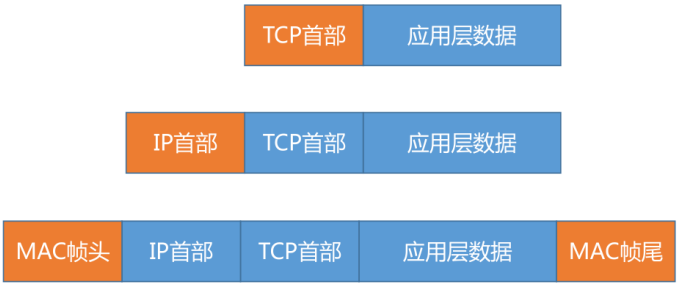
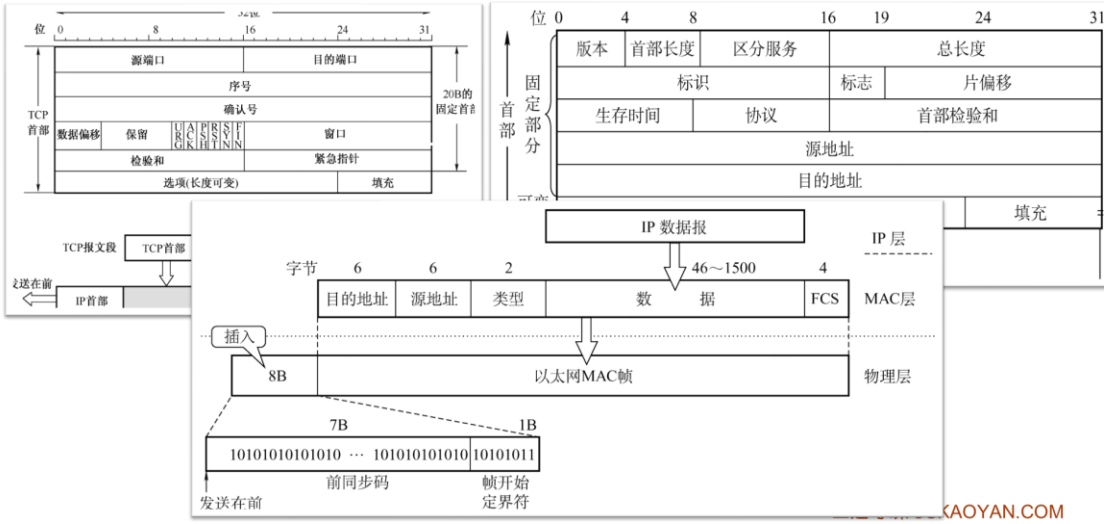
计算机网络

1.

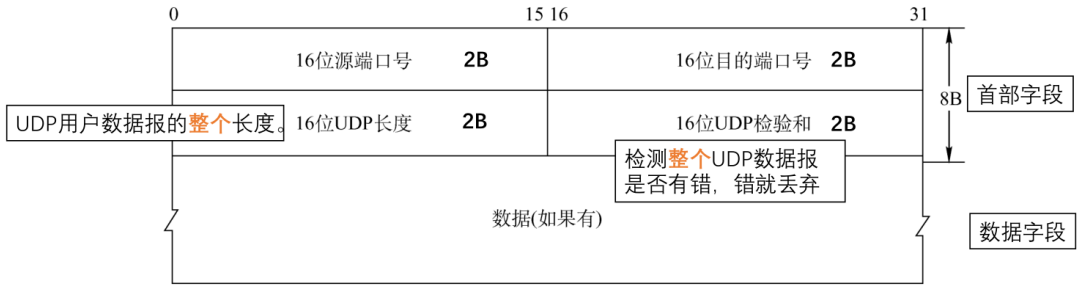


MAC帧首部+尾部=18B，数据部分为46B～1500B

TCP报文段、IP分组、MAC帧



UDP首部格式



分用时，找不到对应的目的端口号，就丢弃报文，并给发送方发送ICMP “端口不可达” 差错报告报文。

附：各报文段需要记忆的内容

☆	HTTP报文	HTTP报文分为请求报文&响应报文 请求报文： 1.请求行：请求方法（常用get/post）、请求URL、HTTP协议版本 2.首部行 3.请求体/实体主体 响应报文： 1.状态行 2.响应头部 3.响应体
☆☆	UDP数据报	1.首部8B，由4个字段组成（都是2B） 2.长度字段包括首部+数据部分 3.检验和检验首部+数据部分（可选）
☆☆☆	TCP报文段	1.首部固定部分为20B，最大值为60B（和IP分组一样） 2.源端口和目的端口各占2B 3.序号（本报文段第一个字节的序号）和确认号（期望收到下一个的序号）各占4B 4.数据偏移=首部长度（4B整数倍） 5.确认位ACK、同步位SYN、终止位FIN什么时候为0/1 6.窗口字段表示允许对方发送的数据量（流量控制用）
☆☆☆☆	IP分组	1.首部固定部分为20B，首部最大值为60B 2.总长度（1）+片偏移的单位（8）+首部长度（4）（“一种八片首饰”） 3.标志位MF和DF在分片时的取值 4.生存时间TTL，经过一个路由器减去1，直到为0 5.首部校验和字段只校验首部 6.源地址和目的地址字段长度都为4B
☆☆☆	MAC帧	1.前同步码8B 2.MAC地址长度6B 3.数据长度为46-1500B，首部和尾部是18B，因此最短帧长64B。

IP地址：分类的IP地址

主机号全0：本主机所连接到的单个网络地址
主机号全1：该网络上的所有主机(广播地址)

网络类别	最大可用网络数	第一个可用的网络号	最后一个可用的网络号	每个网络中的最大主机数
A	2^7-2	1	126	2^{24}-2
B	2^{14}-1	128.1	191.255	2^{16}-2
C	2^{21}-1	192.0.1	223.255.255	2^8-2

特殊IP地址

NetID 网络号	HostID主机号	作为IP分组源地址	作为IP分组目的地址	用途
全0	全0	可以	不可以	本网范围内表示主机，路由表中用于表示默认路由（表示整个Internet网络）
全0	特定值	可以	不可以	表示本网内某个特定主机
全1	全1	不可以	可以	本网广播地址（路由器不转发）
特定值	全0	不可以	不可以	网络地址，表示一个网络
特定值	全1	不可以	可以	直接广播地址，对特定网络上的所有主机进行广播
127	任何数（非全0/1）	可以	可以	用于本地软件环回测试，称为环回地址 (但现在基本只用127.0.0.1)

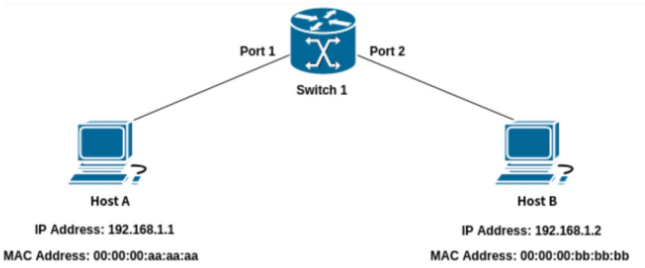
特殊IP地址 - 私有IP地址

地址类别	地址范围	网段个数
A类	10.0.0.0~10.255.255.255	1
B类	172.16.0.0~172.31.255.255	16
C类	192.168.0.0~192.168.255.255	256

路由器对目的地址是私有IP地址的数据报一律不进行转发。

2.

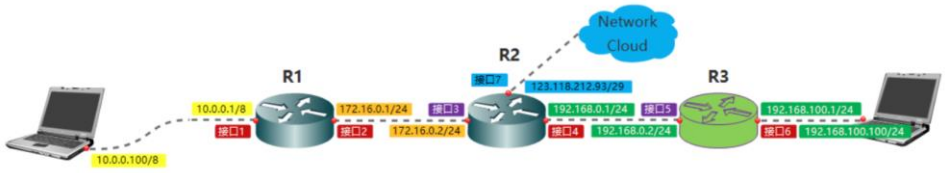
ARP表建立过程



检查**ARP高速缓存**，有对应表项则写入MAC帧，没有则用目的MAC地址为FF-FF-FF-FF-FF-FF的帧封装并广播ARP请求分组，**同一局域网中**所有主机都能收到该请求。目的主机收到请求后就会向源主机单播一个ARP响应分组，源主机收到后将此映射**写入ARP缓存**（10-20min更新一次）。

3.路由表

路由器 R3 路由表



网络ID	子网掩码	接口	网关
10.0.0.0	255.0.0.0	5	192.168.0.1
172.16.0.0	255.255.0.0	5	192.168.0.1
192.168.0.0	255.255.255.0	5	-
192.168.100.0	255.255.255.0	6	-
0.0.0.0	0.0.0.0	5	192.168.0.1

4.

CSMA/CD协议

载波监听多点接入/碰撞检测CSMA/CD（carrier sense multiple access with collision detection）

多点接入说明是总线型网络，计算机以多点接入的方式连接在一根总线上，协议的实质是"载波监听"和"碰撞检测"。

载波监听就是利用电子技术检测总线上有没有其他计算机也在发送。载波监听实际上就是检测信道。在发送前，每个站不停地检测信道，是为了获得发送权；在发送中检测信道，是为了及时发现有没有其他站的发送和本站发送的碰撞，这就是碰撞检测。总之，载波监听是全程都在进行的。

碰撞检测就是边发送边监听。就是网卡边发送数据边检测信道上的信号电压的变化情况，以便判断自己在发送数据的时候其他站是否也在发送数据。当几个站同时在总线上发送数据时，总线上的信号电压变化幅度将会增大（互相叠加），当网卡检测到的信号电压超过一定的门限值时，说明总线上至少有两个站同时在发送数据，表明产生了碰撞（冲突），所以也称为冲突检测。这时，由于接收的信号已经识别不出来，所以任何一个正在发送的站就会立即停止发送数据，然后等待一段随机事件以后再次发送。

传播时延对碰撞检测的影响：
因为网卡只有在接收到电压幅度不正常的信号以后才能判断是否产生了冲突，所以它在接收到信号之前会认为信道是空闲的。

5.

DHCP协议

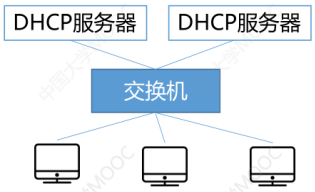
动态主机配置协议DHCP是**应用层**协议，使用**客户/服务器**方式，客户端和服务端通过**广播**方式进行交互，基于**UDP**。
DHCP提供**即插即用**联网的机制，主机可以从服务器动态获取IP地址、子网掩码、默认网关、DNS服务器名称与IP地址，允许**地址重用**，支持**移动用户加入网络**，支持**在用地址续租**。

1.主机广播DHCP**发现**报文　"有没有DHCP服务器呀？"　试图找到网络中的服务器，服务器获得一个IP地址。

2.DHCP服务器广播DHCP**提供**报文　"有！""有！""有！"　服务器拟分配给主机一个IP地址及相关配置，先到先得。

3.主机广播DHCP**请求**报文　"我用你给我的IP地址啦？"　主机向服务器请求提供IP地址。

4.DHCP服务器广播DHCP**确认**报文用吧！"　正式将IP地址分配给主机。



6.

ICMP差错报告报文（5种）

1.终点不可达：当路由器或主机不能交付数据报时就向源点发送终点不可达报文。
无法交付
2.源点抑制：当路由器或主机由于拥塞而丢弃数据报时，就向源点发送源点抑制报文，使源点知道应当把数据报的发送速率放慢。拥塞丢数据

3.时间超过：当路由器收到生存时间TTL=0的数据报时，除丢弃该数据报外，还要向源点发送时间超过报文。当终点在预先规定的时间内不能收到一个数据报的全部数据报片时，就把已收到的数据报片都丢弃，并向源点发送时间超过报文。TTL=0

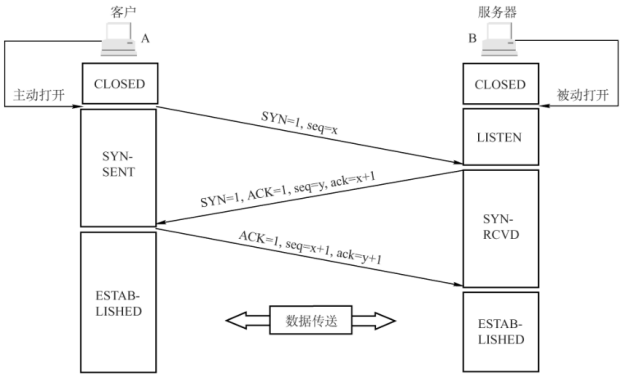
4.参数问题：当路由器或目的主机收到的数据报的首部中有的字段的值不正确时，就丢弃该数据报，并向源点发送参数问题报文。首部字段有问题

5.改变路由（重定向）：路由器把改变路由报文发送给主机，让主机知道下次应将数据报发送给另外的路由器（可通过更好的路由）。　值得更好的路由

7.TCP 关键

TCP的连接建立

假设运行在一台主机（客户）上的一个进程想与另一台主机（服务器）上的一个进程建立一条连接，客户应用进程首先通知客户TCP，他想建立一个与服务器上某个进程之间的连接，客户中的TCP会用以下步骤与服务器中的TCP建立一条TCP连接：



ROUND 1:

客户端发送**连接请求报文段**，无应用层数据。

SYN=1，seq=x(随机)

ROUND 2:

服务器端为该TCP连接**分配缓存和变量**，并向客户端返回**确认报文段**，允许连接，无应用层数据。

SYN=1，ACK=1，seq=y(随机)，ack=x+1

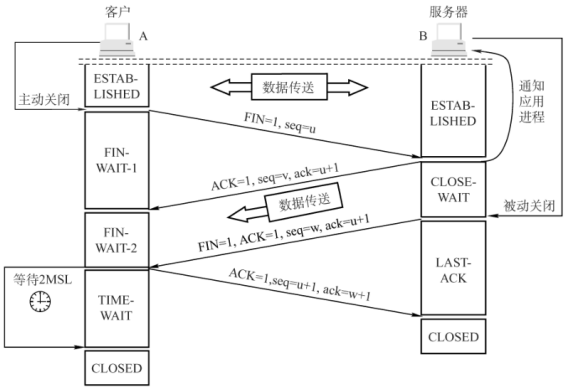
ROUND 3:

客户端为该TCP连接**分配缓存和变量**，并向服务器端返回确认的确认，可以携带数据。

SYN=0，ACK=1，seq=x+1，ack=y+1

TCP的连接释放

参与一条TCP连接的两个进程中的任何一个都能终止该连接，连接结束后，主机中的“资源”（缓存和变量）将被释放。



ROUND 1:

客户端发送**连接释放报文段**，停止发送数据，主动关闭TCP连接。

FIN=1，seq=u

ROUND 2:

服务器端回送一个确认报文段，客户到服务器这个方向的连接就释放了——半关闭状态。

ACK=1，seq=v，ack=u+1

ROUND 3:

服务器端发完数据，就发出连接释放报文段，主动关闭TCP连接。

FIN=1，ACK=1，seq=w，ack=u+1

ROUND 4:

客户端回送一个确认报文段，再等到时间等待计时器设置的2MSL（最长报文段寿命）后，连接彻底关闭。 ACK=1，seq=u+1，ack=w+1

8.端口号

应用程序	FTP	TELNET	SMTP	DNS	TFTP	HTTP	SNMP
熟知端口号	21	23	25	53	69	80	161

9.访问网站过程

解析：过程如下，

1. 浏览器首先通过DNS域名解析到服务器IP地址。
2. 浏览器接着查询ARP缓存，查询服务器IP地址对应的MAC地址。
   a. 如果缓存命中，则返回结果：目标IP地址——MAC地址；
   b. 如果没有命中：
      i. 查看本机维护路由表（见下图），看目标IP地址是否在本地路由表中的某个子网内：是则使用目标IP地址，否则使用默认网关的IP地址；
      ii. 查询选择的网络接口IP地址的MAC地址：发送一个数据链路层的广播ARP请求分组，该网段内都可以收到这个广播分组，但只有对应网关路由器接口才会返回一个ARP单播响应分组，将MAC地址回传。
3. 找到MAC地址后，便找到了下一跳，数据就可以转发到网关，依此类推，客户机就可以通过TCP/IP协议建立到服务器的TCP连接。
4. 客户端向服务器发送HTTP协议请求包，请求服务器里的资源文档。
5. 服务器向客户机发送HTTP协议应答包，将资源返回给客户端。
6. 客户机与服务器断开，由客户端解释HTML文档，在客户端屏幕上渲染图形效果等。

在上述通信过程中，不同的编址方案发挥了各自的作用：

- 以太网 MAC 地址：用于在局域网中唯一标识网络设备，以太网帧使用 MAC 地址进行目的地和源地址的标识，用于在局域网中的数据链路层通信。
- IP 地址：用于在网络层进行主机之间的寻址和路由选择，确定数据的源和目的地，实现跨网络的通信。
- TCP 端口：用于在传输层标识应用程序或服务，TCP 报文段中的源端口和目的端口标识了发送和接收数据的应用程序。
- 域名：作为人类可读的网址，用于提供更友好的访问方式。域名通过 DNS 解析为 IP 地址，使浏览器能够直接与目标服务器创建连接。

10.

57. 某单位获得一个 210.34.0.*的 C 类地址段，该单位的 4 个部门各需要 30、15、16、2 台机器，请给出划分子网的方案，用 CIDR 表示法。

答：IP 地址为 256，假设划分的子网网络号为 N，主机号为 32-N，扣掉：网络地址 1 个（主机位全 0）、广播地址 1 个（主机位全 1）、路由器地址 1 个，剩余 2^{32-N}-3 个。故此：（多个答案）

(1)第一部门，30 台机器，使用 210.34.0.0/26（地址范围 210.34.0.0 至 210.34.0.63）。

(2)第二部门，15 台机器，使用 210.34.0.64/27（地址范围 210.34.0.64 至 210.34.0.95）。

(3)第三部门，16 台机器，使用 210.34.0.96/27（地址范围 210.34.0.96 至 210.34.0.127）。

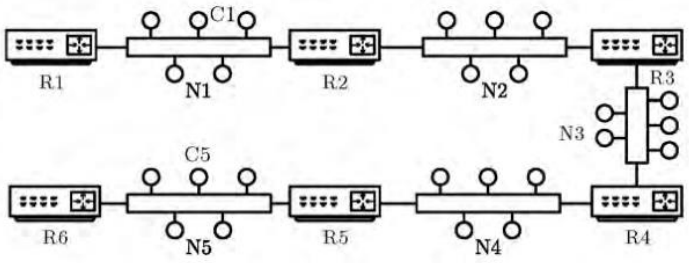
(4)第四部门，2 台机器，使用 210.34.0.128/27（地址范围 210.34.0.128 至 210.34.0.159）。

11. RIP (路由信息协议）使用何种传输层协议？请简述其工作原理。
答：RIP（Routing Information Protocol）使用传输层协议为 UDP（User Datagram Protocol）。RIP 是一种距离矢量路由协议，用于在一个自治系统（AS）内部的路由器之间交换路由信息。其工作原理如下：
1. 距离矢量更新：每个路由器维护一个路由表，其中包含了到达目标网络的距离矢量信息。初始时，路由器将自己直接连接的网络添加到路由表中，并将距离设置为 0。然后，路由器周期性地向相邻的路由器广播其完整的路由表，以及每个目标网络的距离。
2. 距离计算：当路由器接收到其他路由器广播的路由表时，它会比较当前路由表中的距离与接收到的距离信息。如果接收到的距离比当前路由表中的距离更短，则更新路由表中的距离，并将下一跳指向发送该路由信息的路由器。
3. 路由表更新：如果路由器的路由表发生了变化，它将向相邻的路由器发送更新的路由信息。这个过程会在整个网络中不断传播，直到所有的路由器都收敛到一个稳定的路由表。
4. 定时更新：RIP 使用定时器来控制路由表的更新频率。每隔一段时间，路由器会发送自己的路由表，以确保网络中的所有路由器都具有最新的路由信息。
12. **请简述 TCP 协议中流量控制机制，并指出流量控制与拥塞控制的区别。**
    - 答：流量控制（Flow Control）：用于控制发送方向接收方发送数据的速率，确保接收方能够有效处理接收到的数据，防止接收方被过量的数据淹没。TCP 使用滑动窗口机制来进行流量控制，接收方通过窗口大小告知发送方可以接收的数据量，发送方根据窗口大小调整发送速率。
    - 拥塞控制（Congestion Control）：用于控制网络中的拥塞情况，防止网络负载过大而导致的丢包和延迟增加。拥塞控制主要通过动态调整发送速率来适应当前网络的拥塞程度。TCP 使用拥塞窗口机制和拥塞避免算法来进行拥塞控制，根据网络的拥塞程度调整发送速率和拥塞窗口大小。

异同点：
- 流量控制和拥塞控制都是 TCP 协议中的控制机制，用于维护网络的稳定性和可靠性。它们都是通过调整发送方的发送速率来实现控制，但目标和触发机制不同。
- 流量控制是为了保证接收方能够有效处理数据，防止数据溢出和丢失，控制发送速率；而拥塞控制是为了避免网络拥塞，调整发送速率来适应当前网络的负载情况。
- 流量控制是点对点的机制，在发送方和接收方之间进行控制；而拥塞控制是针对整个网络的机制，通过网络中的路由器和拥塞信号来调整发送速率。
- 流量控制是根据接收方的接收能力来进行控制，控制范围相对较小；而拥塞控制是根据网络中的拥塞程度来进行控制，控制范围更广，涉及到整个网络的拥塞状态。

总之，流量控制和拥塞控制是 TCP 协议中的两个重要机制，用于维持网络的稳定和可靠性，但它们的目标和触发机制有所不同。
13.

75. 如图 14-2 (a)所示，从 C1 向 C5 发送一个 IP 报文（报文总长 24KB），其中 MTU 如图 14-2 (b)所示。请写出此 IP 报文经 N1、N2、N3、N4、N5，在 R6 处每个分片的大小及其偏移量。（提示：IP 报头为 20B。）



(a) 网络拓扑图

Net	Type	MTU	Net	Type	MTU	Net	Type	MTU
N1	FDDI	4325B	N2	802.11n	2346B	N3	Ethernet	1500B
N4	TokenRing	4464B	N5	802.11n	2346B			

规律为：共 23 分片，设第 i 分片 $(i=0,\cdots,22)$ 的头部长度为 h_i、数据长度为 d_i、偏移量为 f_i，单位为字节。则：$h_i=20\ (i=0,\cdots,22)$，

$$d_i=\begin{cases}1480, & i=4k,0\le k\le 5;\\ 840, & i=4k+1,0\le k\le 5;\\ 504, & i=4k+3,0\le k\le 4;\\ 716, & i=22.\end{cases}\qquad f_i=\begin{cases}1480k, & i=4k,0\le k\le 5.\\ 538k+185, & i=4k+1,0\le k\le 5;\\ 538k+290, & i=4k+2,0\le k\le 5;\\ 538k+475, & i=4k+3,0\le k\le 4.\end{cases}$$

解答：报文头部 20B，报文数据 24556B。

在第 N1 处，按 20B+4304B 划分。(大小,偏移)为(4304,0), (4304,538), (4304,1076), (4304,1614), (4304,2152), (3036,2690)。

在第 N2 处，按 20B+2320B 划分。(大小,偏移)为(2320,0), (1984, 290), (2320,538), (1984,828), (2320,1076), (1984,1366), (2320,1614), (1984,1904), (2320,2152), (1984,2442), (2320,2690), (716,2980)。

在第 N3 处，按 20B+1480B 划分。(大小,偏移)为(1480,0), (840,185), (1480, 290), (504, 475), (1480,538), (840,723), (1480,828), (504,1013), (1480,1076), (840,1261), (1480,1366), (504,1366), (1480,1614), (840,1799), (1480,1904), (504,2089), (1480,2152), (840,2337), (1480,2442), (504,2627), (1480,2690), (840,2875), (716,2980)。

在 N4、N5 处，同 N3 的情况。

报文分片重组的位置和条件：

- 报文分片重组发生在目标主机。
- 目标主机根据 IP 报文的标识符字段和偏移量字段将分片重新组装成完整的 IP 报文。
- 重组条件：目标主机收到的所有分片具有相同的标识符，并按照偏移量进行正确的排序和组装。

14. 请画出流程图说明 Socket API 在 Client-Server 模式中的执行模式。注意：分为面向连接和面向无连接的两种情况。

答：Socket 接口是应用程序的基本网络接口，由操作系统提供、进程的通信端点。Socket 包括一个五元组：协议类型，本地地址，本地端口号，远端地址，远端端口号。Socket-API 接口包括：socket()，bind()，listen()，accept()，send()/sendto()，recv()/recvfrom()，close()/closesocket()。

Socket 在 Client-Server 模式中的执行模式主要有两种：面向连接的和无连接的。其中，面向连接的 socket 过程如下左图，无连接的 socket 过程如下右图。注意下右图左半部分也可以用下左图的左半部分代替，但须注意此时两种模式下 connect 函数的作用完全不同。