

# 期末复习

---

网络协议分析

# 重要章节： IP 协议

---

第一章：网络协议、网络协议分层（计算机网络）

链路层协议： PPP 协议（了解）、以太网协议（计算机网络）、硬件设备

ARP 协议：原理、工作流程等

**IP 协议：分片、校验和、5 类地址类型、进制转化，根据掩码与 IP 地址求子网地址范围，有关子网的计算，子网划分（定长划分、变长划分），IP 选路（直接、间接交付）、路由表的设计、路由聚合等**

路由协议（计算机网络）： RIP 协议（协议原理，路由表更新方法）、OSPF 协议（协议原理、最短路径优先算法、分层路由与 5 类链路通告）

传输层协议：出错控制、流量控制、拥塞控制、连接管理

# 网络协议帧分析

---

通过 WireShark 抓包软件抓取了一个完整的 Ethernet 数据帧，如下所示。请回答下列问题：

C8 8D 83 A9 F6 F7 98 EE

CB 2E 76 9E 08 00 45 00

00 28 60 CE 40 00 40 06

33 4B 0A 19 0B 0D 65 25

2C 6C CF 9C 01 BB D3 4E

69 7A 75 FB 2B 6A 50 14

00 00 59 93 00 0

该数据帧的数据来自于网络层什么协议？ IP

# 网络协议帧分析

---

通过 WireShark 抓包软件抓取了一个完整的 Ethernet 数据帧，如下所示。请回答下列问题：

C8 8D 83 A9 F6 F7 98 EE

CB 2E 76 9E 08 00 45 00

00 28 60 CE 40 00 40 06

33 4B 0A 19 0B 0D 65 25

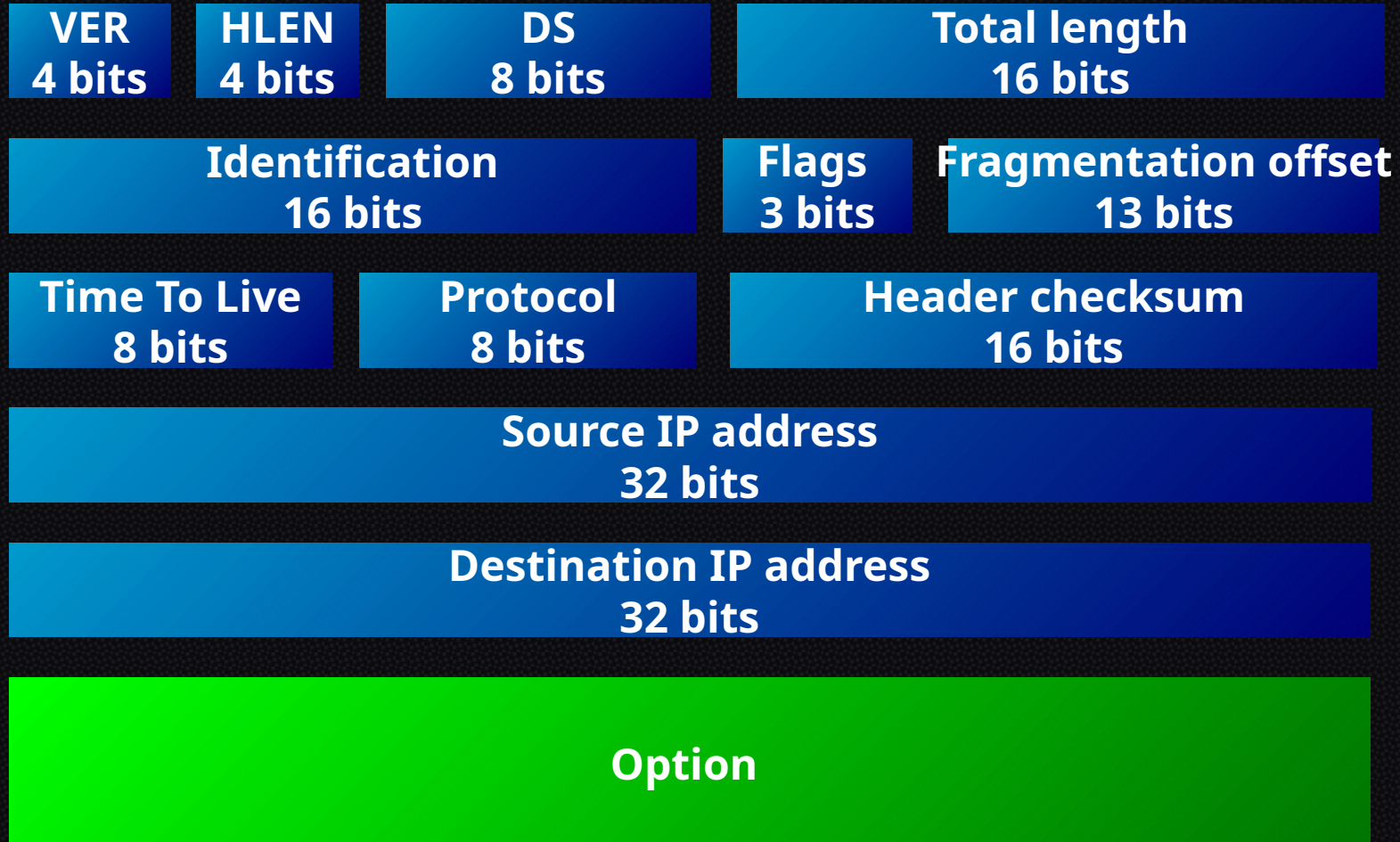
2C 6C CF 9C 01 BB D3 4E

69 7A 75 FB 2B 6A 50 14

00 00 59 93 00 0

该数据帧的数据来自于运输层什么协议？ TCP (06), 如果 UDP 是 17

# IP header



# 网络协议帧分析

---

通过 WireShark 抓包软件抓取了一个完整的 Ethernet 数据帧，如下所示。请回答下列问题：

C8 8D 83 A9 F6 F7 98 EE

CB 2E 76 9E 08 00 45 00

00 28 60 CE 40 00 40 06

33 4B 0A 19 0B 0D 65 25

2C 6C CF 9C 01 BB D3 4E

69 7A 75 FB 2B 6A 50 14

00 00 59 93 00 0

该数据帧的源主机 IP 地址是多少？ 10.25.11.13

# 网络协议帧分析

---

通过 WireShark 抓包软件抓取了一个完整的 Ethernet 数据帧，如下所示。请回答下列问题：

C8 8D 83 A9 F6 F7 98 EE

CB 2E 76 9E 08 00 45 00

00 28 60 CE 40 00 40 06

33 4B 0A 19 0B 0D 65 25

2C 6C CF 9C 01 BB D3 4E

69 7A 75 FB 2B 6A 50 14

00 00 59 93 00 0

该数据帧的数据帧的源端口是多少？ 53148 目的端口是多少？ 443

# 数据报大小与网络 MTU

- 每个 IP 分组都封装在帧中传输
  - 在 Internet 中，数据报可能穿越多种不同的物理网络到达目的地
    - MTU (Maximum Transfer Unit): 物理网络所能传输的数据的最大长度
    - 不同物理网，MTU 不同！
- 如何确定分组的大小？（IP 的解决思路）
  - IP 向高层隐藏底层网络，方便用户通信，IP 协议自己处理不同物理网络的 MTU 要求
  - 方法：
    - 对于高层协议：数据报最大长度 =  $2^{16}$ Byte（独立于任何物理网络的 MTU）
    - 面对物理网络：提供 Fragmentation（分片）和 reassembly（重组）机制

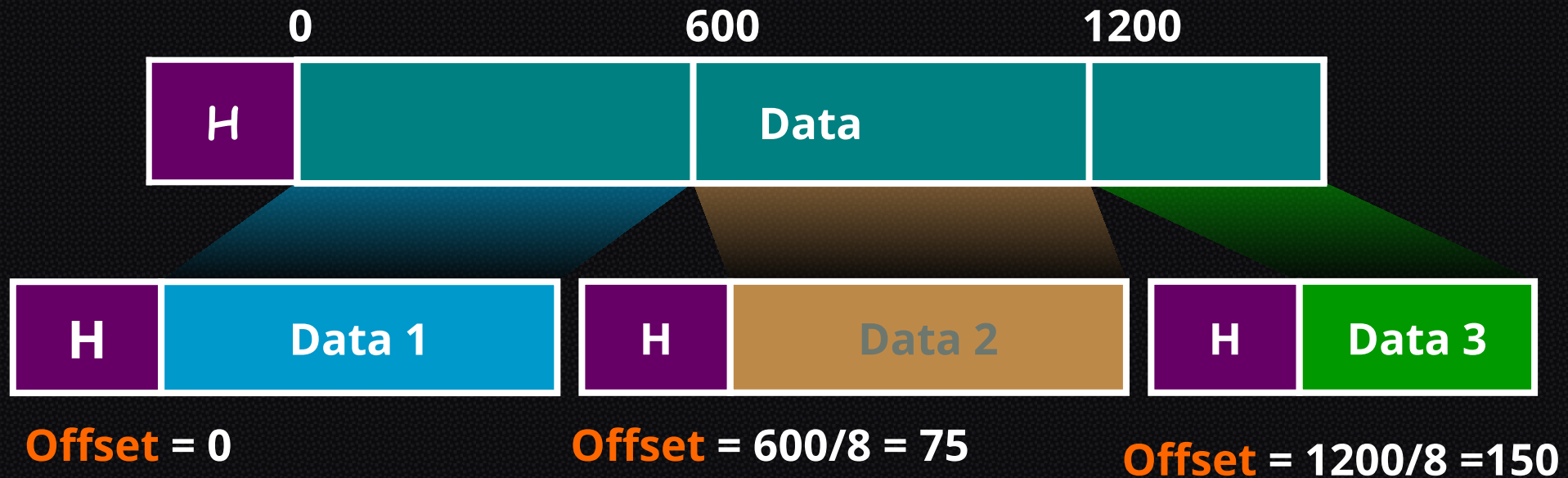
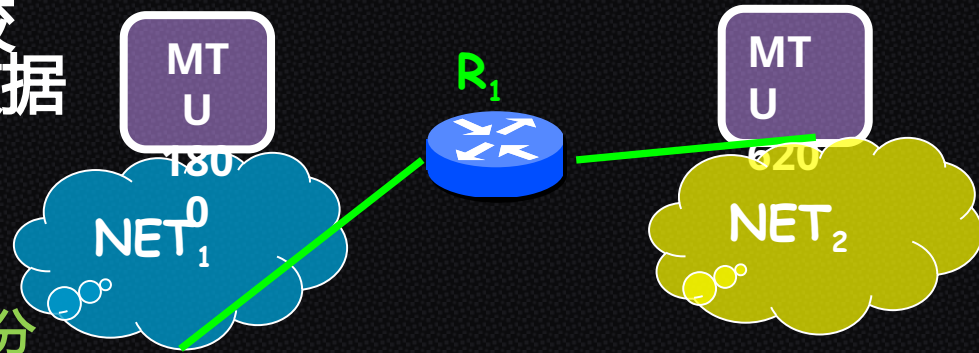


# Fragmentation (分片)

- Fragmentation: 在 MTU 较小的网络上, IP 将较长的数据报划分成更小的部分 (Fragment)

- 分片的大小:

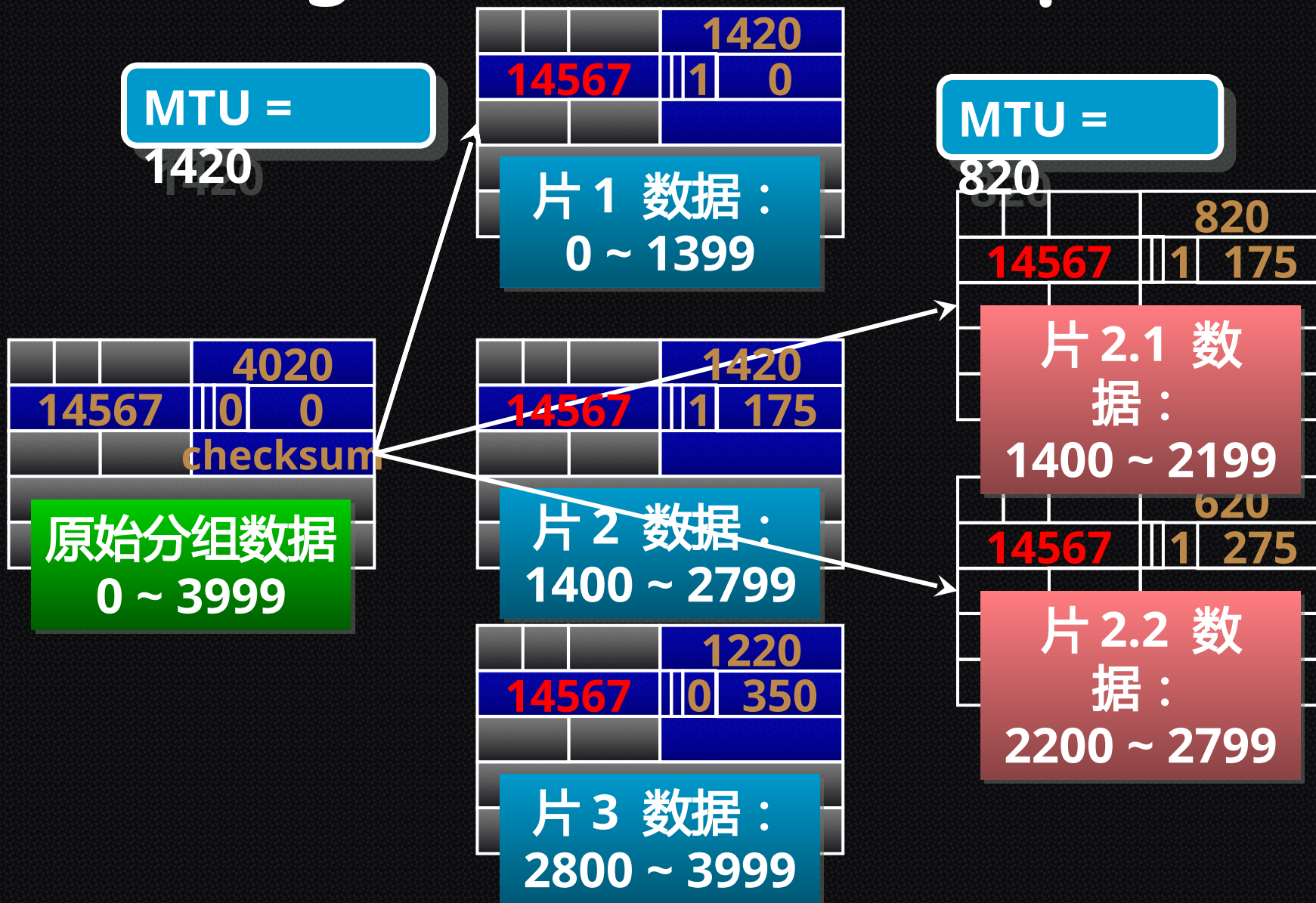
- 8 字节的整数倍 (除最后 1 个分片) —— 保证偏移值是以 8 字节为单位
- 最接近网络的 MTU



# Fragmentation Operation

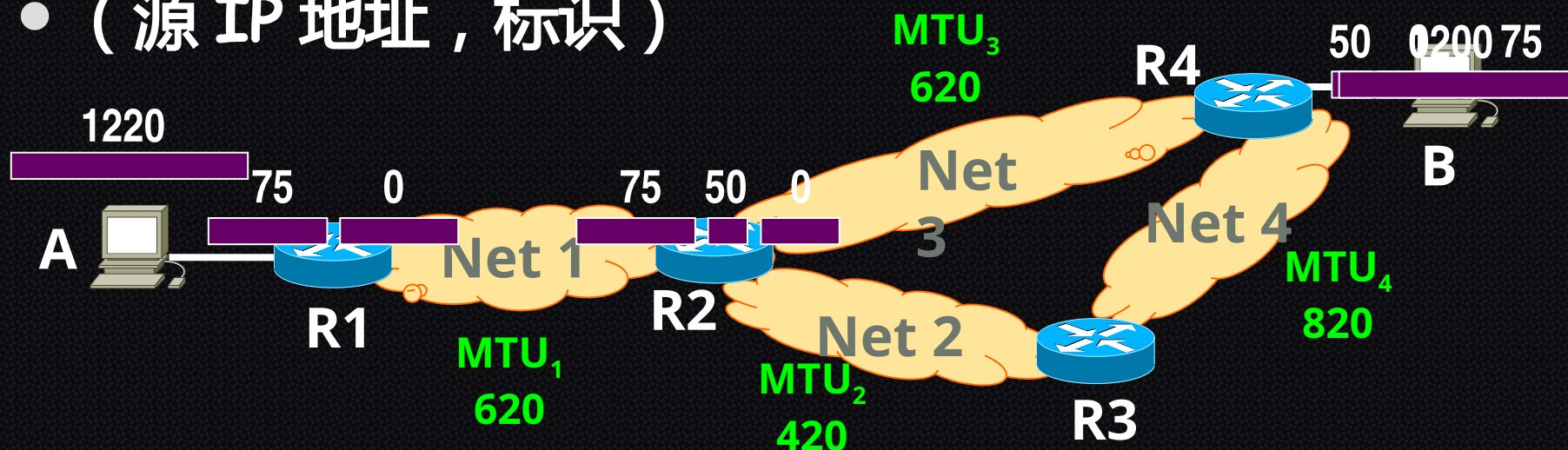
- To be fragmented by *routers* and a *source host*
- A datagram can be fragmented more than once
- An IP header must be included in each fragment
  - All fragments have the same *identification number* as the original datagram
  - The values of the three fields must be changed :
    - *Flags, Fragmentation offset, Total length*
  - *Checksum* must be recalculate
- The reassembly of datagram is done only by the *destination host*
  - 原因
    - Each fragment is an independent datagram
    - Maybe travel through different routes

# Fragmentation Example



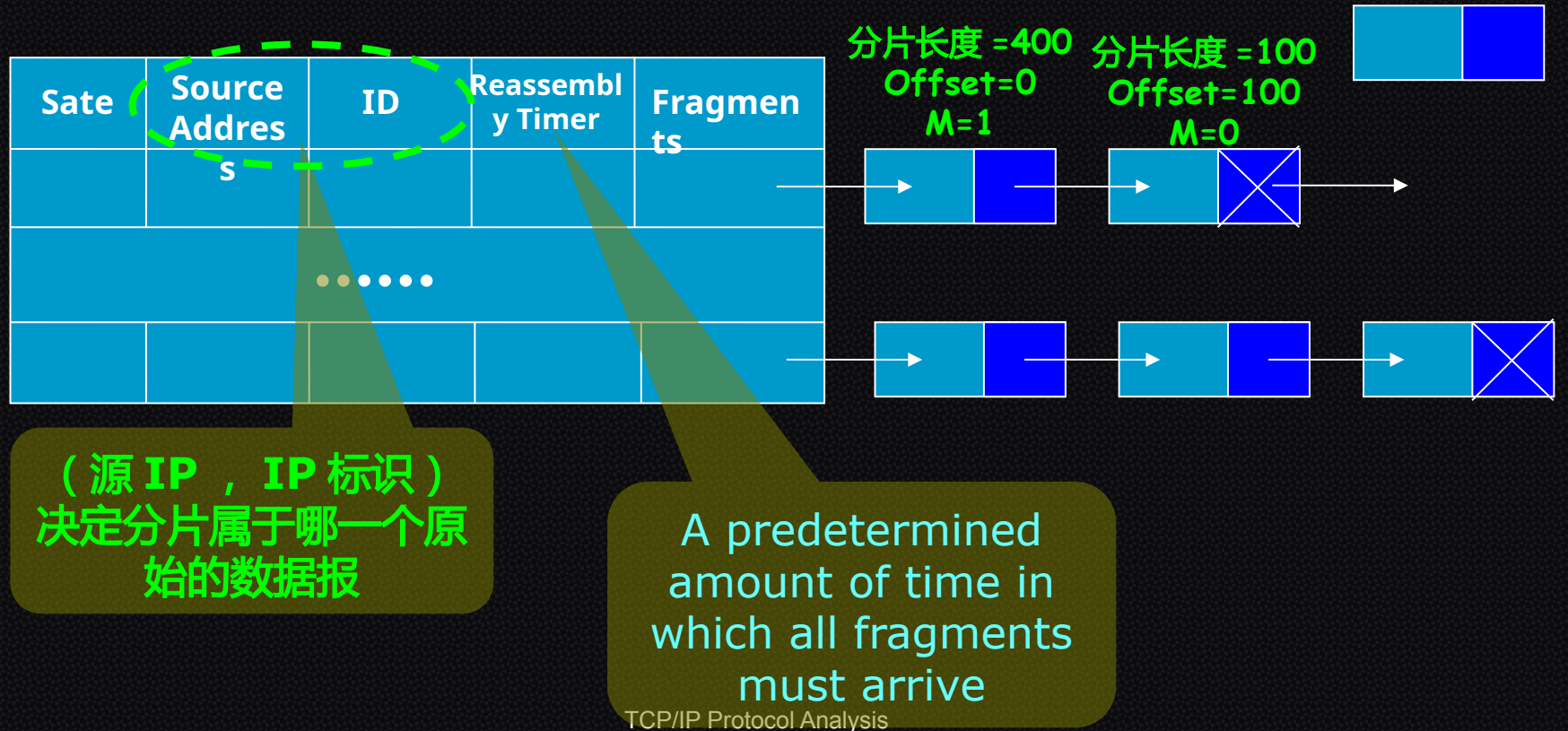
# Reassembly ( 重组操作 )

- Only by *destination host*
  - Connectionless → 各分片的传输路径可不同
  - 有重组时限✉ *Reassembly Timer*
  - 若丢失分片，则无法重组 IP 分组
- 目的主机能区分和重组不同的源 IP 报文
  - ( 源 IP 地址，标识 )



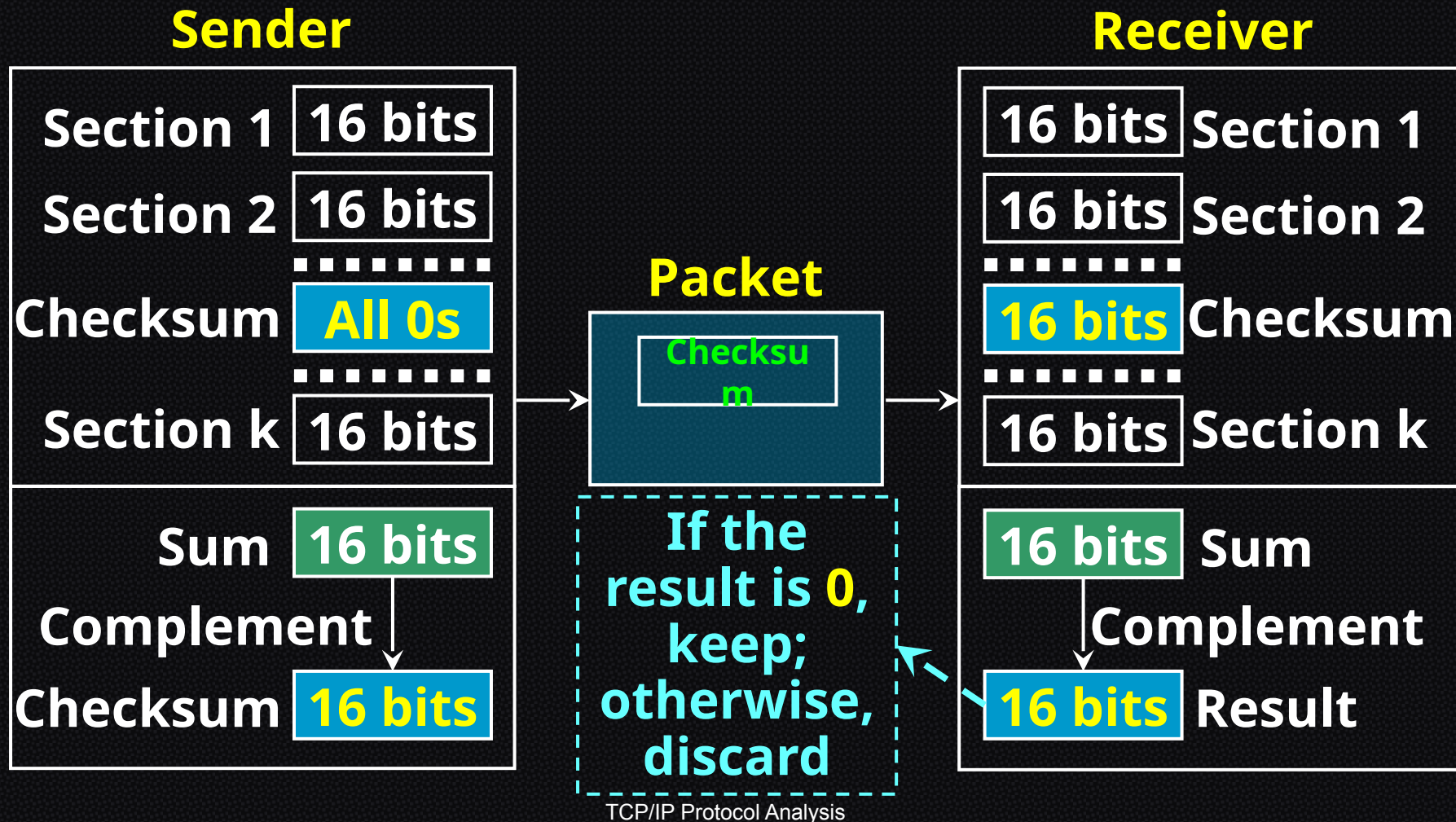
# 如何重组——重组表

- 重组表的功能：
  - 找出一个分片属于哪一个原始的数据报
  - 将属于同一个数据报的分片进行排序



# Checksum

- RFC 1071 ( RFC 1141 , 1624 , 1936 ) , 附录 C
- IP 分组的校验和只包括**分组首部** , 而不包括数据



# Example

4	5	0	30	
1			0	0
10	17	0		
100.12.14.5				
102.36.7.9				

```

01000101 00000000
00000000 00011110
00000000 00000001
000 00000 00000000
00001010 00010001
00000000 00000000
01100100 00001100
00001110 00000101
01100110 00100100
+ 00000111 00001001
-----
00101110 01101110

```

Checksum : 0x D190

11010001 10010000

取反

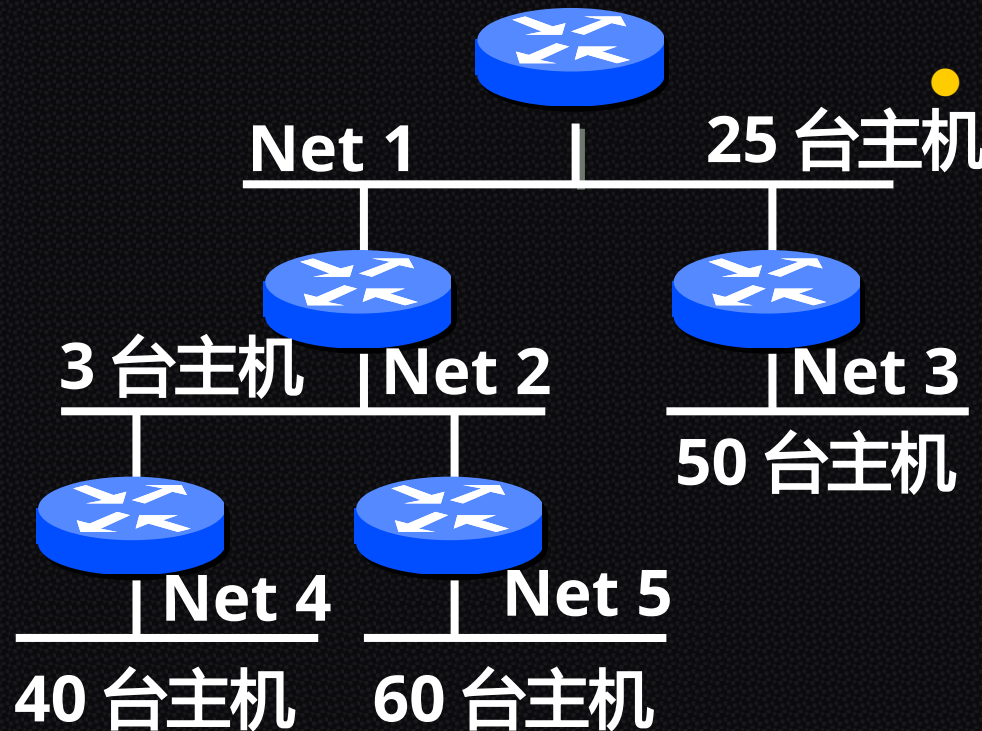


# 子网划分方法

- Fix length subnetting ( 定长子网划分 )
  - 共享同一 IP 网络前缀 ( Net id + Subnet id ) 的子网大小相同
  - 划分依据：子网数量与子网内主机数量折中
  - 特点：划分简单，地址分配较浪费
- Variable length subnetting ( 变长子网划分 )
  - 共享同一 IP 网络前缀 ( Netid ) 的子网大小不同
  - 划分依据：子网内的主机数量
  - 特点：灵活、高效利用地址空间
  - 变长子网掩码 ( Variable-Length Subnet Mask , VLSM )



# Subnetting Example 4

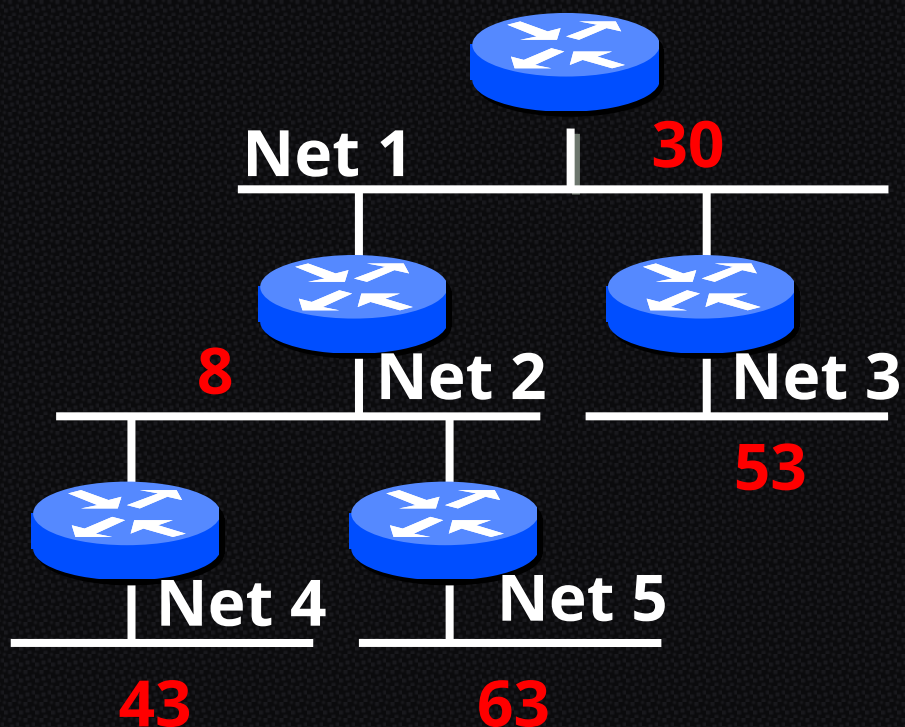


• 每个物理网络所需的 IP 地址数量

- $\text{Net 1} = 25 + 3 + 2 = 30$
- $\text{Net 2} = 3 + 3 + 2 = 8$
- $\text{Net 3} = 50 + 1 + 2 = 53$
- $\text{Net 4} = 40 + 1 + 2 = 43$
- $\text{Net 5} = 60 + 1 + 2 = 63$

本子网内主机数 + 本子网内路由器接口数 + 2

# Example 3 —— 定长划分



- 依据：
  - 子网数 = 5  $\rightarrow$  Subnetid  $\geq$  3bit
  - 主机数  $\geq$  63  $\rightarrow$  Hostid  $\geq$  6bit
- 结论：
  - 至少是 B 类地址块
  - Mask = 255.255.255.192
- 每个 IP 子网的实际 IP 数量：64
- IP 地址总空间： $5 * 64 = 320$

假设使用 B 类地址块

129.1.0.0/26

Net 2: 129.1.0.64/26

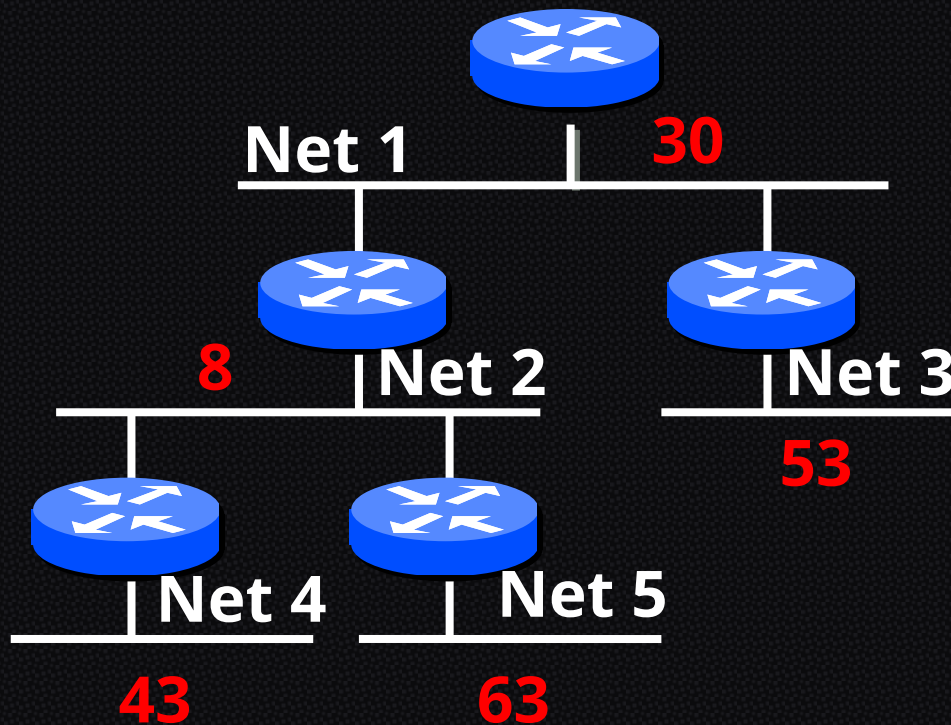
Net 4: 129.1.0.192/26

Net 1:

Net 3: 129.1.0.128/26

Net 5: 129.1.1.0/26

# Example 3 —— 变长划分



- Net 1 Mask =  $/(32-5)=/27$
- Net 2 Mask =  $/(32-3)=/29$
- Net 3 Mask =  $/(32-6)=/26$
- Net 4 Mask =  $/26$
- Net 5 Mask =  $/26$
- IP 地址总空间：  
 $32+8+64+64+64 = 232$

VLS

M

Net 1: 211.1.0.0/27

Net 2: 211.1.0.32/29

Net 3: 211.1.0.64/26

Net 4: 211.1.0.128/26

Net 5: 211.1.0.192/26

# Discussion 1

- Net 1(/27) :  
211.1.0.0/27
  - 211.1.0.0 ~ 211.1.0.31
- Net 2(/29) :  
211.1.0.32 ~ 211.1.0.39
- Net 3(/26) :  
211.1.0.64 ~ 211.1.0.127
- Net 4(/26) :  
211.1.0.128/26
  - 211.1.0.128 ~ 211.1.0.191
- Net 5(26) :  
211.1.0.192/26
  - 211.1.0.192 ~ 211.1.0.255

掩码中 0 bit 个数  $= 32 - 27 = 5$

子网中 IP 地址个数  $= 2^5 = 32$

问题：不能

Net3 的 IP 地址范围能否是  
211.1.0.40~  
211.1.0.103 ?

40    00101000

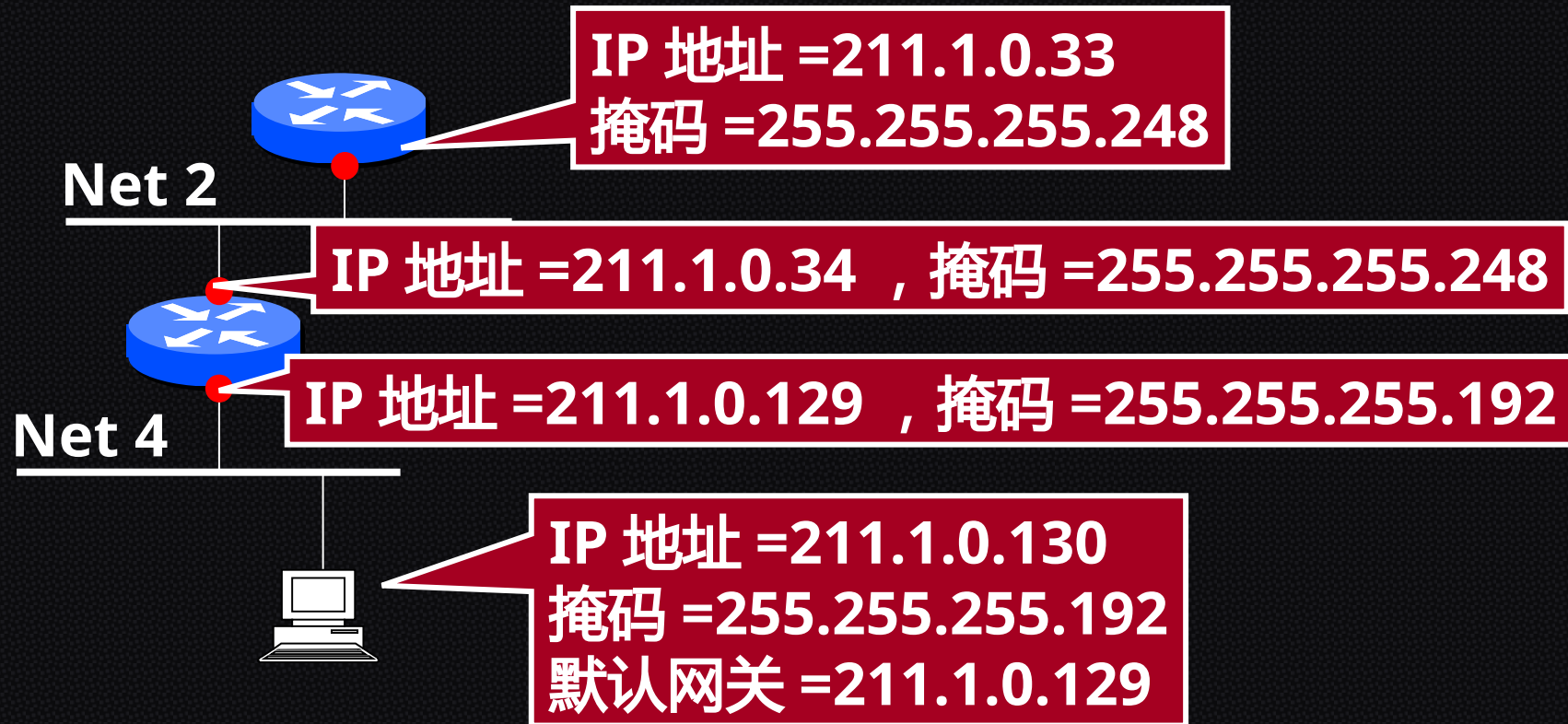
64    01000000

# Discussion 2

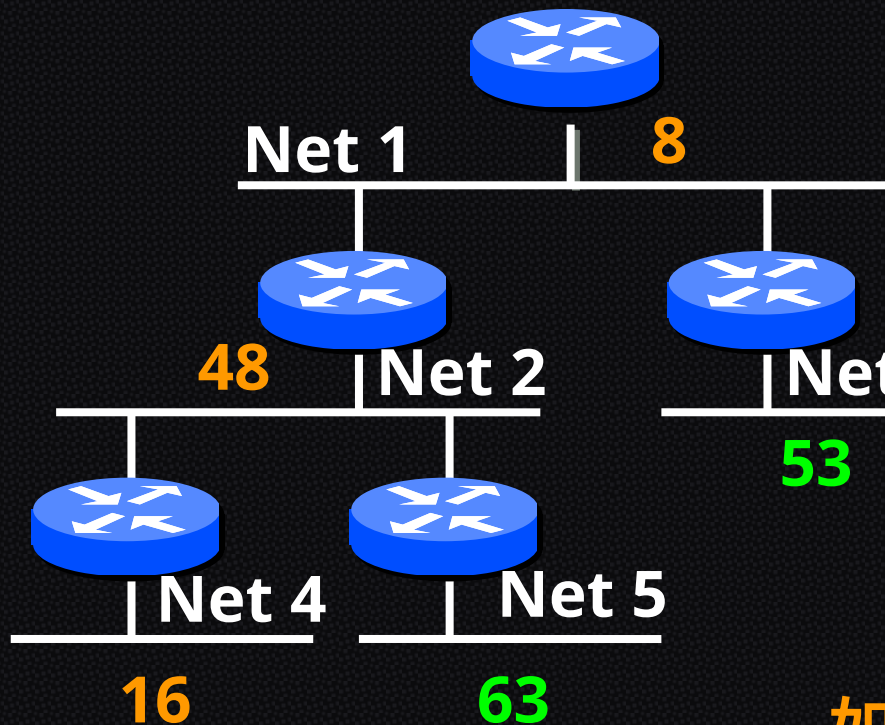
- 为 Net 4 分配 IP 地址
  - 211.1.0.128/26 ( 211.1.0.128 ~ 211.1.0.191 )
- 分析
  - 需要分配 IP 地址的设备：路由器、主机
  - 可以分配的 IP 地址：211.1.0.129 ~ 211.1.0.190
- 解答
  - 路由器接入 Net 4 的接口 IP 地址：211.1.0.129
  - Net 4 中的主机 IP 地址：211.1.0.130 ~ 211.1.0.190

# Assign IP Address

- Net 2 : 211.1.0.32/29 211.1.0.32 ~ 211.1.0.38
- Net 4 : 211.1.0.128/26 211.1.0.129 ~ 211.1.0.190



# Example 3 —— 编址技巧



Net 1: 211.1.0.0/29

Net 2: 211.1.0.64/26

Net 3: 211.1.0.128/26

- Net 1 Mask =  $/(32-3)=/29$
- Net 2 Mask =  $/(32-6)=/26$
- Net 3 Mask =  $/(32-6)=/26$
- Net 4 Mask =  $/28$
- Net 5 Mask =  $/26$
- IP 地址总空间：

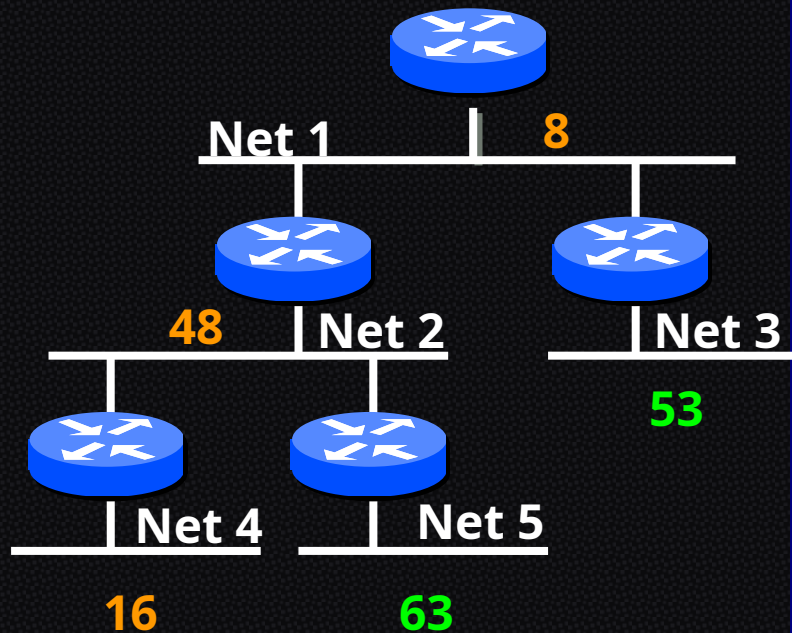
$$8+64+64+16+64 = 232$$

如果：Net 4: 211.1.0.192/28

那么 Net 5

211.1.1.0/26 ?

# Example 3 —— 编址技巧



Net 2: 211.1.0.0/26

Net 3: 211.1.0.64/26

Net 5: 211.1.0.128/26

Net 4: 211.1.0.192/28

Net 1: 211.1.0.208/29

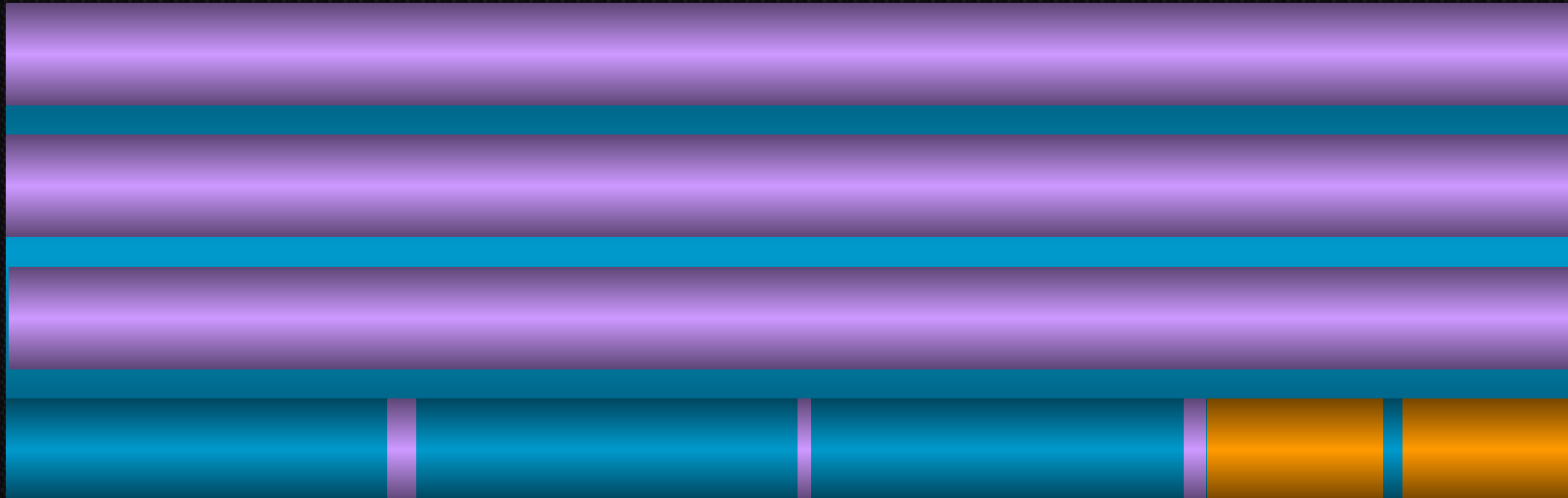
- 按照子网空间的大小，从大到小逐次确定变长掩码长度
  - Net2 , Net3 , Net5 ( Hostid = 6bit )
  - Net4 ( Hostid = 4bit )
  - Net1 ( Hostid = 3bit )
- 第一个掩码：/26，将地址空间分为 4 个 1 级子网 ( 地址空间 = 64 )
  - 三个 1 级子网地址空间分配给 Net2 , Net3 , Net5
- 对于剩下的那个 1 级子网空间 ( 64 )，使用第二个掩码：/28，将地址空间分为 4 个 2 级子网 ( 地址空间 = 16 )
  - 给 Net4 分配一个 2 级子网
- 使用第三个掩码：/29，将剩下的某个 2 级子网分成 2 个 3 级子网 ( 地址空间 = 8 )
  - 给 Net1 分配一个 3 级子网



# Example 3 —— 编址技巧

- 第一个掩码： $/26$ ，将某个 C 类地址空间分为 4 个 1 级子网（地址空间 = 64）
- 第二个掩码： $/28$ ，将剩下的某个 1 级子网地址空间分为 4 个 2 级子网（地址空间 = 16）
- 第三个掩码： $/29$ ，将剩下的某个 2 级子网分成 2 个 3 级子网（地址空间 = 8）

Address space



# Example 2

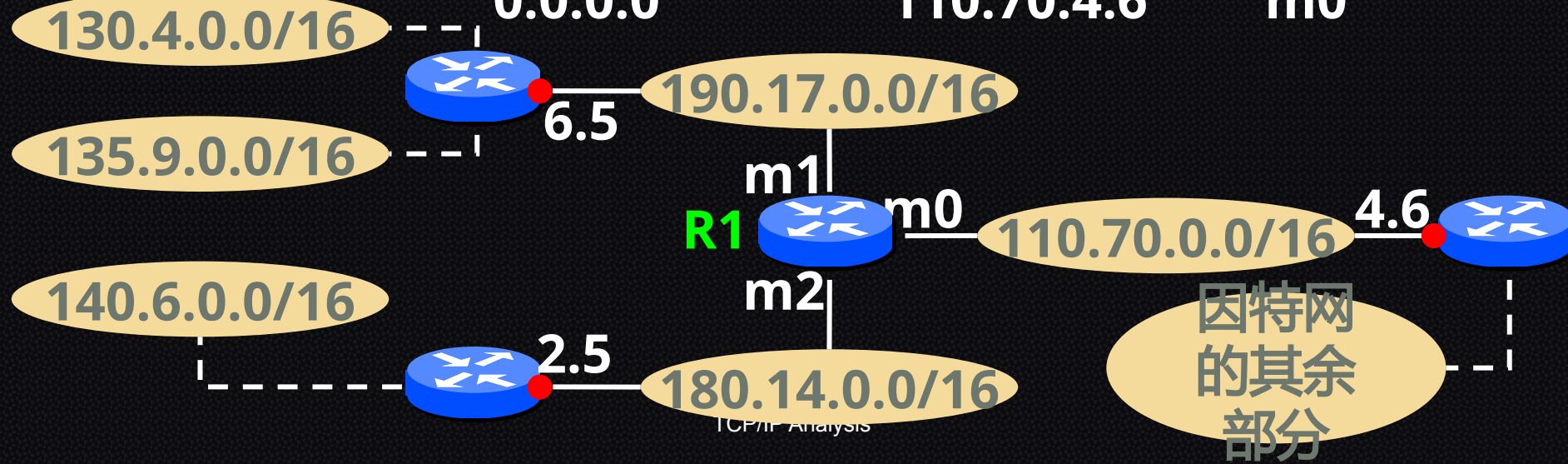
- 路由器 R1 的路由表如下：

Mask	Destination	Next Hop	Interface
255.255.0.0	110.70.0.0	--	m0
255.255.0.0	180.14.0.0	--	m2
255.255.0.0	190.17.0.0	--	m1
255.255.0.0	130.4.0.0	190.17.6.5	m1
255.255.0.0	135.9.0.0	190.17.6.5	m1
255.255.0.0	140.6.0.0	180.14.2.5	m2
0.0.0.0	0.0.0.0	110.70.4.6	m0

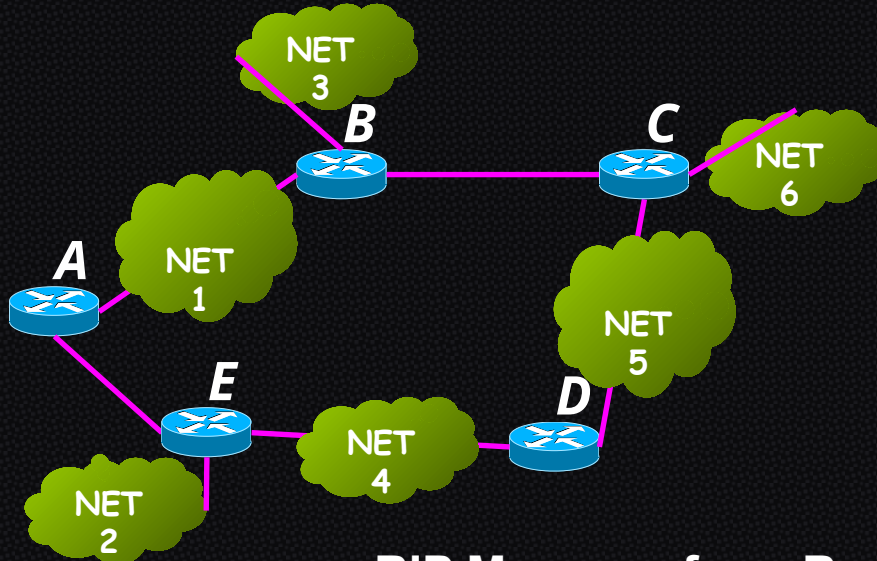
根据以上路由表画出网络拓扑图。

# Solution

Mask	Destination	Next Hop	
255.255.0.0	110.70.0.0	--	m0
255.255.0.0	180.14.0.0	--	m2
255.255.0.0	190.17.0.0	--	m1
255.255.0.0	130.4.0.0	190.17.6.5	m1
255.255.0.0	135.9.0.0	190.17.6.5	m1
255.255.0.0	140.6.0.0	180.14.2.5	m2
0.0.0.0	0.0.0.0	110.70.4.6	m0



# Updating the Routing Table



RIP Message from  $R_B$

Dst.	Metric
NET1	1
NET3	1
NET4	3
NET5	2
NET6	2

RIP Message from  $R_B$   
after **increment**

Dst.	Metric
NET1	<del>1</del> 2
NET3	<del>1</del> 2
NET4	<del>3</del> 4
NET5	<del>2</del> 3
NET6	<del>2</del> 3

$R_A$  路由表

Dst.	Nexthop	Metric
NET1	-	1
NET2	$R_E$	2
NET3	$R_B$	2
NET4	$R_E$	2
NET5	$R_E$	3
NET6	$R_B$	2

$R_A$ 's RIP Message

Dst.	Metric
NET1	1
NET2	2
NET3	2
NET4	2
NET5	3
NET6	3

# Transport Layer Protocol

- Responsibilities
  - To create a *process-to-process communication*
    - 使用两个端点地址 ( IP + 端口号 ) 通信
  - To provide a *flow-control and error-control mechanism* at the transport layer
    - Error control: *ARQ → Window based control*
      - Acknowledgement packet
      - Time-out
      - Retransmission
    - Flow control: *sliding window protocol → Window based control*
  - To provide a *congestion control*
  - To provide a *connection mechanism* for the processes
- TCP: a *connection-oriented, reliable* transport protocol

# 拥塞举例

