

简答题：

1、为什么TCP需要三次握手？两次握手为什么不行？

- 三次握手的关键在于防止已过期的连接请求被再次传到主机
- 如果两次握手的话
 - 如果因为种种原因信号超时重传
 - 此时信号接收方已经接收到了重传的确认信号，通信结束
 - 再次因为种种原因信号此时超时的信号重新被接受
 - 此时接收方开始重新等待信息发送方发送信号
 - 但是由于发送方已经完成了超时重传，不再发送信号，造成死锁。

2、TCP断开连接时为什么是4次挥手？为什么主动断开的一方要经历TIME_WAIT状态。

- 为什么是四次挥手：
 - TCP协议的全双工机制，
 - 所以当**数据发送方**告诉**数据接收方**数据发送完毕，即发送方没有数据发送了
 - 此时**发送方**给**接收方**发送FIN，表示需要断开连接，自身进入FIN_WAIT1状态
 - **接收方**接受FIN请求，发送ACK，表示收到**发送方**信号，自身进入CLOSE_WAIT
 - **发送方**此时收到ACK信号，进入FIN_WAIT2状态
 - 这里是两次挥手
 - 但是此时**接收方**仍然可以向**发送方**发送信息。
 - 所以需要重复同样的过程（还需要两次挥手）。
 - 此时**数据接收方**发送FIN信号，自身进入LAST_WAIT
 - **数据发送方**接收到FIN信号之后发送ACK信号，自身进入TIME_WAIT状态
 - **数据接收方**接收到ACK信号，彻底关闭（CLOSE状态）
 - 加起来就是四次挥手。
 - **发送方**进入TIME_WAIT时：
 - 持续2倍MSL时长，在linux体系中大概是60s
 - 之后**发送方**彻底关闭（CLOSE状态）
- 为什么主动断开的一方要经历TIME_WAIT状态
 - 这为了TCP协议的可靠性：因为TCP协议是建立在不可靠网络上的可靠协议
 - 当主动关闭的一方收到被动关闭的一方发出的FIN包后，回应ACK包，同时进入TIME_WAIT状态
 - 由于网络和其他种种原因，ACK可能会发送失败，可能延迟，从而触发被动连接一方重传FIN包
 - 在极端情况下，这个时间最多可以是两倍的MSL时长（两方都要计算报文生存时间）
 - 如果此时主动关闭的一方跳过TIME_WAIT直接进入CLOSED，或者在TIME_WAIT停留的时长不足两倍的MSL，那么当被动关闭的一方早先发出的延迟包到达后（没有追回机制）可能会造成：

- TCP连接已经不存在了（因为已经关闭了），系统此时只能返回RST
- TCP连接还在运行阶段，延迟到来的包可能干扰此时的通信（造成死锁？）
- （这里没查具体会咋样，初步猜测是和两次握手一样，会造成死锁）

○ 无论是哪种情况都会让可靠的TCP协议不再可靠，所以TIME_WAIT状态有存在的必要性