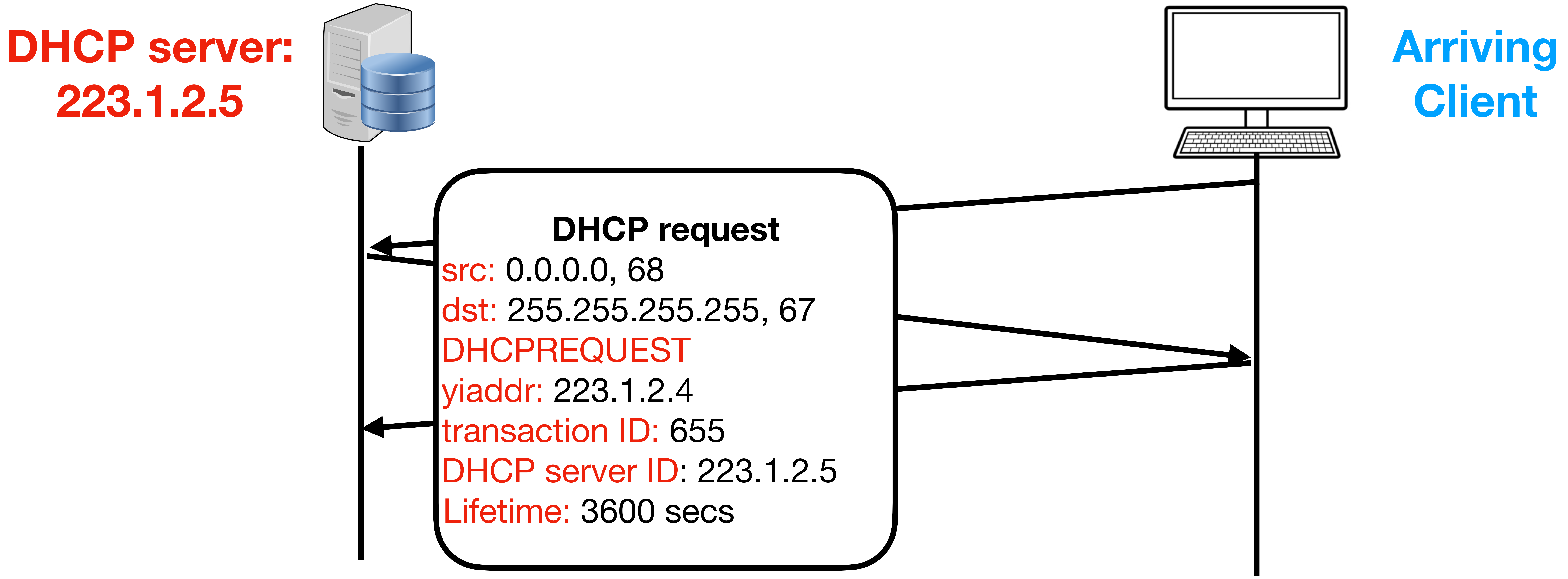
How DHCP Works

- #2: DHCP offer
  - A client can receive multiple offers



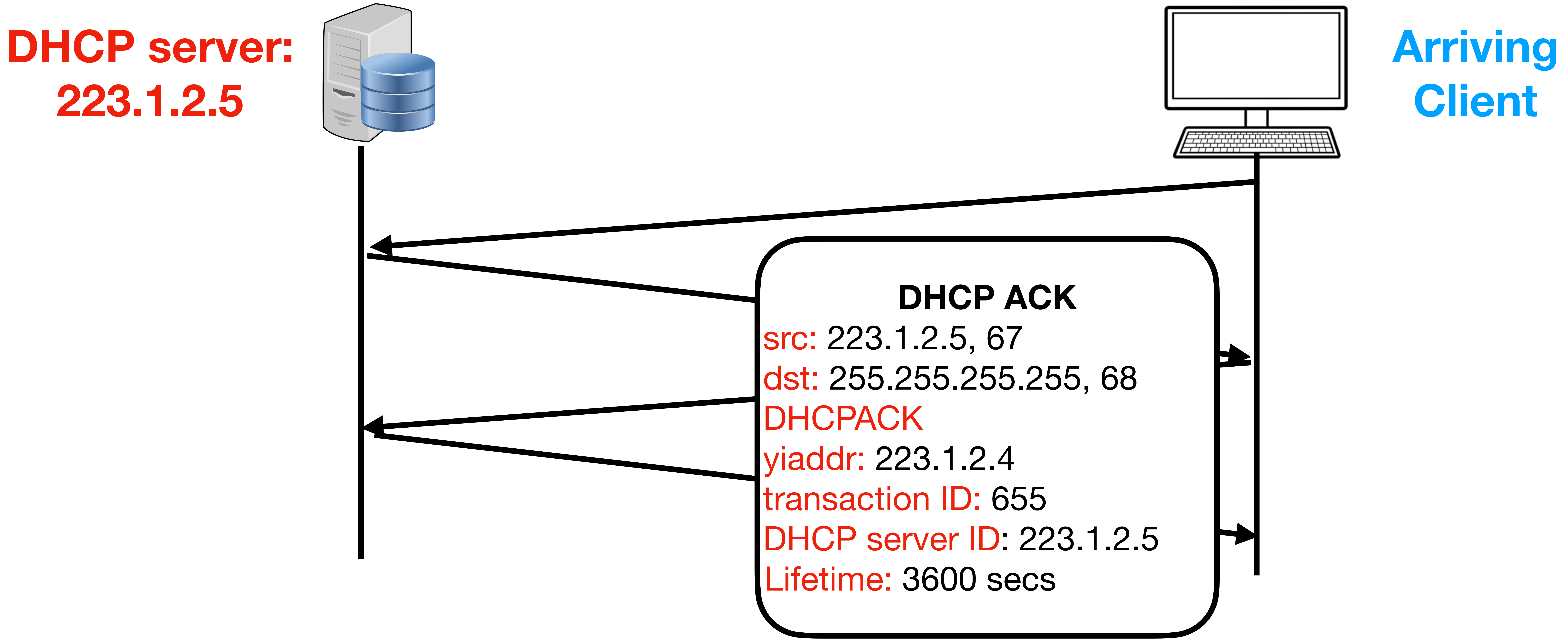
How DHCP Works

- #3: DHCP request
  - The client makes the decision



How DHCP Works

- #4: DHCP ACK
  - The server confirms



Summary

- Today
  - Efficient Addressing

- Next lecture
  - Distance vector routing