浅谈TCP/IP

每个人每天都在依赖网络通信，那么这些设备之间究竟是如何通信的？我在这里写下这篇文章，你却能在千山万水之外阅读它。今天简要谈一下通信协议。

网络通信协议

网络通信协议主要是对信息的传输速率、传输代码、代码结构、传输控制步骤、差错控制等做出的规定并制订的标准。

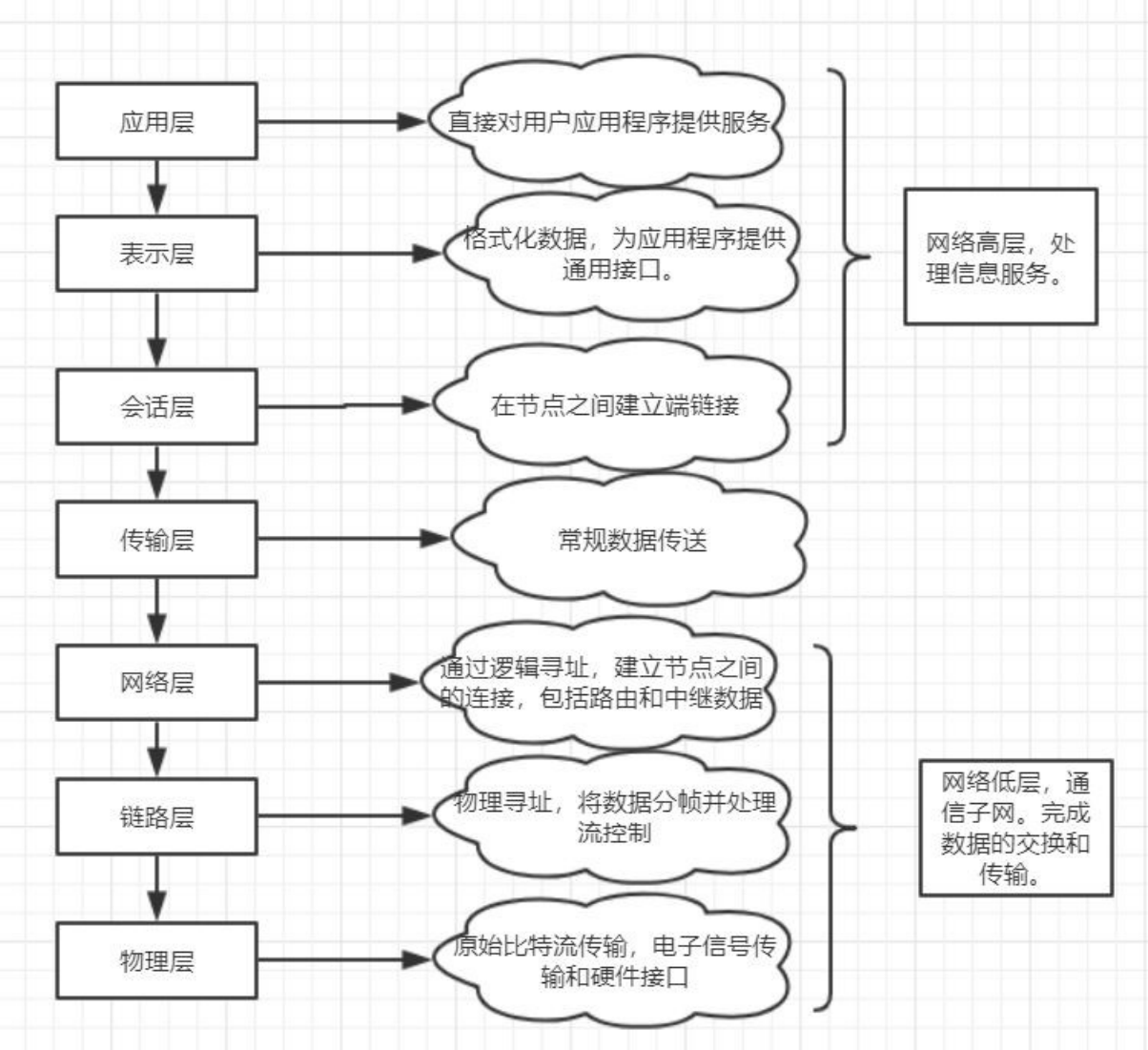
协议主要由以下三个部分组成：

- 语义：需要的控制信息以及执行的动作
- 语法：交换的数据或控制信息的格式与结构
- 时序：双方的应答关系，包括速度的匹配和顺序

OSI参考模型

OSI不是规范，准确的来说，是一个抽象的参考模型，他没有提供任何具体的实现标准。现有网络大多数可以通过OSI模型来进行分析，了解OSI模型有助于分析和管理网络。

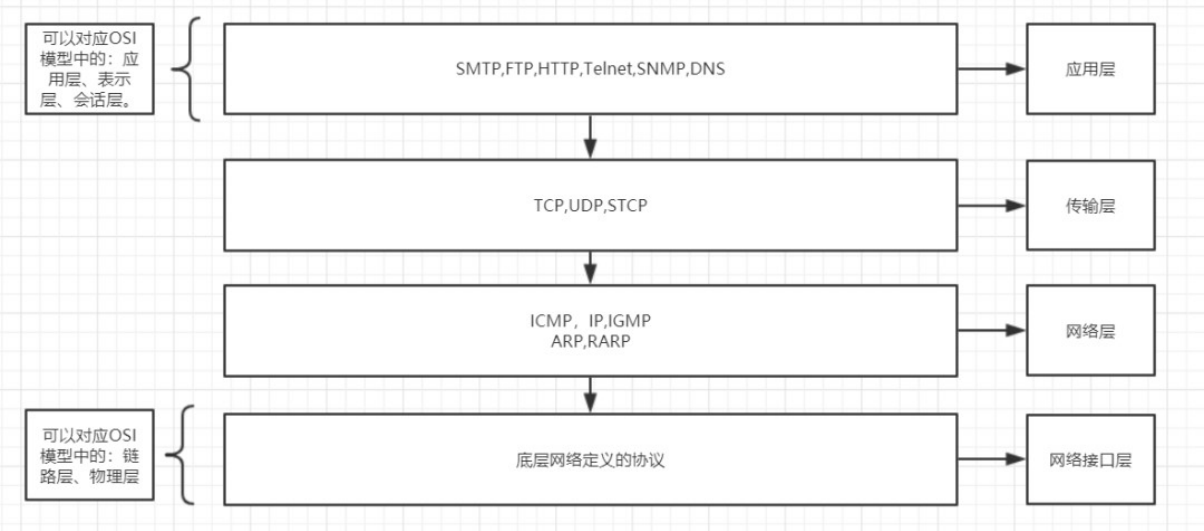
OSI是一个分层结构，共有七层：



TCP/IP协议簇

介绍

TCP/IP协议簇先于OSI参考模型开发，因此层次无法完全和OSI模型对应起来。他将不同的通信功能集成到不同的网络层次，形成了一个具有四层的体系结构。



有人也将其分为5层，将链路层从网络接口层剥离出来。

各层的职责

- 网络接口层：主要负责向网络物理介质发送/接收数据包。因为TCP/IP没有对物理层和链路层进行定义，所以它支持各种底层的网络技术和标准。

- 网络层：负责处理IP数据包的传输、路由选择、流量控制和拥塞控制。ARP/RARP协议用于将IP地址与底层物理地址进行相互转换。IP协议既是网络层的核心协议，也是整个TCP/IP协议簇的核心协议。

- 传输层：为两台主机提供端到端的通信。传输层主要包括传输控制协议TCP（提供可靠的面向连接的传输服务），用户数据报协议UDP（简单高效的无连接的服务）。可以根据上层应用的实际需求来选择

- 应用层：直接为特定的应用程序提供服务，如文件传输协议FTP，简单邮件传输协议SMTP，超文本传输协议HTTP。

重要概念

面向连接与无连接：

想要通过面向连接的协议在两台主机之间通信，需要两台主机之间首先建立连接，才能通信。如何建立/断开连接？这就涉及到三次握手和四次挥手。后续再讲。

而无连接的协议在通信前不需要建立连接，就像寄信，只需要知道目的地地址就行了（请注意，这只是一个比喻，发邮件并不是使用的无连接协议，因为无连接的协议通常是不可靠的）

可靠与不可靠

可靠的协议保证数据能传输到目的地，而且内容不会发生变化。TCP就是一种可靠的协议。

不可靠的协议不能保证将数据传送到目的地，但是它会尽力而为，还会检验送到目的地的数据是否完整。UDP就是一种不可靠的协议。

那么有人可能会说了？既然有了可靠的协议，还需要不可靠的协议干什么？不是多此一举吗？非也，下面详细说说TCP和UDP的优缺点。

字节流与数据报

字节流协议表示可以将发送方传输给接收方的数据看作是字节流。先发出的数据将会被先接收到。TCP属于字节流协议。

数据报协议是将数据一个个传送，没有顺序。UDP就是一种数据报协议。

套接字（Socket）

网络层中，IP利用协议号来指定传输协议，传输层中，TCP/UDP使用端口号来区分应用程序。将一个IP地址和一个端口号结合，就形成了一个套接字（也称为插座），套接字用来标明网络中的唯一网络进程。

TCP与UDP的优缺点

TCP：

- 优点： 可靠，稳定 TCP的可靠体现在TCP在传递数据之前，会有三次握手来建立连接，而且在数据传递时，有确认、窗口、重传、拥塞控制机制，在数据传完后，还会断开连接用来节约系统资源。

- 缺点：慢，效率低，占用系统资源高，易被攻击 TCP在传递数据之前，要先建连接，这会消耗时间，而且在数据传递时，确认机制、重传机制、拥塞控制机制等都会消耗大量的时间，而且要在每台设备上维护所有的传输连接，事实上，每个连接都会占用系统的CPU、内存等硬件资源。 而且，因为TCP有确认机制、三次握手机制，这些也导致TCP容易被人利用，实现DOS、DDOS、CC等攻击。

UDP：

- 优点： 快，比TCP稍安全 UDP没有TCP的握手、确认、窗口、重传、拥塞控制等机制，UDP是一个无状态的传输协议，所以它在传递数据时非常快。没有TCP的这些机制，UDP较TCP被攻击者利用的漏洞就要少一些。但UDP也是无法避免攻击的，比如：UDP Flood攻击。

- 缺点：不可靠，不稳定 因为UDP没有TCP那些可靠的机制，在数据传递时，如果网络质量不好，就会很容易丢包。

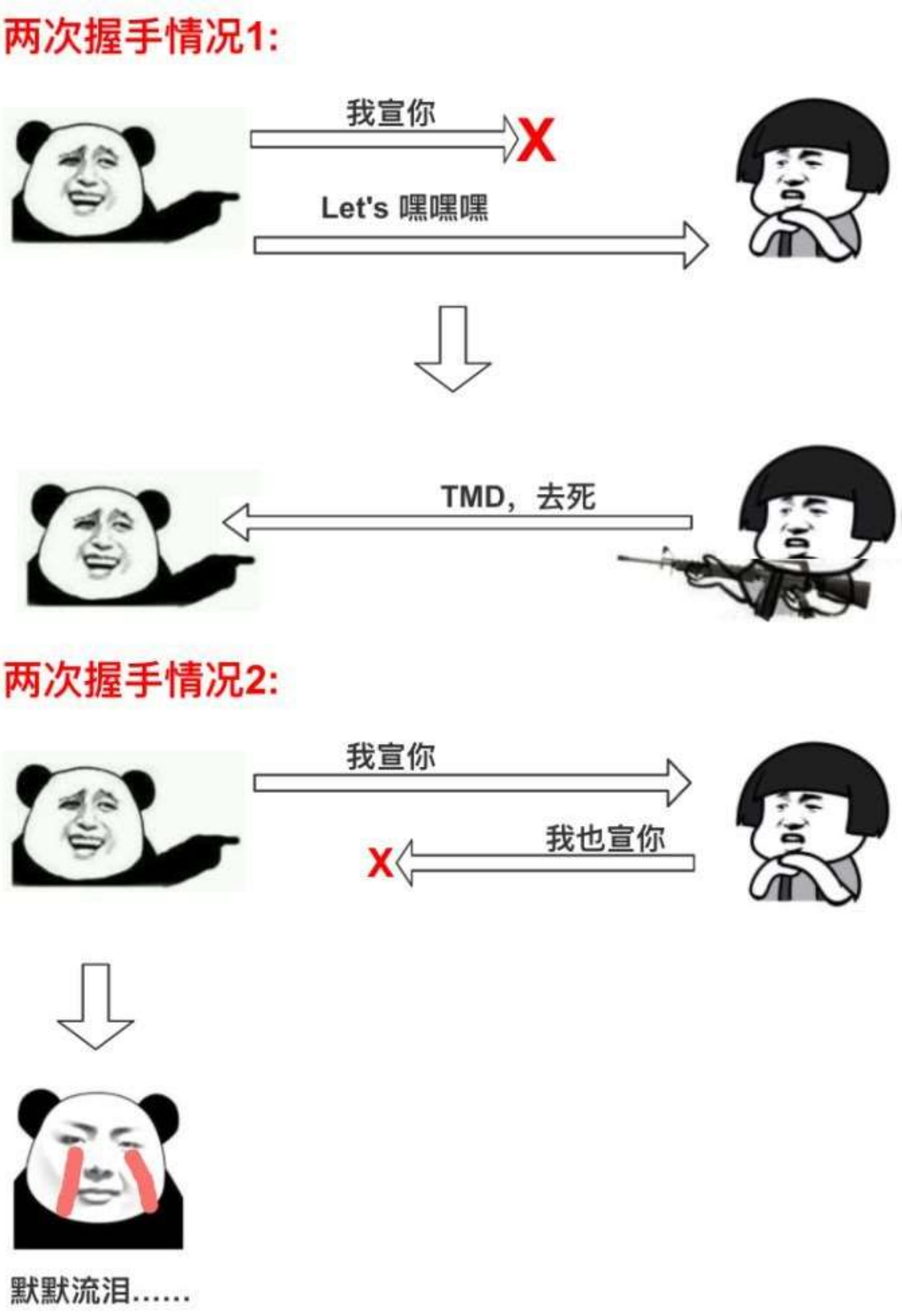
那么，哪些场景使用TCP，哪些场景使用UDP?

TCP：邮件，远程登录等等。UDP：NDS，广播，即时通讯，视频电话等等。

TCP的三次握手和四次挥手

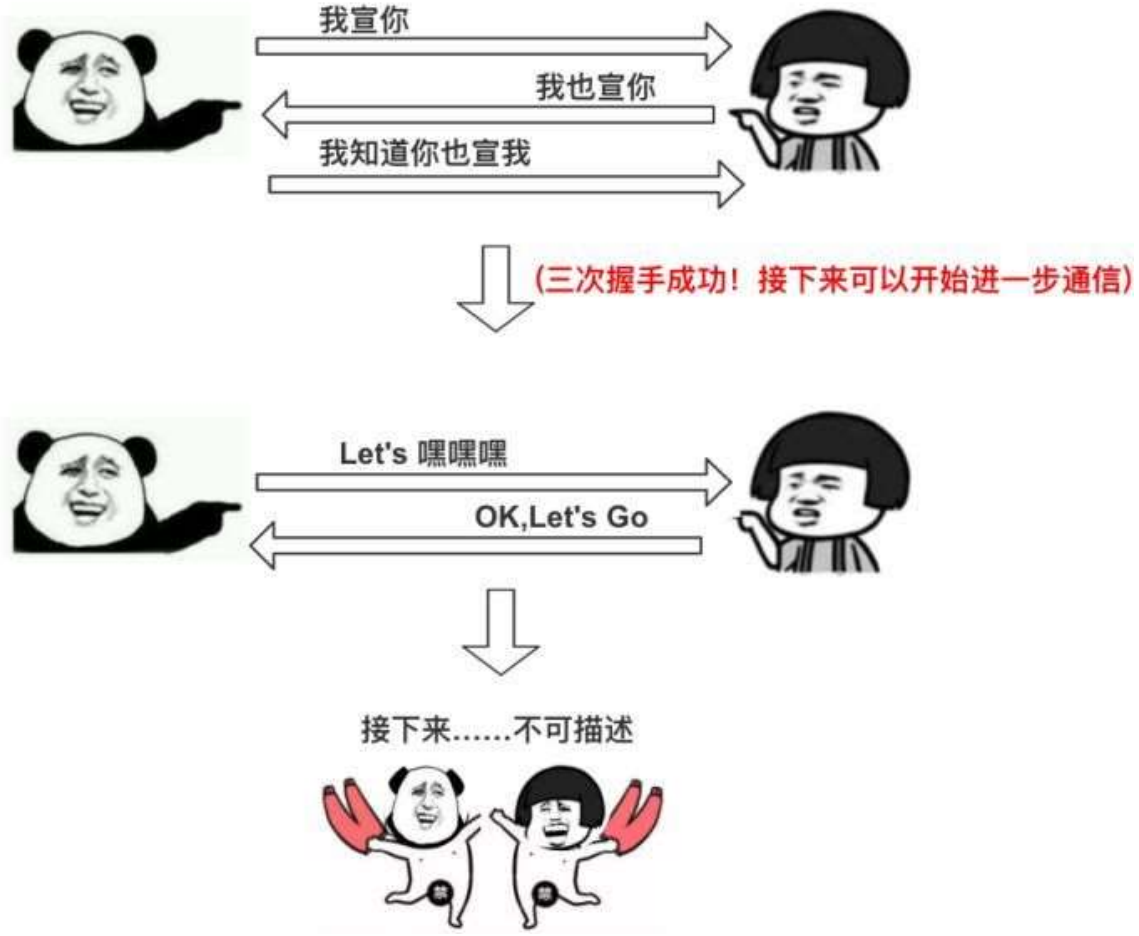
三次握手建立连接

TCP建立连接在理论上似乎只要一次请求和一次响应就可以了，但是在实际情况中，请求或者响应可能会丢失，此时需要重传来建立连接。假如只通过一次请求和一次响应，可能会出现以下问题（图片转自知乎用户@大闲人柴毛毛）：

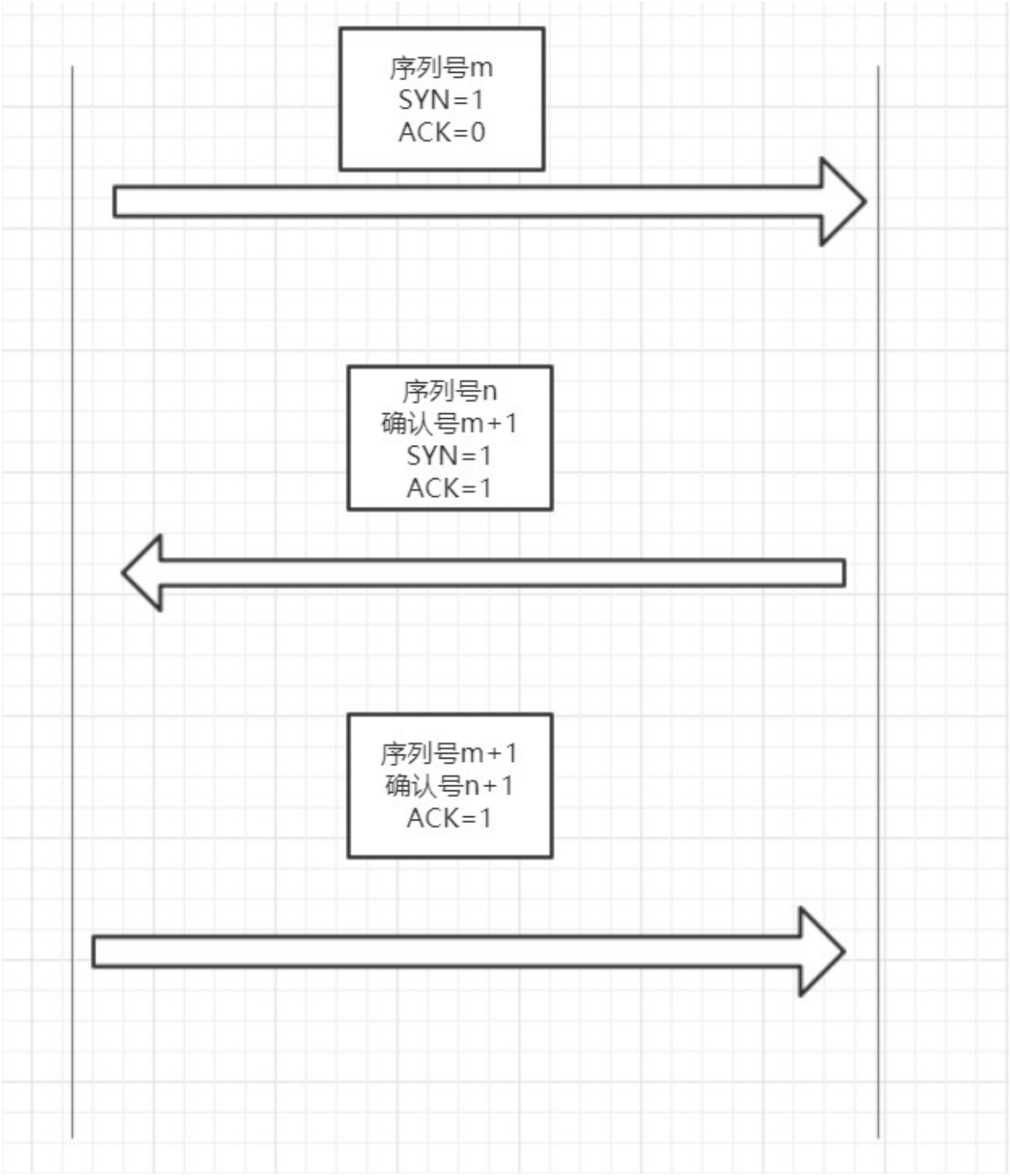


针对这个问题，连接时的“三次握手”可以有效解决。（其实无论握手多少次，都不能完全保证一条信道是完全可靠的，只能说明其是可用的，三次握手是能互相明确对方，同时开销是最小的，所以常用三次握手建立连接）

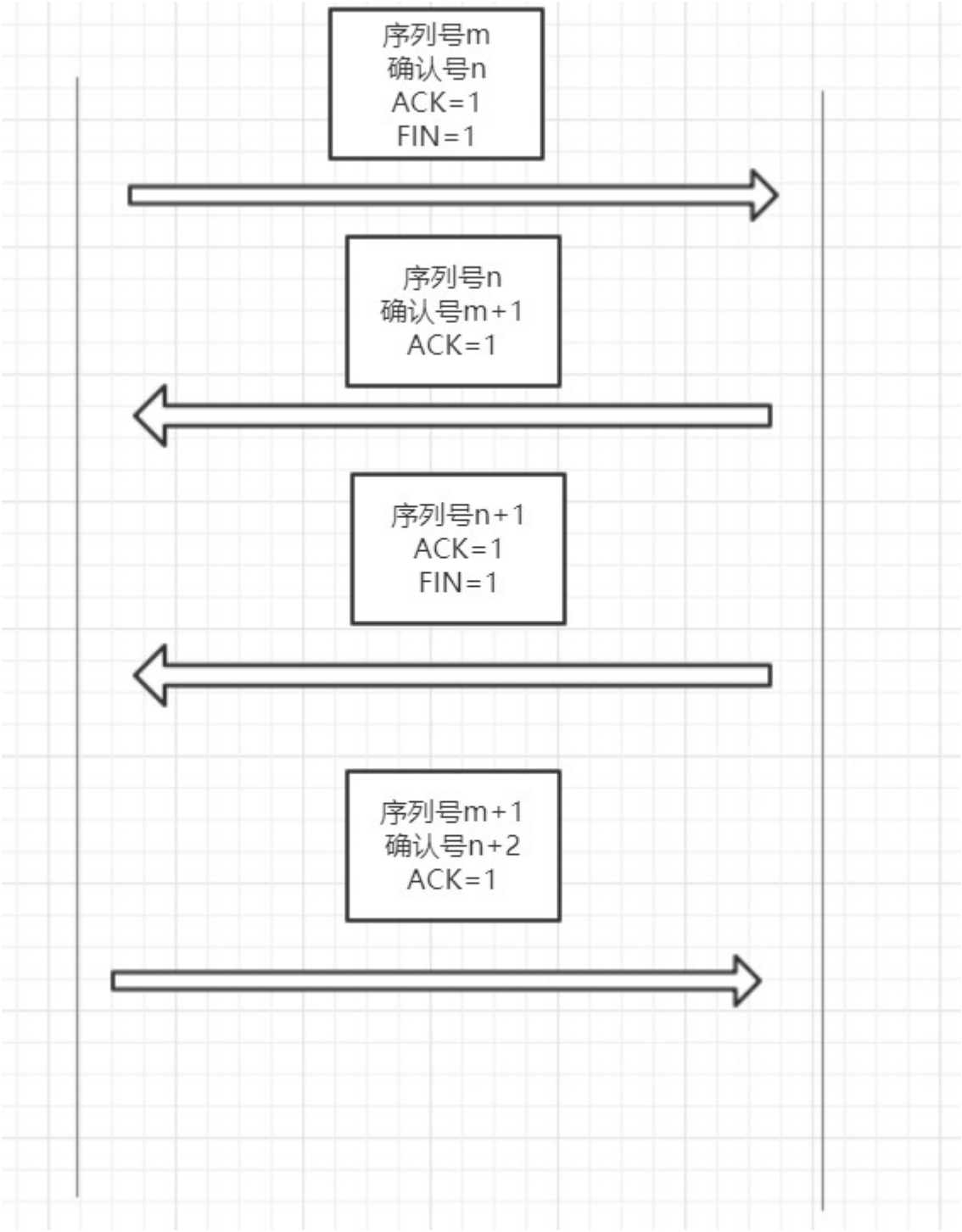
三次握手：



下面我还是画一张图来描述这个流程：



四次挥手断开连接



分层分析和排查故障

之前一次电话面试的时候，面试官问我：假如在调用第三方服务时，没有返回数据，可能是什么问题？我当时回答说通过返回的状态码来判断，他补充状态码也没返回。我一时语塞，不知道面试官想要考察什么，只好尴尬地跳过。（我比较菜）

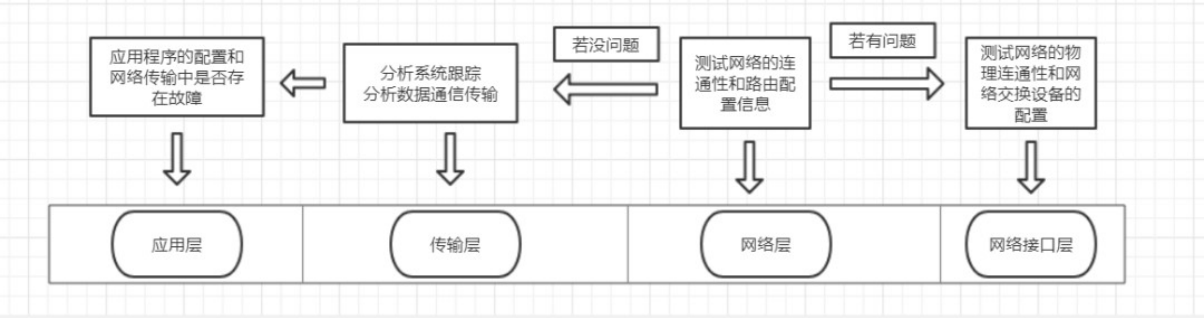
后来，才想到通过对协议簇的分层分析可以较为有效地排查故障。

一般有两种排查方式：

- 从低层开始，首先检查物理层，比如查看网络线缆是否松动损坏。这一般用于刚组建网络或者调整了网络线缆的情况。否则效率太低了。

- 从高层开始，首先检查应用层，比如查看浏览器是否正常配置，这一般用于网络环境比较稳定的情况下。

为了高效解决问题，在实际应用中，往往会从中间层开始检测，这似乎也有点像二分查找思想。



那么回到一开始的一个问题：

假如在调用第三方服务时，没有返回数据，可能是什么问题?

1、ping目标远程计算机

若成功则说明网络是正常的，可以去考虑更高层的事，需要去测试服务或应用程序。

若失败，则继续2.

2、ping同一子网的网关，确认正在使用的这台主机是否连接到了本地网络。

若成功，说明本地网关和远程目标计算机之间的连通有问题，可以跟踪测试路由。

若失败，则继续3.

3、ping环回地址127.0.0.1。

若成功，说明本地网关和当前计算机之间通信有问题。

若失败，检查IP是否有问题，若有问题，再检查本地TCP/IP协议软件是否有问题等等。

原创： 寒食君