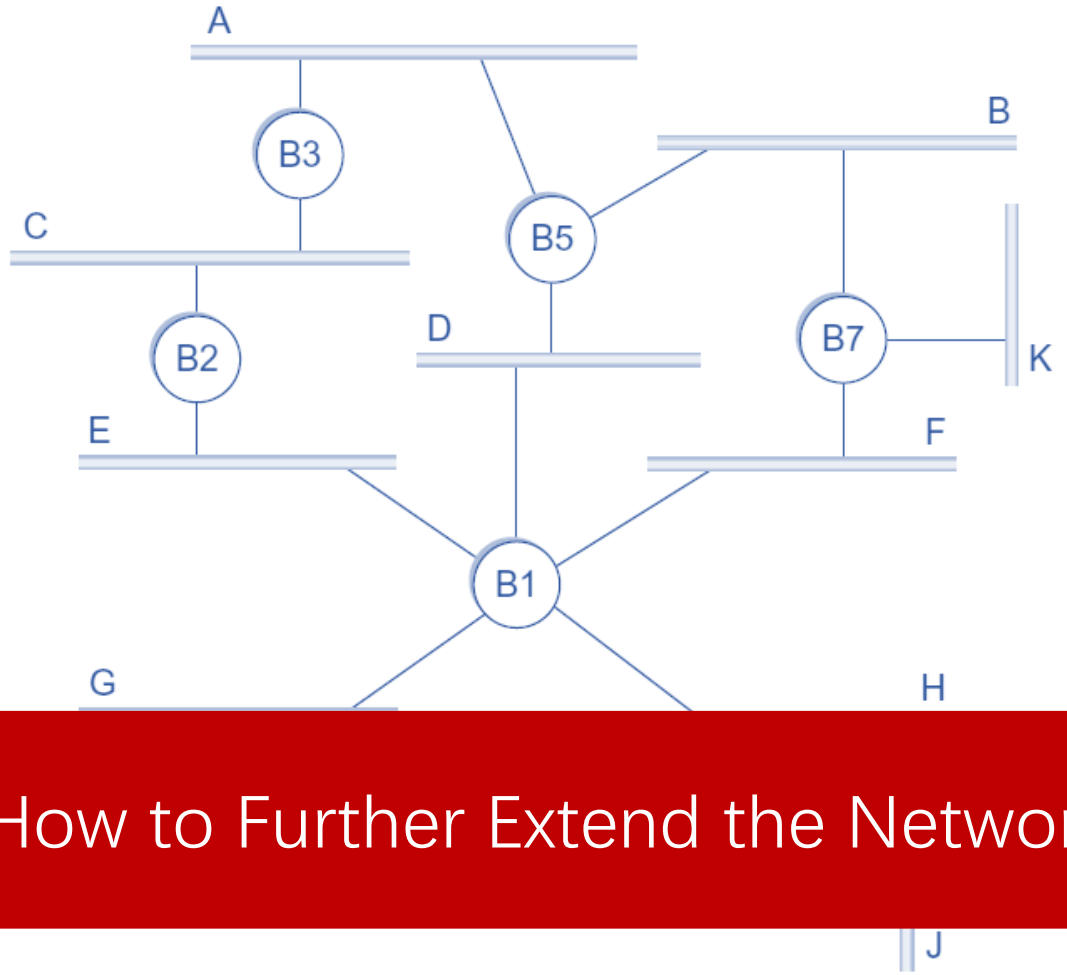
CS120: Computer Networks

Lecture 9. Internet Protocol

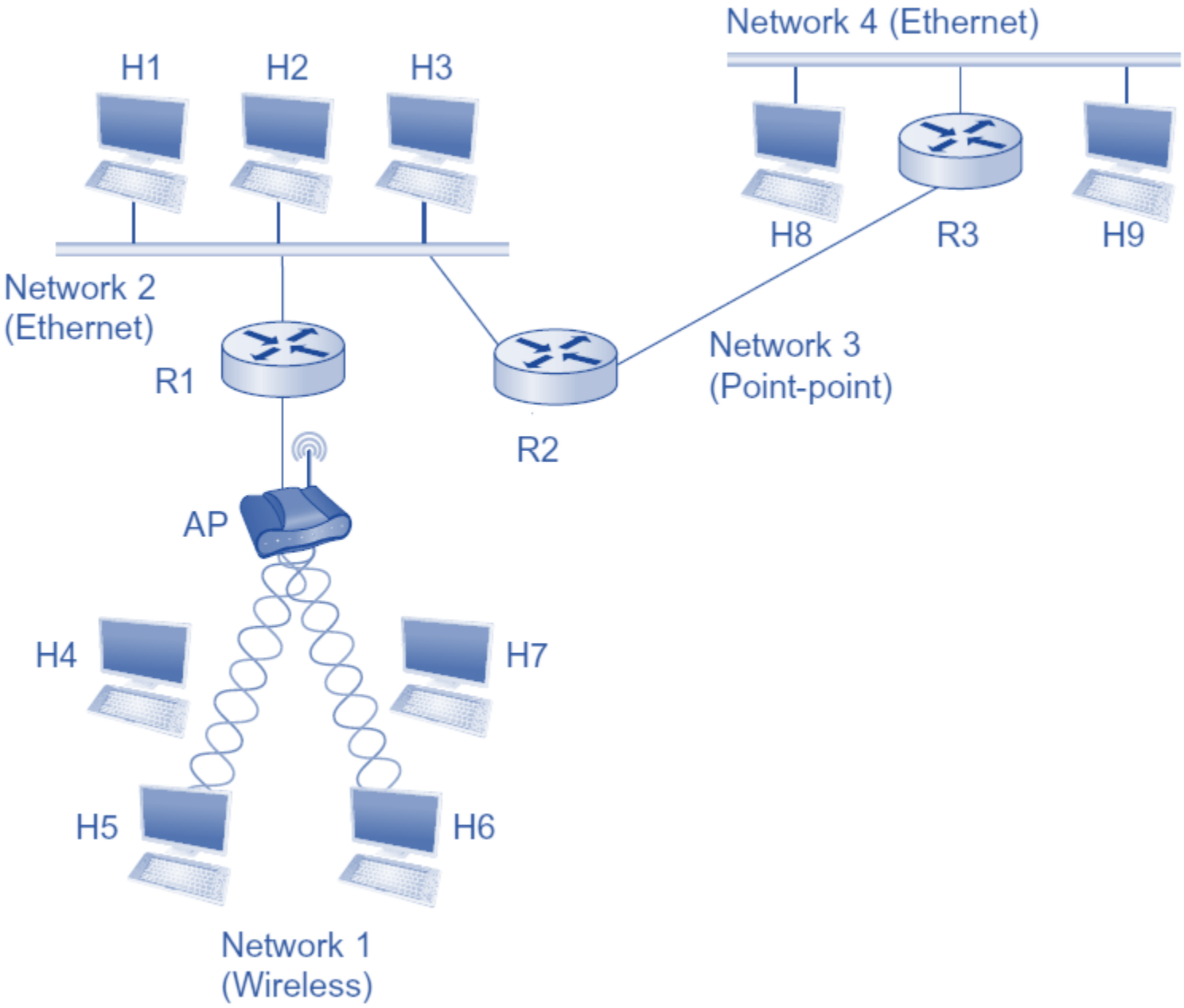
Zhice Yang



How to Further Extend the Network ?

Limitation of Extended Ethernet

- Addressing Scalability
  - Spanning Tree does not scale
    - Large network
      - Switches store too many forwarding entries
      - Huge broadcasting overhead
- Network Heterogeneity
  - Cannot communicate with other networks
    - Cannot addressing nodes in other networks


Network 4 (Ethernet)
H1
H2
H3
H8
R3
H9
Network 2
(Ethernet)
R1
R2
Network 3
(Point-point)
AP
H4
H7
H5
H6
Network 1
(Wireless)

Internet Protocol (IP)

- Goal:
  - Scalable Addressing Scheme
  - Support Heterogeneous Networks
- Service Model: Datagram (Connectionless)
  - Packets can be lost
  - Packets can be delivered out of order
  - Duplicate copies of a packet can be delivered
  - Packets can be delayed for a long time

Outline

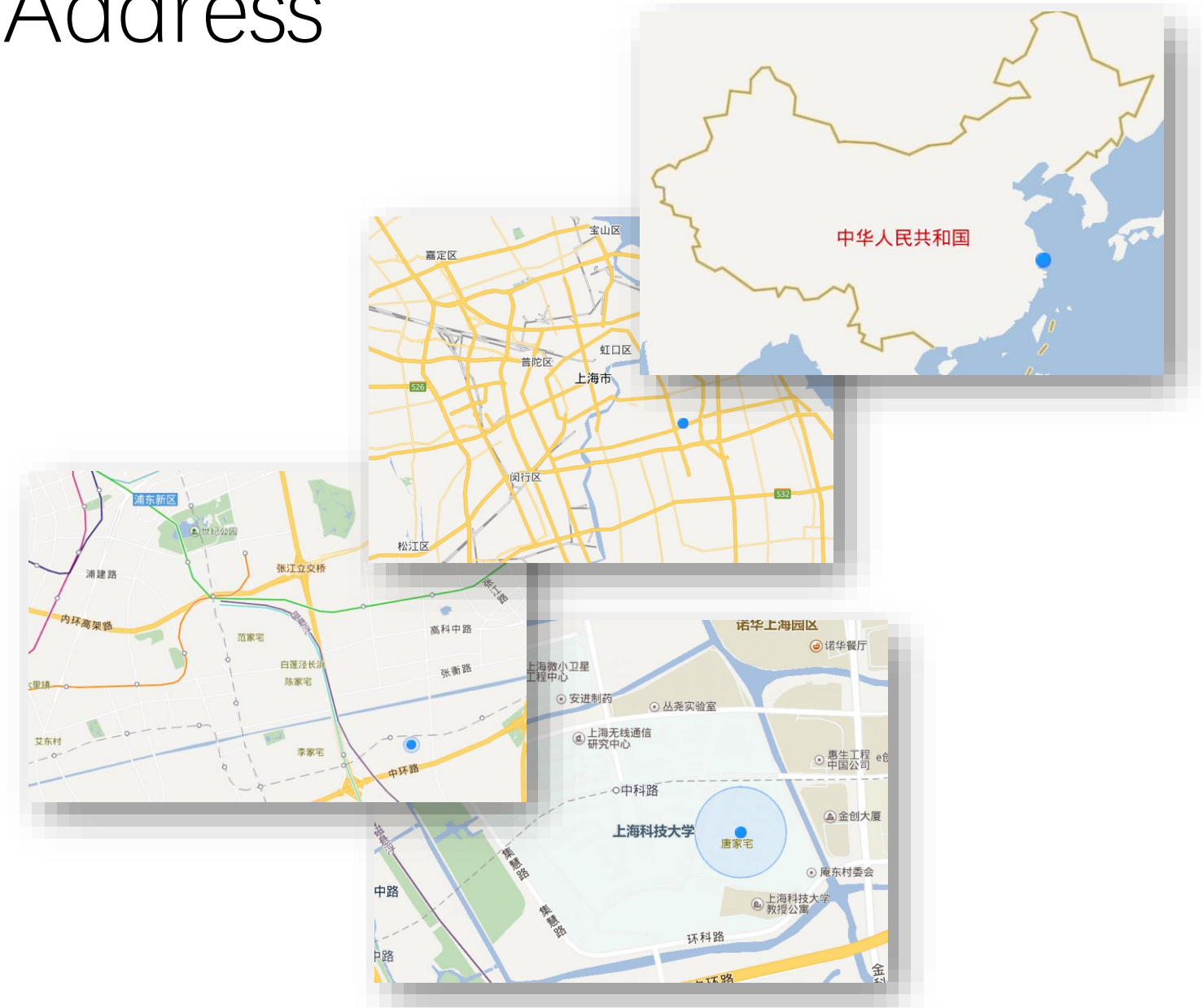
- IP Addressing
  - IP Address
  - Subnet
  - Routing Aggregation
  - IP Distribution: DHCP
  - IP and Switching: ARP
- IP Packet
  - Fragmentation

Addressing in Postal Service

- NAME => Ethernet MAC Address
  - Unique
  - but less informative in finding route to deliver
- In practice we use: Location Address + NAME

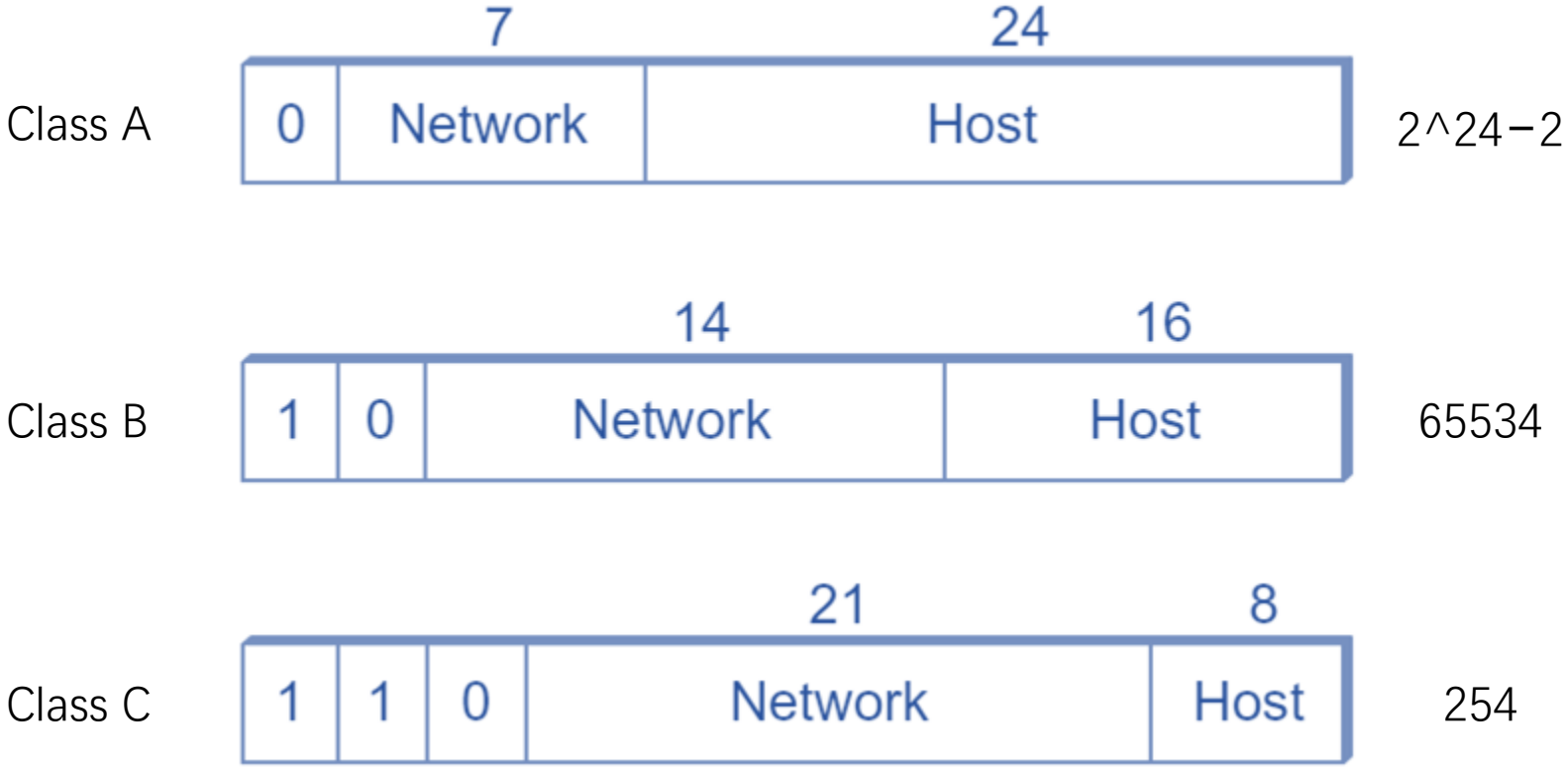
Hierarchical Address

- China
- Shanghai
- Pudong
- ShanghaiTech



IP Address

- IP Address: 32-bit identifier for host, router ports
  - Globally unique (original goal)
  - Hierarchical: network + host (original goal)



IP Address

- Dot notation
  - 10.3.2.4
  - 128.96.33.15
  - 192.12.69.77

10000000 01100000 00100010 00001111

128. 96. 33. 15

Assigning IP Address

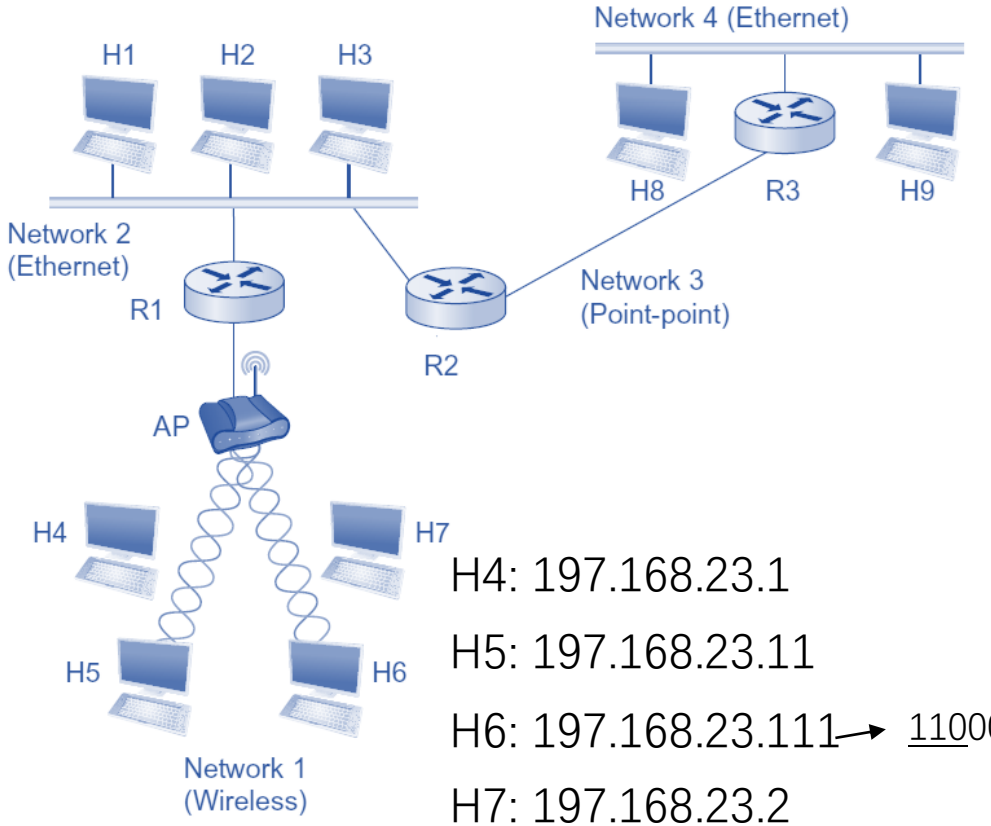
- Each host has a unique IP address
- Hosts in the same physical network have the same network part

H1: 200.155.11.5

H2: 200.155.11.3

H3: 200.155.11.2

11001000.10011011.00001011.XXXXXXXX



H8: 210.168.1.10

H9: 210.168.1.200

11010010.10101000.00000001.XXXXXXXX

H4: 197.168.23.1

H5: 197.168.23.11

H6: 197.168.23.111 → 11000101.10101000.00010111.XXXXXXXX

H7: 197.168.23.2

Assigning IP Address

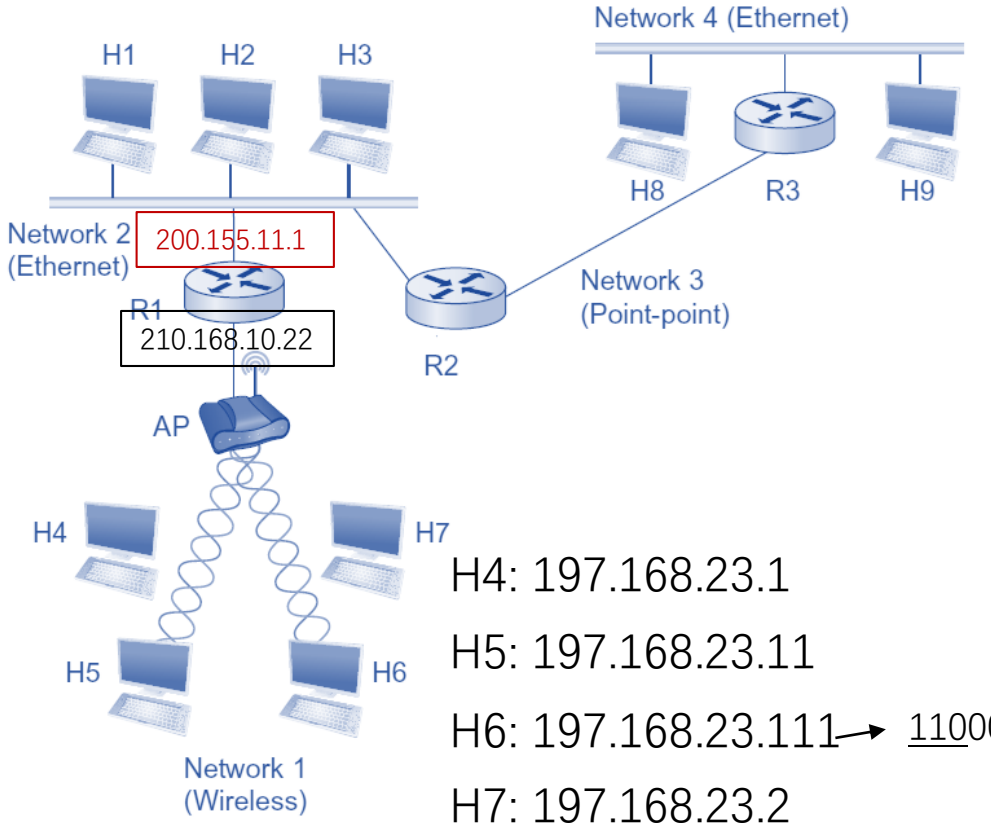
- Each router contains multiple network interfaces
- Each interface has the IP address of the connected network

H1: 200.155.11.5

H2: 200.155.11.3

H3: 200.155.11.2

11001000.10011011.00001011.XXXXXXXX



H8: 210.168.1.10

H9: 210.168.1.200

11010010.10101000.00000001.XXXXXXXX

H4: 197.168.23.1

H5: 197.168.23.11

H6: 197.168.23.111 → 11000101.10101000.00010111.XXXXXXXX

H7: 197.168.23.2

Forwarding with IP Address

- Each router maintains a forwarding table
  - if IP.network == Connected network
    - forward to the host (How? ARP: IP->MAC)
  - if IP.network != Connected network
    - forward to some router

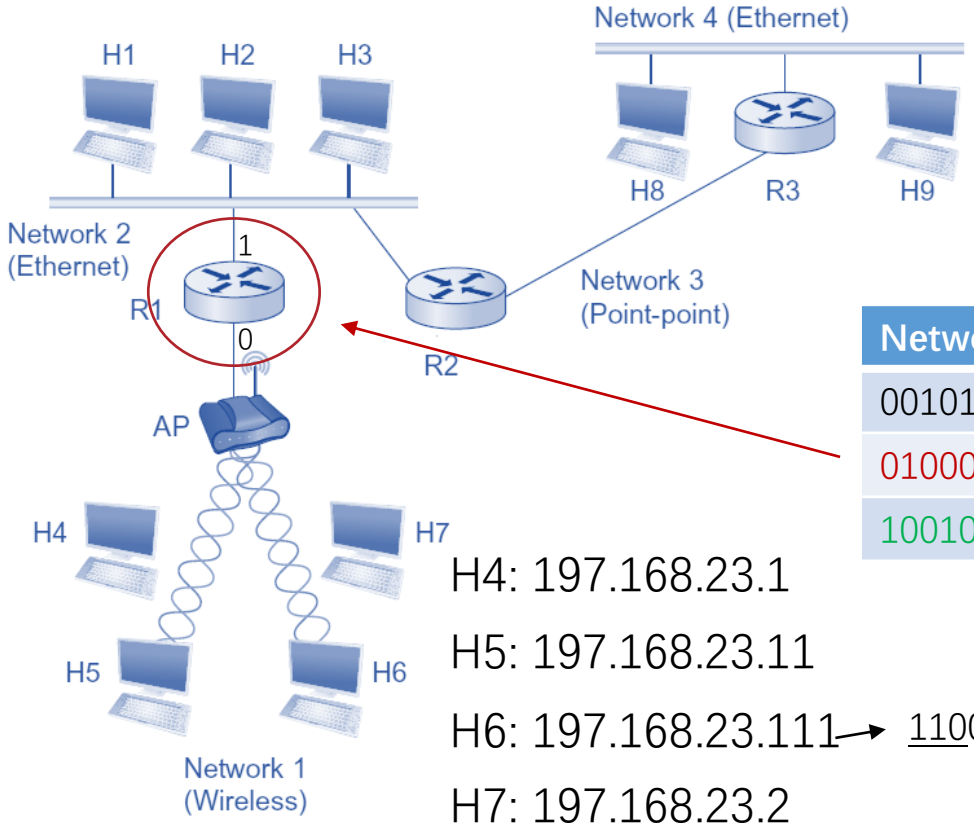
NO need to check the host part

H1: 200.155.11.5

H2: 200.155.11.3

H3: 200.155.11.2

11001000.10011011.00001011.XXXXXXXX



H8: 210.168.1.10

H9: 210.168.1.200

11010010.10101000.00000001.XXXXXXXX

NetworkNum	NextHop
00101.10101000.00010111	Interface 0
01000.10011011.00001011	Interface 1
10010.10101000.00000001	R2

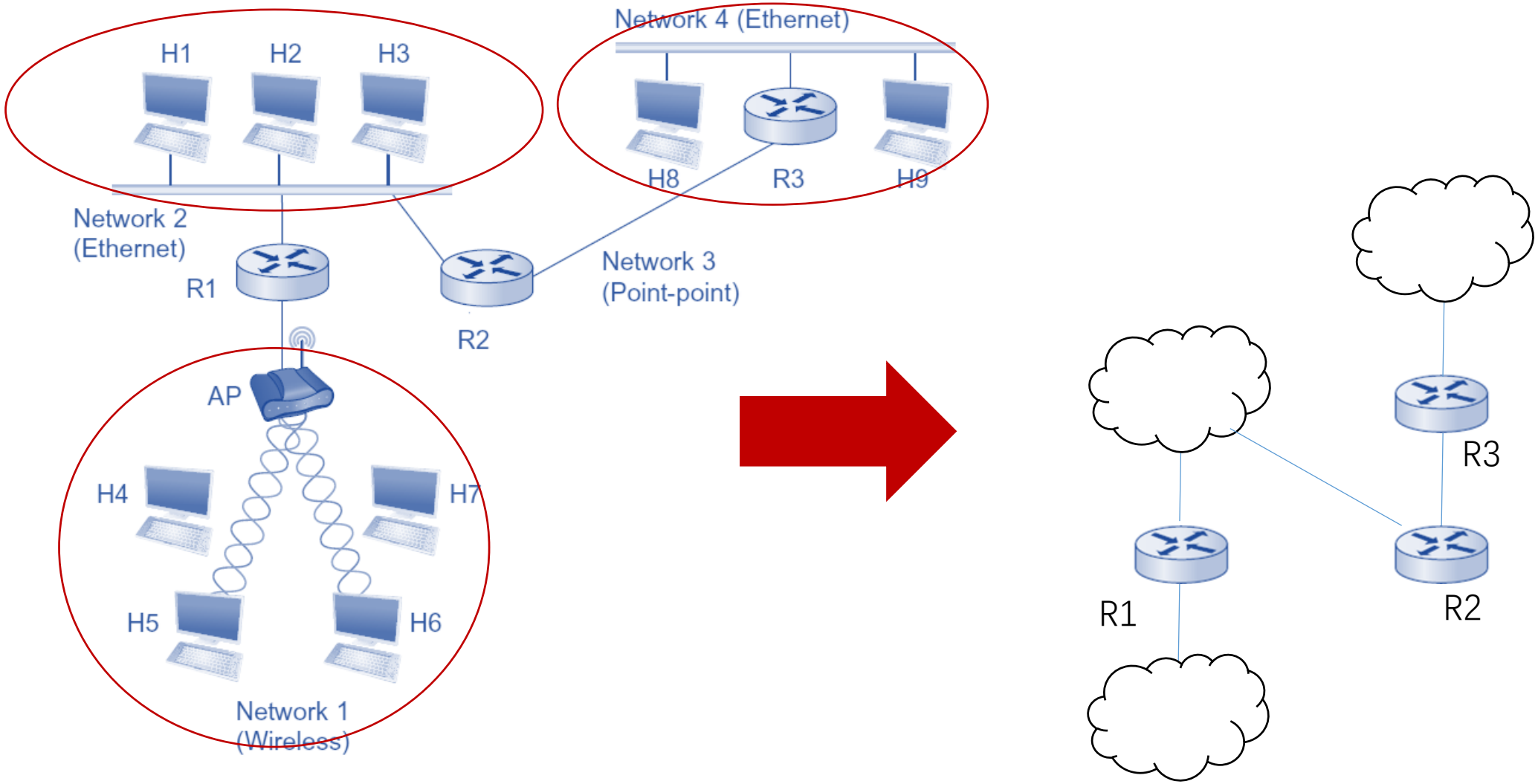
H4: 197.168.23.1

H5: 197.168.23.11

H6: 197.168.23.111 → 11000101.10101000.00010111.XXXXXXXX

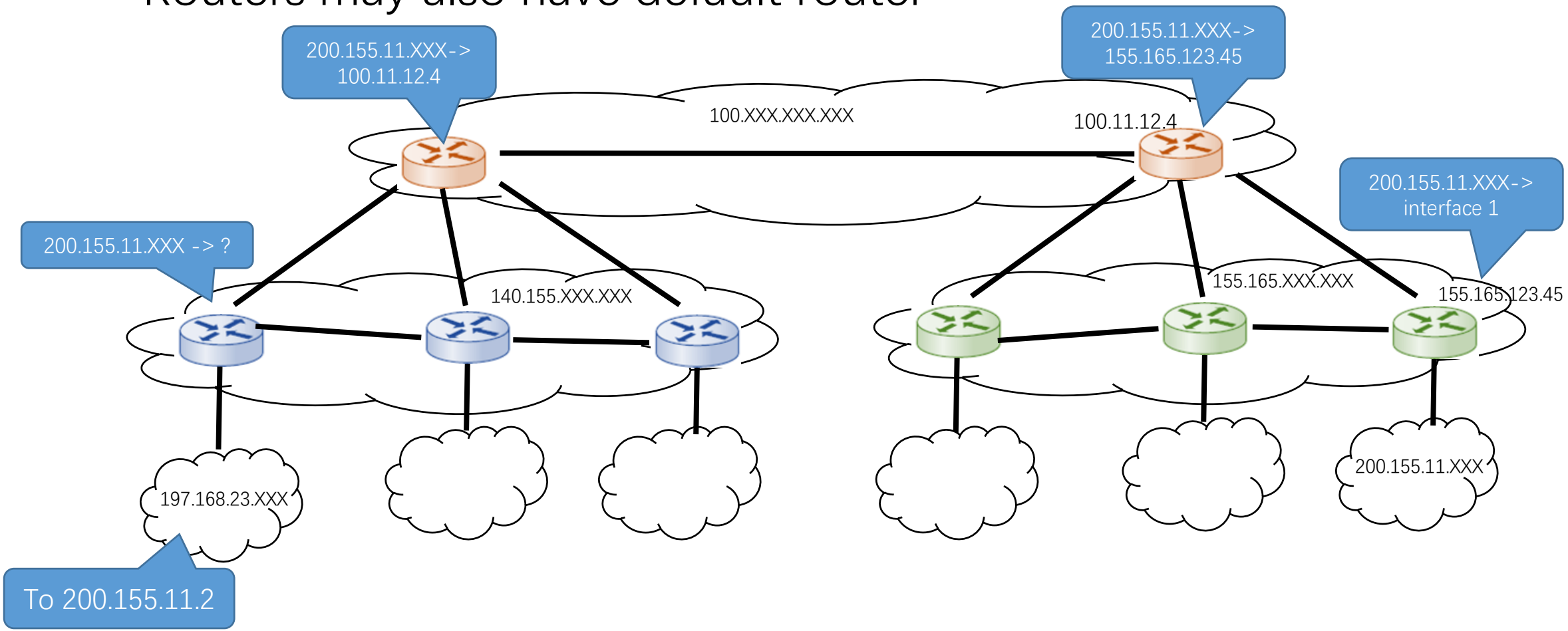
H7: 197.168.23.2

Forwarding with IP Address



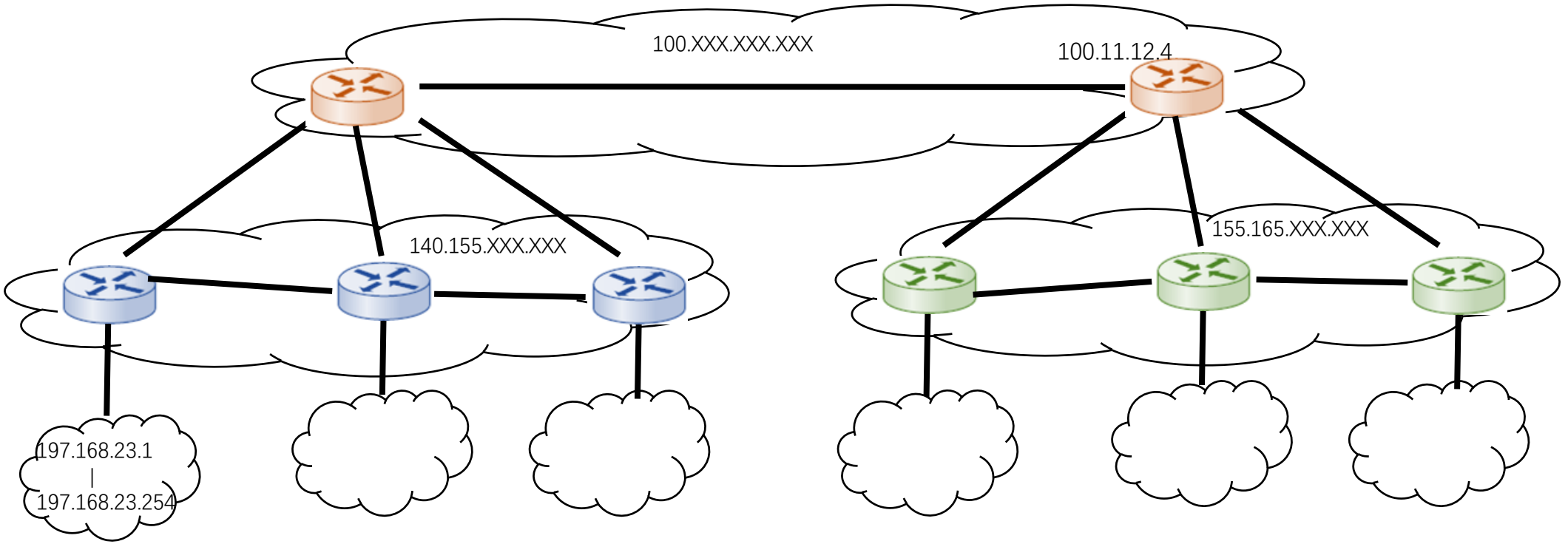
Forwarding with IP Address

- Each host has a default router
- Routers may also have default router

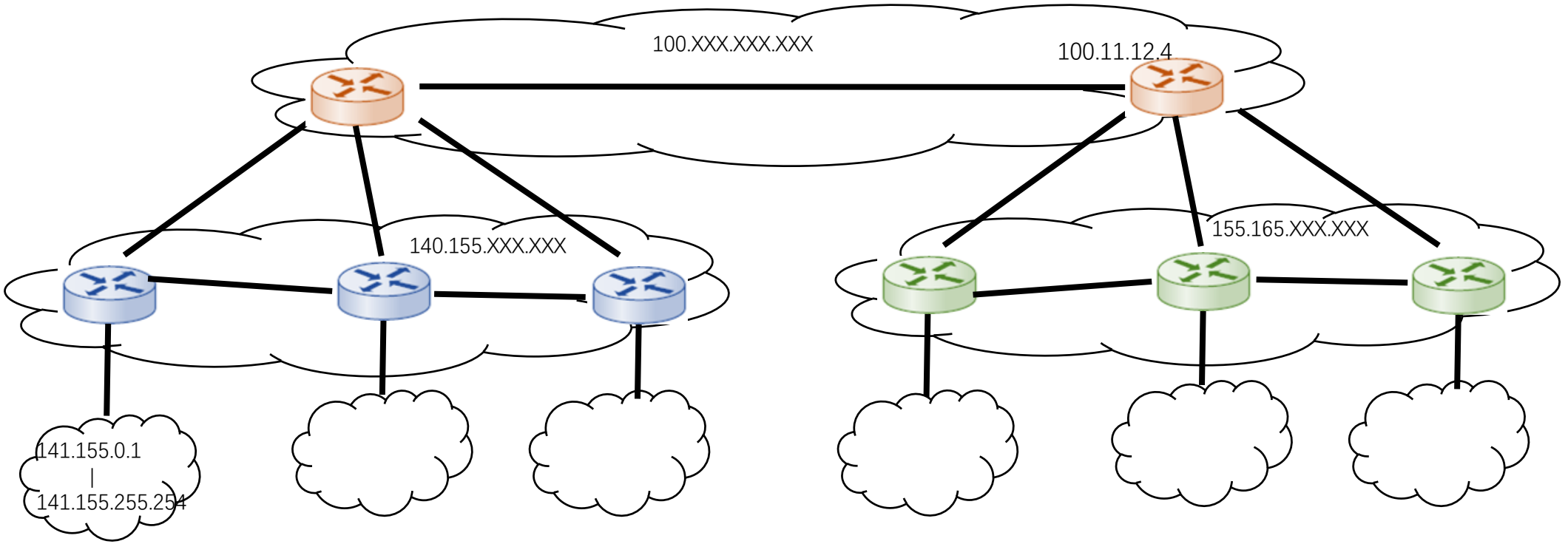


Class Addressing

- Limitation
  - Address utilization is not efficient
    - 255 hosts
      - Class C: not enough
      - Class B: too many addresses are wasted
  - Forwarding table is still large
    - Proportional to the number of networks



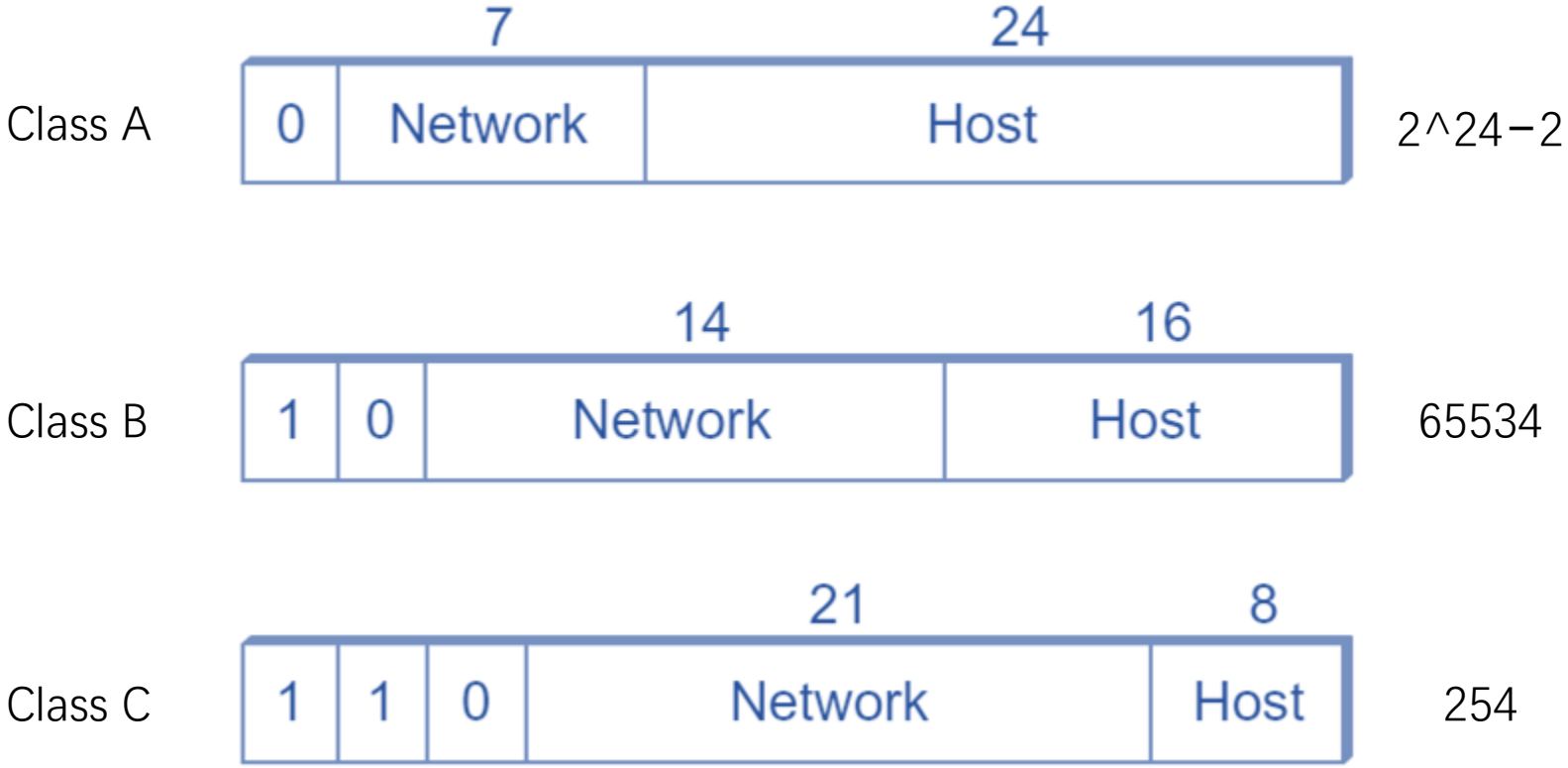
255 hosts
Not Enough!



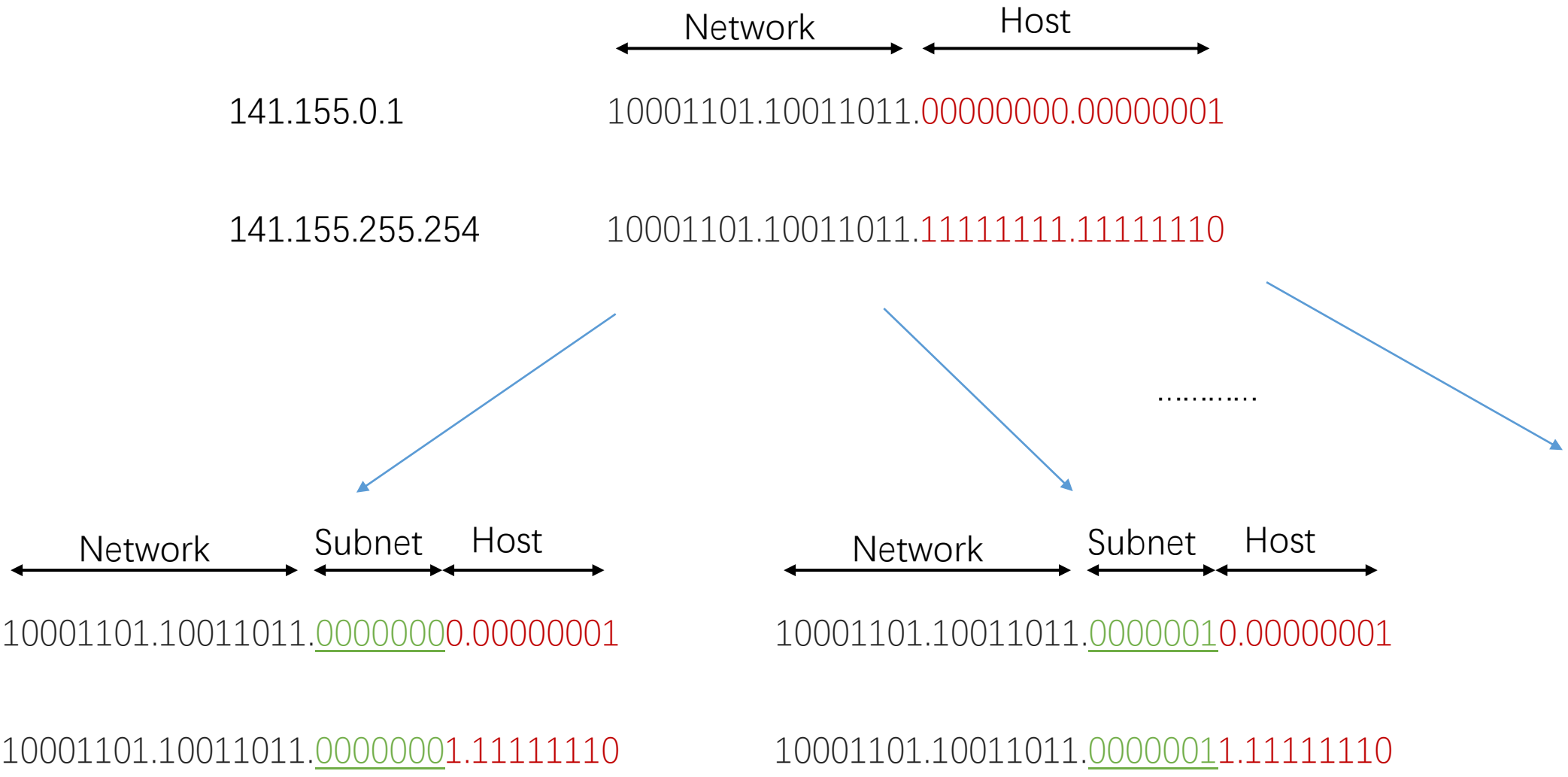
Not Efficient

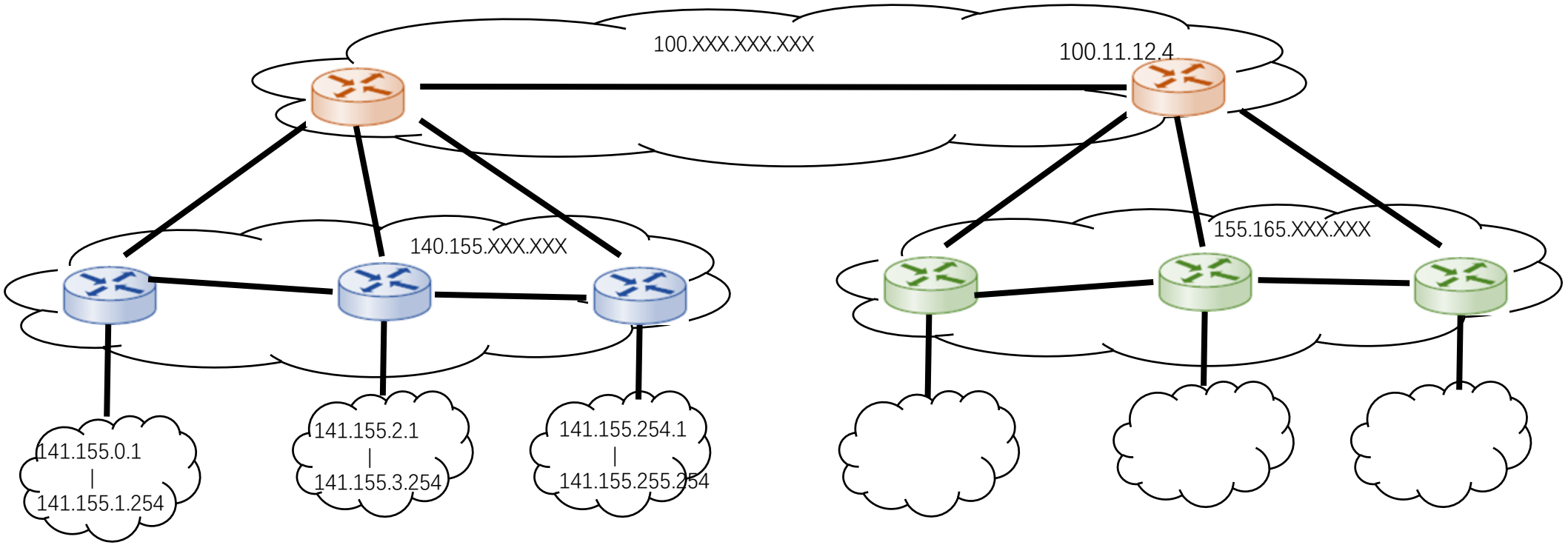
IP Address

- IP Address: 32-bit identifier for host, router ports
  - Globally unique (original goal)
  - Hierarchical: network + host (original goal)



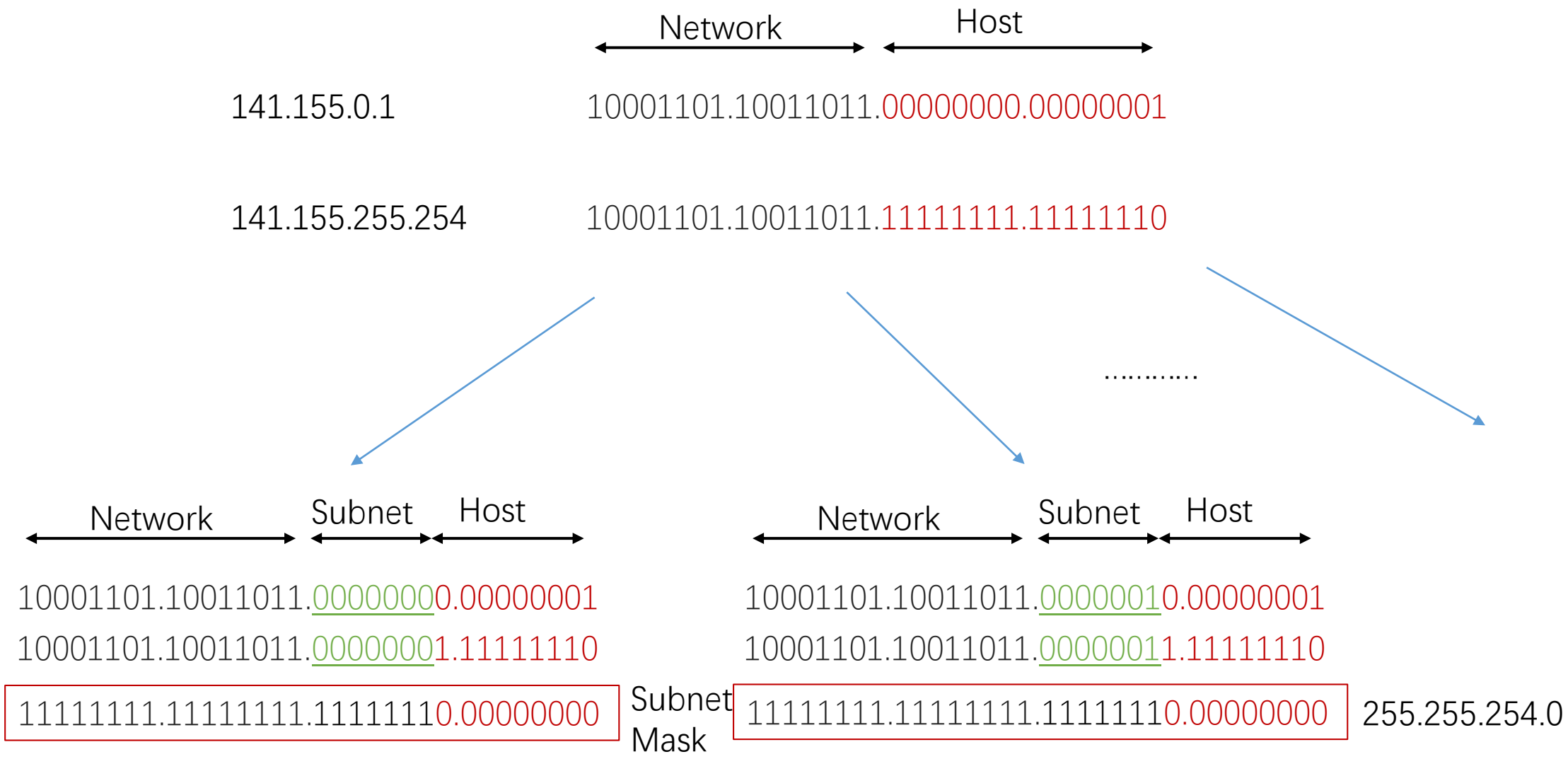
Subnet


Network
Host
141.155.0.1
10001101.10011011.00000000.00000001
141.155.255.254
10001101.10011011.11111111.11111110
.....
Network
Subnet
Host
10001101.10011011.00000000.00000001
10001101.10011011.00000001.11111110
Network
Subnet
Host
10001101.10011011.00000010.00000001
10001101.10011011.00000011.11111110



How to Determine the Network Part ?

Subnet Mask

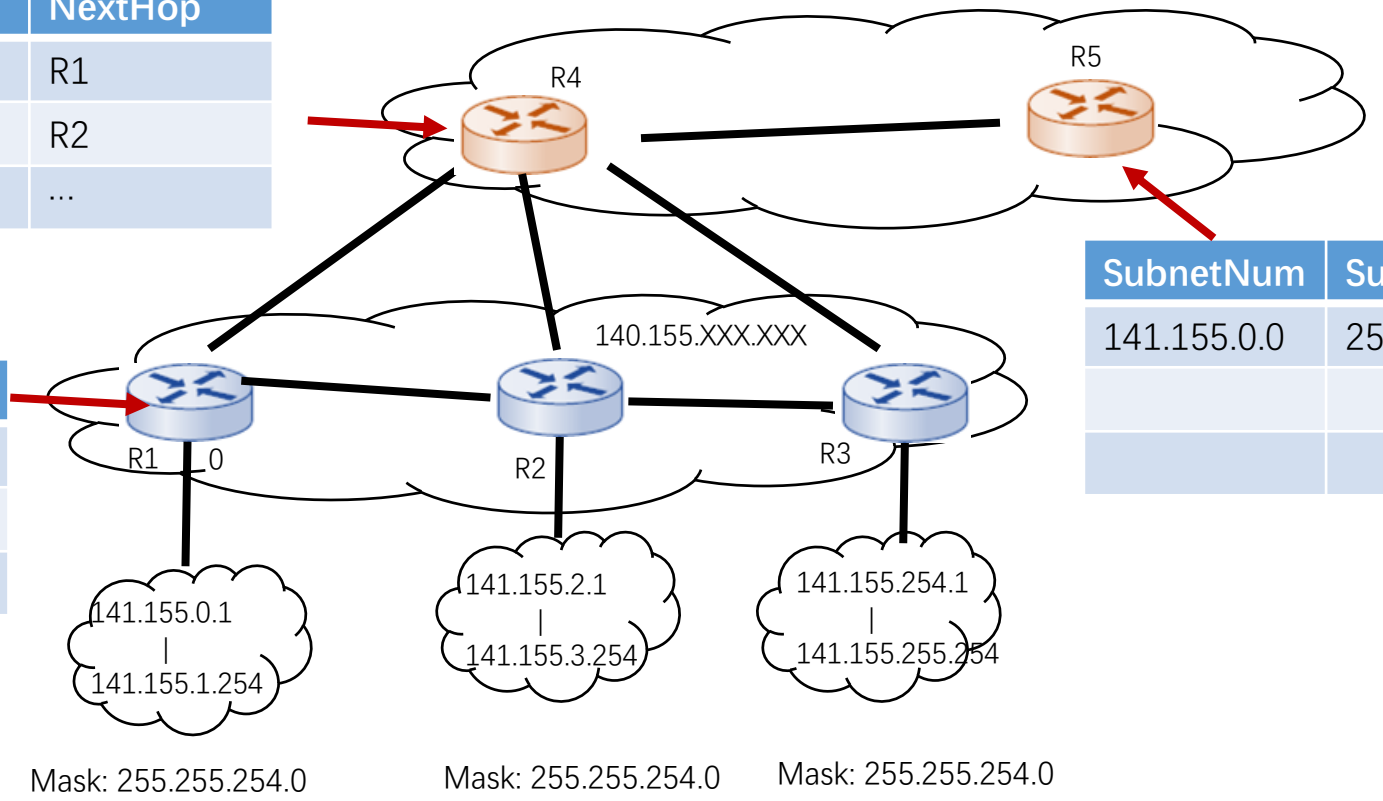

Network
Host
141.155.0.1
10001101.10011011.00000000.00000001
141.155.255.254
10001101.10011011.11111111.11111110
.....
Network
Subnet
Host
10001101.10011011.00000000.00000001
10001101.10011011.00000001.11111110
11111111.11111111.11111110.00000000
Subnet Mask
Network
Subnet
Host
10001101.10011011.00000010.00000001
10001101.10011011.00000011.11111110
11111111.11111111.11111110.00000000
255.255.254.0

Subnet Mask

- "and" IP address with network mask to determine the Subnet

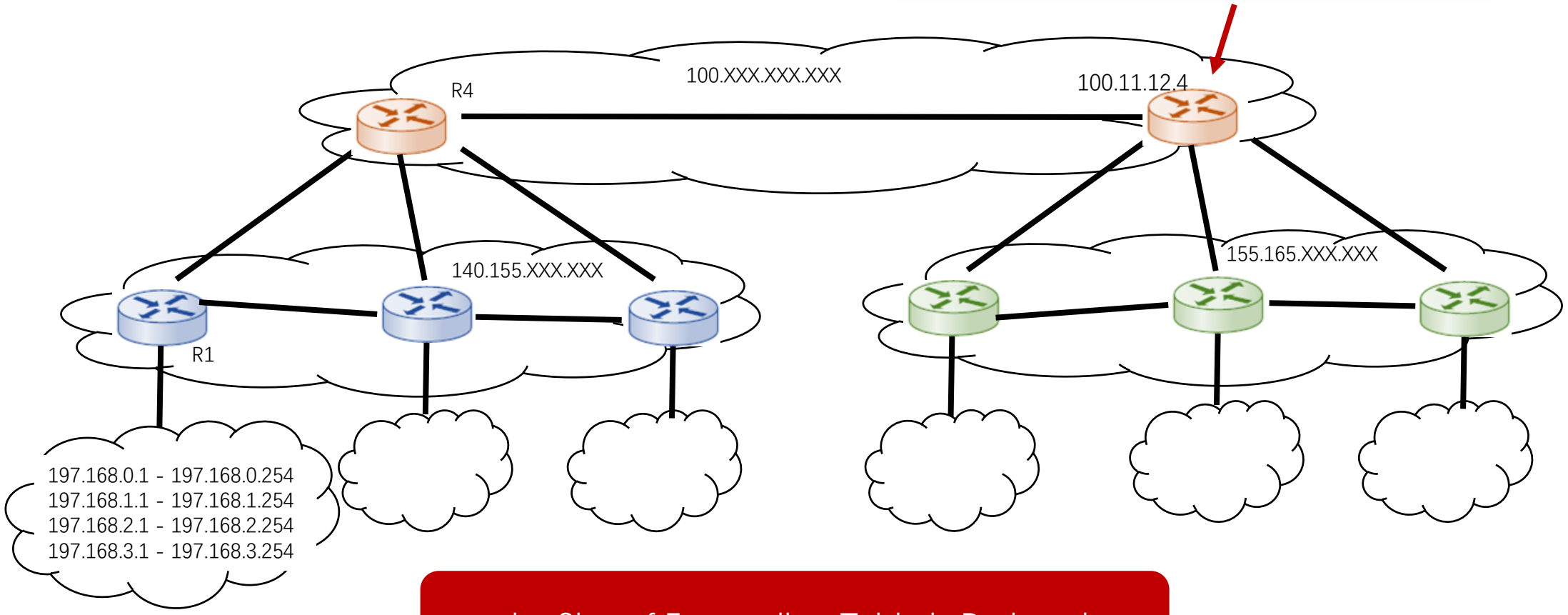
SubnetNum	SubnetMask	NextHop
141.155.0.0	255.255.254.0	R1
141.155.2.0	255.255.254.0	R2
…	…	…

SubnetNum	SubnetMask	NextHop
141.155.0.0	255.255.254.0	Interface 0
141.155.2.0	255.255.254.0	R2
…	…	…



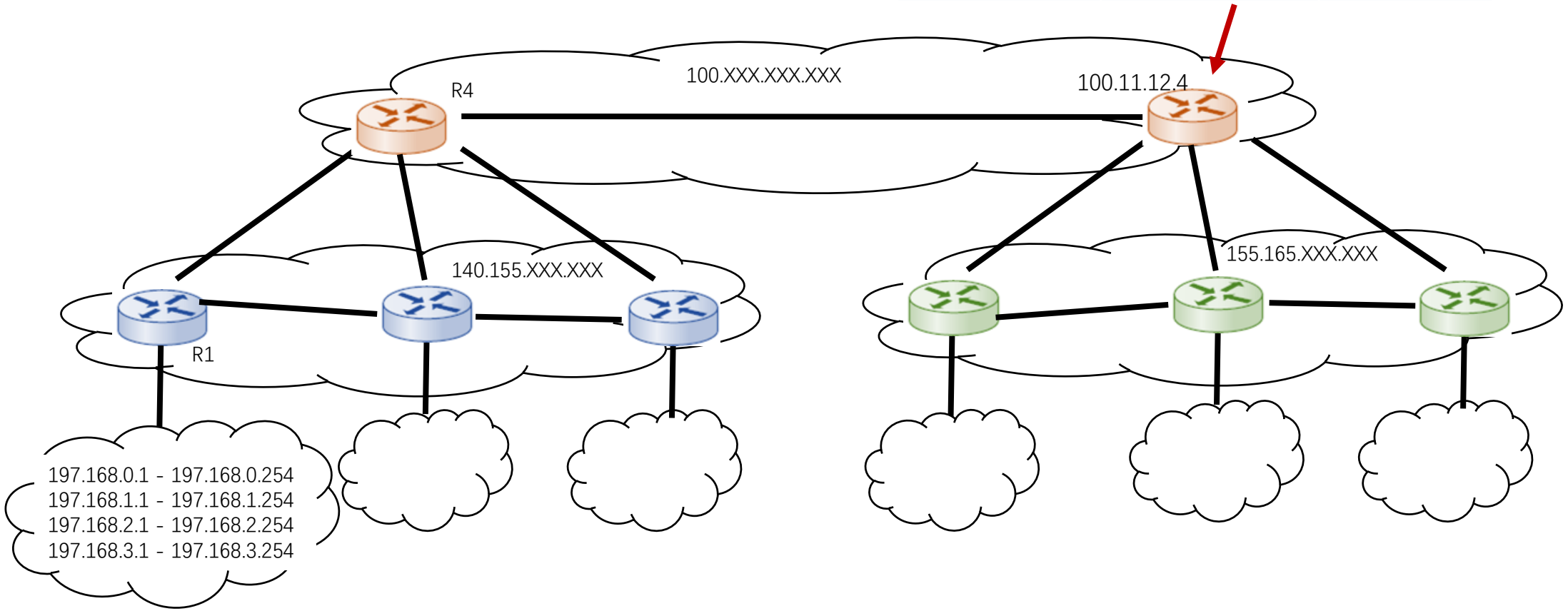
SubnetNum	SubnetMask	NextHop
141.155.0.0	255.255.0.0	R4

SubnetNum	SubnetMask	NextHop
197.168.0.0	255.255.255.0	R4
197.168.1.0	255.255.255.0	R4
197.168.2.0	255.255.255.0	R4
197.168.3.0	255.255.255.0	R4



the Size of Forwarding Table is Reduced

SubnetNum	SubnetMask	NextHop
197.168.0.0	255.255.252.0	R4



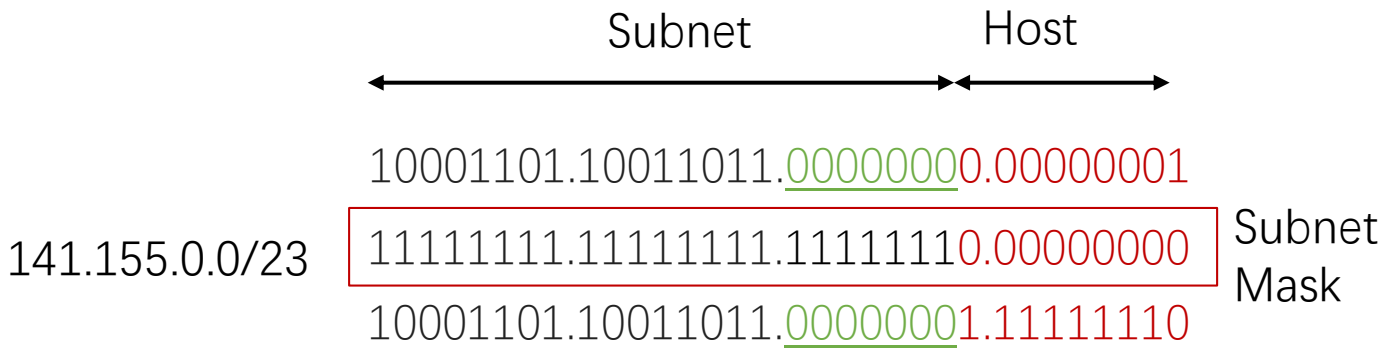
Classless InterDomain Routing (CIDR)

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

141.155.0.1 — 10001101.10011011.00000000.00000001

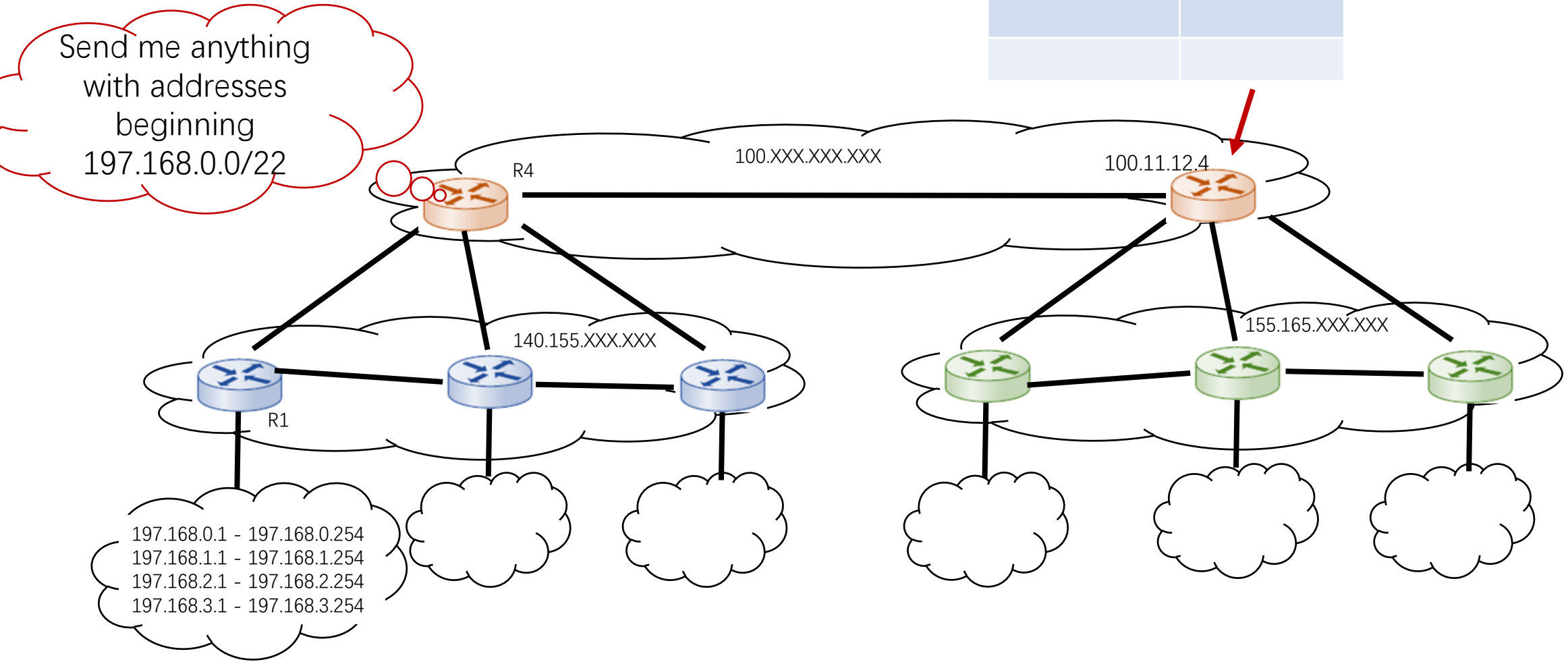
|

141.155.1.254 — 10001101.10011011.00000001.11111110



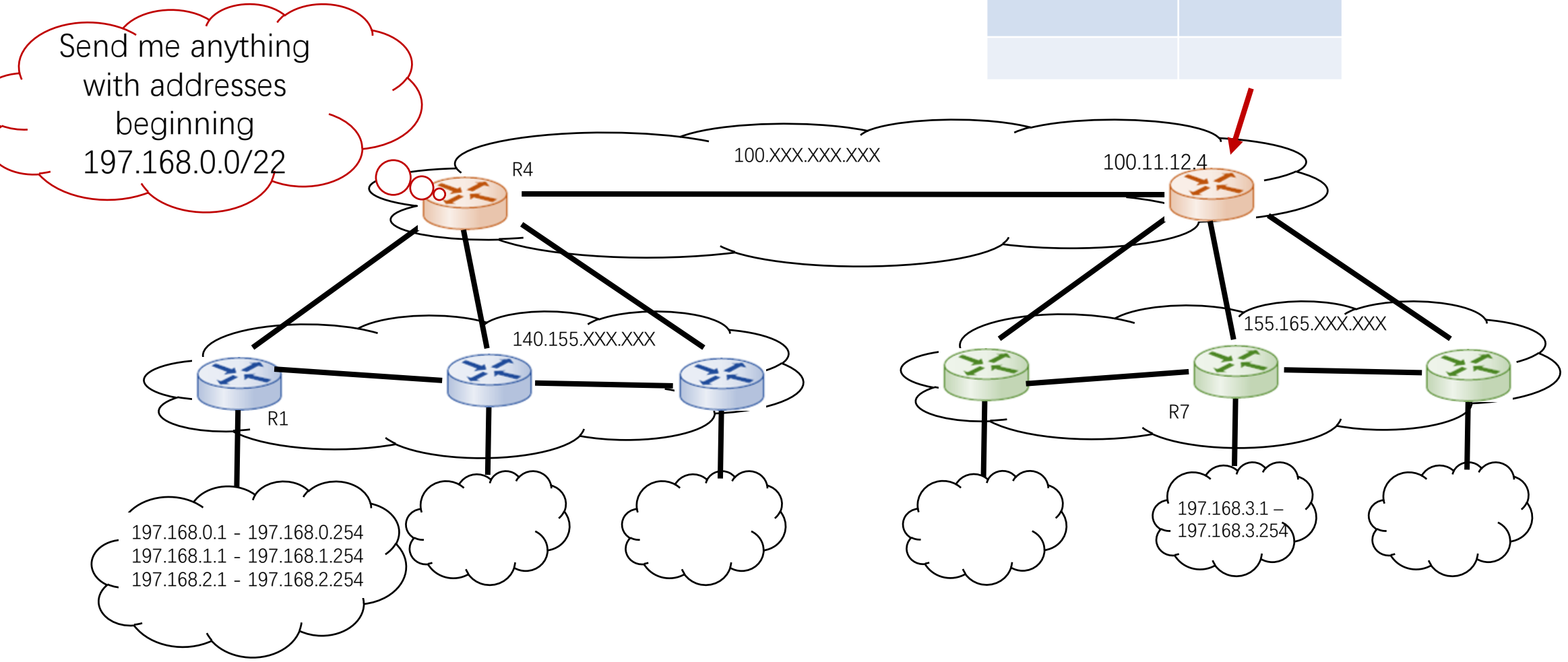
Route Aggregation

SubnetNum	NextHop
197.168.0.0/22	R4



More Specific Routes

SubnetNum	NextHop
197.168.0.0/22	R4
197.168.3.0/24	R7



Longest Prefix Matching

- When looking for forwarding table entry for given destination address, use longest address prefix that matches destination address.

SubnetNum	NextHop	
197.168.0.0/22	R4	11000101.10101000.000000**.********
197.168.3.0/24	R7	11000101.10101000.00000011.********
197.168.4.0/22	R9	11000101.10101000.000001**.********

11000101.10101000.00000011.11010111 → R7

11000101.10101000.00000111.11010111 → R9

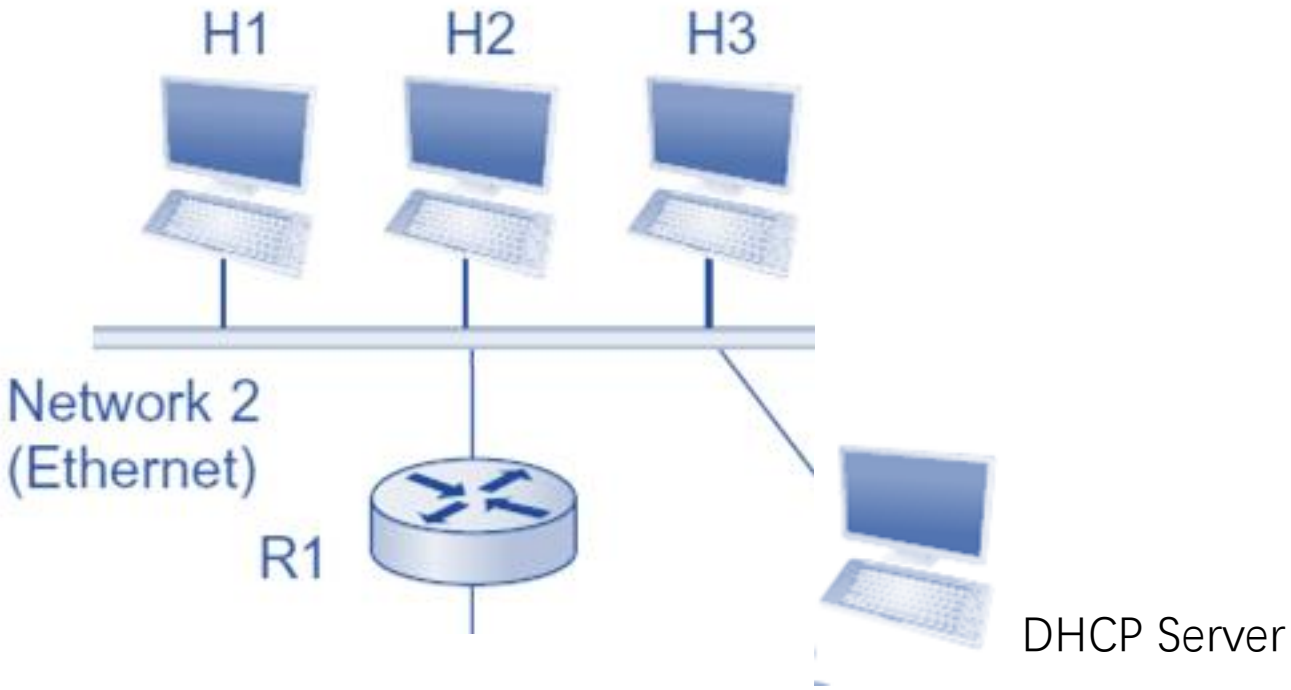
How to Assign IP Addresses ?

- Hard-coded
- Dynamic Host Configuration Protocol (DHCP)
  - Dynamically get IP address from server

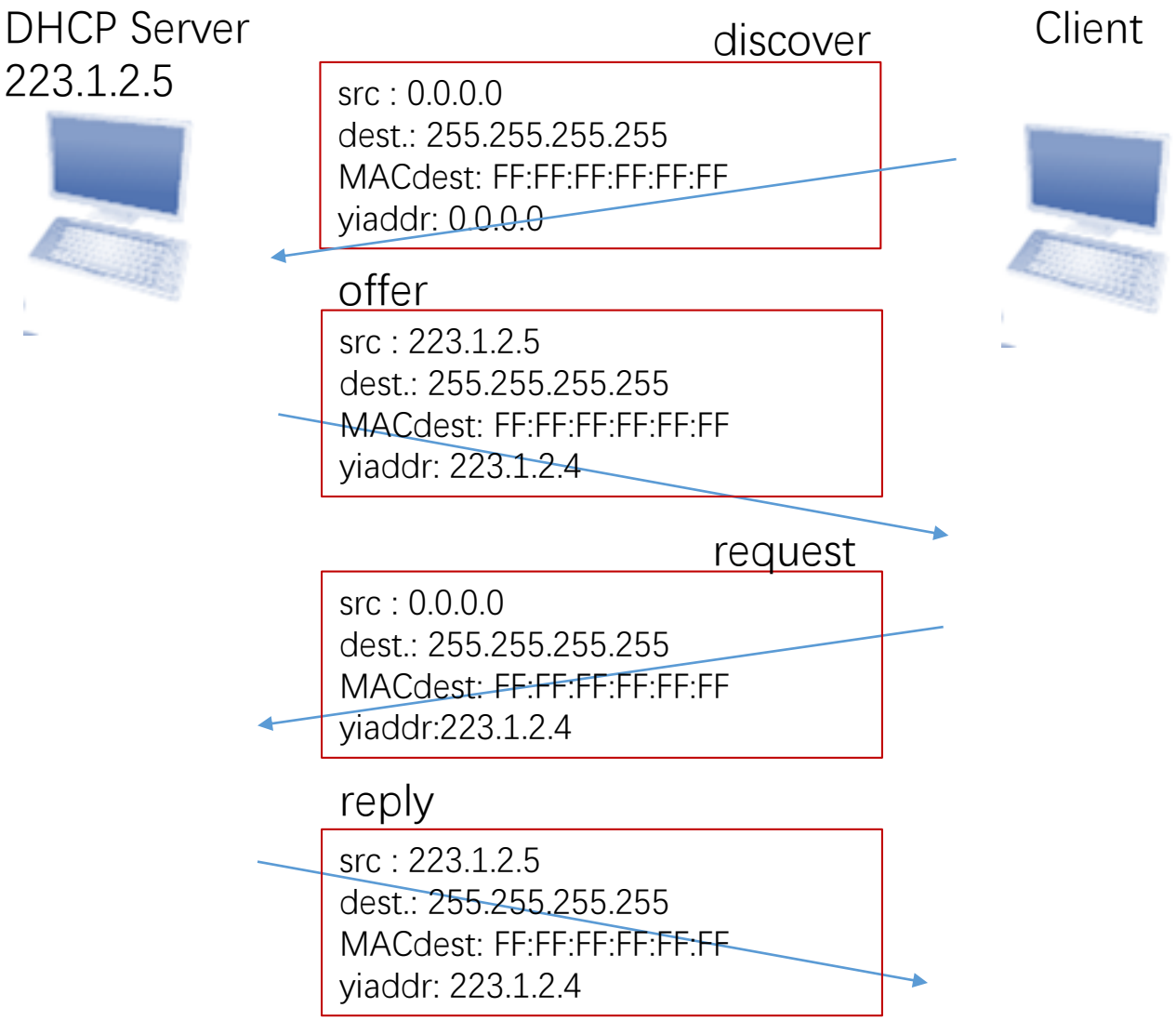
Dynamic Host Configuration Protocol (DHCP)

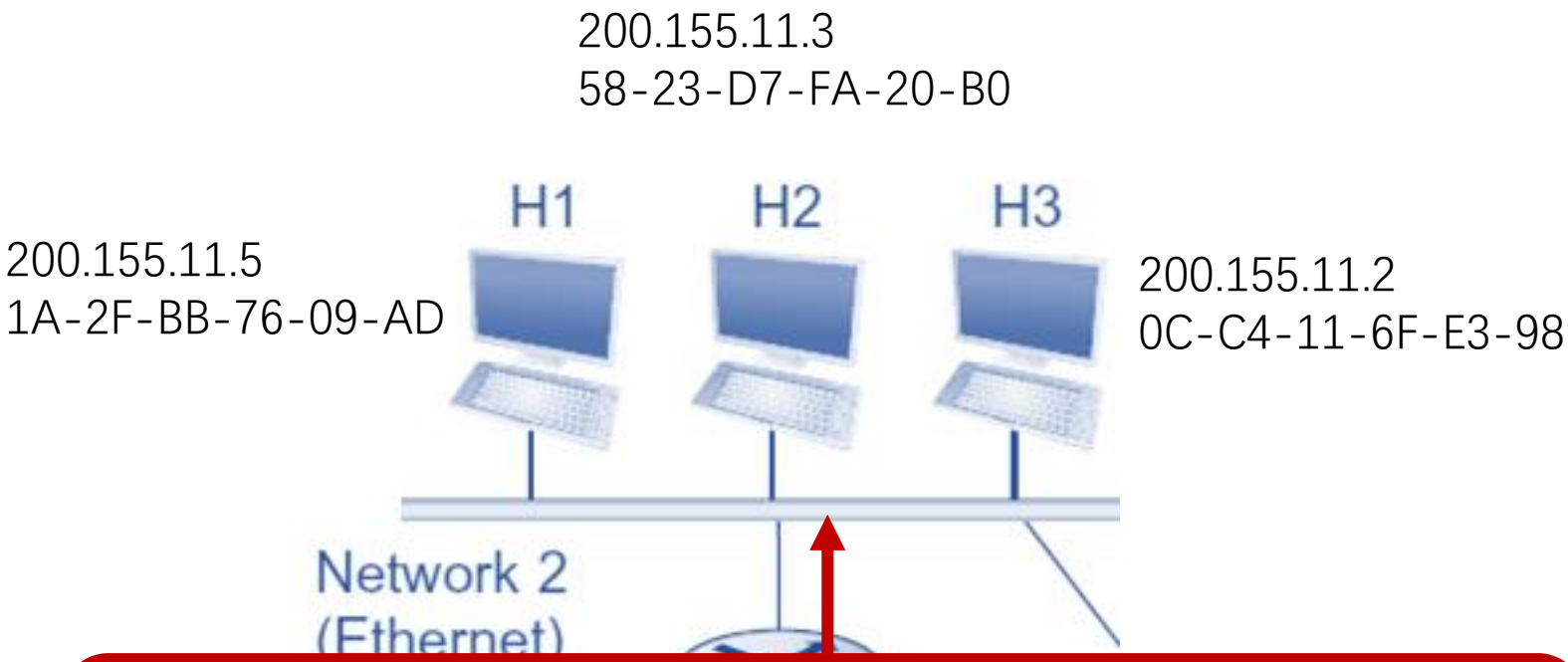
- Goal: allow host to dynamically obtain its IP address from network server when it joins the network
  - Reuse of IP addresses
    - Release IP of unconnected host, e.g. power-off
    - Support for mobile hosts who want to join the network

DHCP



DHCP


DHCP Server
223.1.2.5
Client
discover
src : 0.0.0.0
dest.: 255.255.255.255
MACdest: FF:FF:FF:FF:FF:FF
yiaddr: 0.0.0.0
offer
src : 223.1.2.5
dest.: 255.255.255.255
MACdest: FF:FF:FF:FF:FF:FF
yiaddr: 223.1.2.4
request
src : 0.0.0.0
dest.: 255.255.255.255
MACdest: FF:FF:FF:FF:FF:FF
yiaddr:223.1.2.4
reply
src : 223.1.2.5
dest.: 255.255.255.255
MACdest: FF:FF:FF:FF:FF:FF
yiaddr: 223.1.2.4



How to Determine Interface's MAC address, Knowing its IP address?

Address Resolution Protocol (ARP)

- ARP table: each IP node (host, router) on LAN has table IP/MAC address mappings for some LAN nodes
  - < IP address; MAC address; TTL>
- TTL (Time To Live)
  - Time after which address mapping will be forgotten

200.155.11.3; 58-23-D7-FA-20-B0
200.155.11.5; 1A-2F-BB-76-09-AD
200.155.11.2; 0C-C4-11-6F-E3-98

Address Resolution Protocol (ARP)

- A wants to send datagrams to B
  - if B's MAC address is in the same subnet and B's MAC address not in A's ARP table
    - A broadcasts ARP query packet, containing B's IP address
- B receives ARP packet, replies to A with its (B's) MAC address
  - Frame sent to A's MAC address (unicast)
- A caches (saves) IP-to-MAC address pair in its ARP table until information becomes old (times out)


Addressing: from one
LAN to Another

Demo

- DHCP
  - Four handshake messages
    - `ipconfig /release`
    - `ipconfig /renew`
- ARP
  - Show arp table: `arp -a`
- Forwarding Table
  - Show Forwarding Table: `route print`

Reference

- Textbook 3.2