

计算机网络----三次握手四次挥手

原创 haikuotiankong 程序员面试之道 2021-01-16 08:00

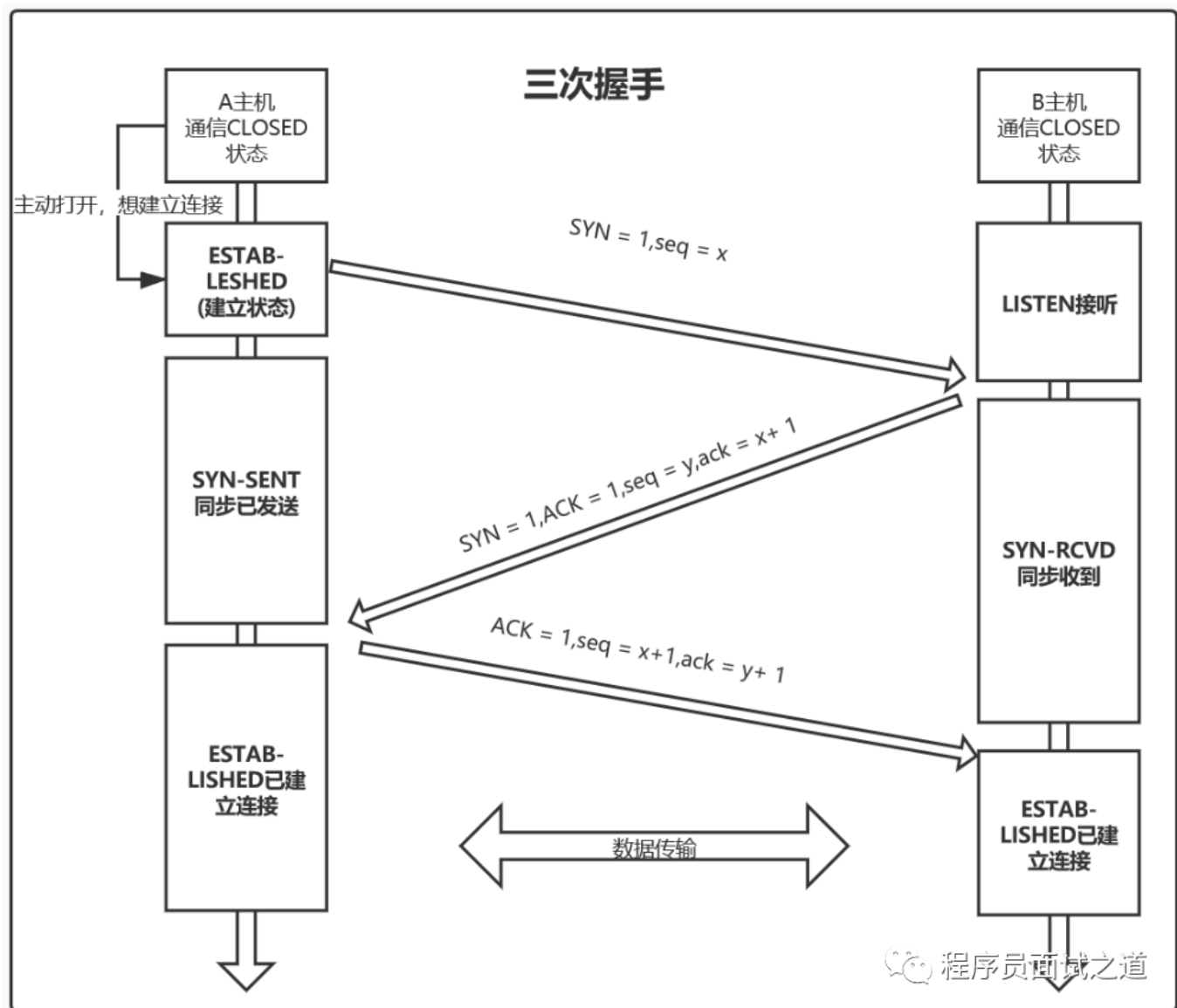
收录于合集

#面试必备计算机网络 5 #计算机网络 13



今天来彻底讲讲计算机网络中必考的知识点--三次握手，四次挥手。总结了所有的考点。后面有视频欧。

三次握手



四次挥手

专有名词讲解

SYN:(连接建立时用于同步序号SYN这个标志位只有在TCP建立连接才会被置为1，握手完成后SYN标志位被置为0)

seq:(标记数据段的顺序，就是这个报文段中的第一个字节的数据编号)

ACK:(确认号，仅当ACK=1，确认号字段才有效。ACK=0，确认号无效)

ack:(期待收到对方下一个报文段的第一个数据字节的序号)

MSL: (最大报文生存时间)

FIN: (用来释放一个连接)

1 为什么是三次握手 (一次两次行不行)

一次握手: 肯定建立不了连接,因为没法判定是否发送成功。

两次握手: 当A想要建立连接时发送一个SYN,然后等待ACK,结果这个SYN因为网络问题没有及时到达B,所以A在一段时间内没收到ACK后,再发送一个SYN,这次B顺利收到,接着A也收到ACK,这时A发送的**第一个SYN终于到了B (网络延迟)**,对于B来说这是一个新连接请求,然后B又为这个连接申请资源,返回ACK,然而这个SYN是个无效的请求,A收到这个SYN的ACK后也并不会理会它,而B却不知道,B会一直为这个连接维持着资源,造成资源的浪费。

三次握手: 可以保证任何一次握手的失败都是可感知的,不会浪费资源。

四次握手: 没有必要,三次足矣。

2 如果已经建立了连接,但是客户端突然出现故障了怎么办

客户端如果出现故障,服务器不能一直等下去,白白浪费资源。服务器每收到一次客户端的请求后都会**重新复位**这个计时器,时间通常是设置为2小时,若两小时还没有收到客户端的任何数据,服务器就会**发送一个探测报文段**,以后每隔75秒钟发送一次。若一连发送**10个探测报文仍然没反应**,服务器就认为客户端出了故障,接着就关闭连接。

3 Client发送一个SYN包给Server后就挂了，这个时候这个连接处于什么状态呢 ☐

Client发送SYN包给Server后挂了，Server回给Client的SYN-ACK**一直没收到Client的ACK确认**，这个时候这个连接既没建立起来这就会设定一个超时时间让Server将这个**连接断开**，否则这个连接就会一直占用Server的SYN连接队列中的一个位置，大量这样的连接就会将Server的SYN连接队列耗尽，让正常的连接无法得到处理。Linux下默认会进行5次**重发SYN-ACK包**，重试的间隔时间从1s开始，下次的重试间隔时间是前一次的双倍，5次的重试时间间隔为1s, 2s, 4s, 8s, 16s，总共31s，第5次发出后还要等32s都知道第5次也超时了，所以，总共需要 $1s + 2s + 4s + 8s + 16s + 32s = 63s$ ，TCP才会把断开这个连接。

4 三次握手带来的危害 ☐

由于，SYN超时需要63秒，那么就给攻击者一个攻击服务器的机会，攻击者在短时间内发送大量的SYN包给Server(俗称 **SYN flood 攻击**)，用于**耗尽Server的SYN队列**。

5 为什么需要四次挥手，三次挥手可以吗 ☐

TCP连接是双向传输的对等的模式，就是说双方都可以同时向对方发送或接收数据。当有一方要关闭连接时，会发送指令告知对方，我要关闭连接了。这时对方会回一个ACK，此时一个方向的连接关闭。**但是另一个方向仍然可以继续传输数据，等到发送完了所有的数据后，会发送一个FIN段来关闭此方向上的连接。**接收方发送ACK确认关闭连接。注意，接收到FIN报文的一方**只能回复一个ACK**，它是无法马上返回对方一个FIN报文段的，因为结束数据传输的“指令”是上层应用层给出的，我只是一个“搬运工”，我无法了解“上层的意志”。

6 为什么Time_wait (2MSL)

2MSL是任何报文在网络上的存在的最长时间，超过这个时间报文将被丢弃。这里的2MSL是时间等待计时器设置的。

2MSL可以使本连接持续的时间内所产生的所有报文段都从网络中消失。这样就可以使不一个新的连接中不会出现这种旧的连接请求报文段。

如果**小于2MSL**，第一个连接结束，第二个拥有相同的连接出现，而第一个连接的重复报文出现，干扰第二个连接。就会出现某个连接的重复报文在连接终止后出现。

7 关闭连接时最后一个ACK丢失了怎么办

如果最后一个ACK丢失的话，**TCP**就会认为它的**FIN**丢失，进行**重发FIN**。在客户端收到FIN后，就会设置一个2MSL计时器，2ML计时器可以使客户等待足够长的时间，使得在ACK丢失的情况下可以等到下一个FIN的到来。如果在TIME—WAIT状态中有一个新的FIN到达了，客户就会发送一个新的ACK，并重新设置2MSL计时器。

如果重传FIN到达客户端时，客户端已经进入 **CLOSED**状态时，那么客户就永远收不到这个重传的FN报文段，服务器收不到ACK，服务器无法关闭连接。

8 如果网络连接中出现大量TIME_WAIT状态会有什么危害

如果系统中有很多socket处于TIME_WAIT状态,当需要创建新的 socket连接的时可能会受到影响,这也会影响到**系统的扩展性**。之所以TIME_WAIT能够影响系统的扩展性是因为在一个TCP连接中,**一个Socket如果关闭的话,它将保持TIME_WAIT状态大约1-4分钟**。如果很多连接快速的打开和关闭的话系统中处于TIME_WAIT状态的socket将会积累很多,由于本地端口数量的限制,**同一时间只有有限数量的 socket连接可以建立**,如果太多的 socket处于TIME_WAIT状态你会发现,由于用于新建连接的本地端口太缺乏将会很难再建立新的。

视频版本 上

05:51

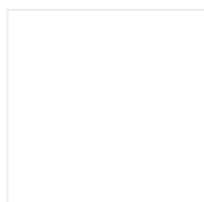
视频版本 下

08:25

点我留言

点我留言

原创面试复习图（点击菜单获取）



关注公众号“程序员面试之道”

回复“面试”获取面试一整套大礼包!!!

回复“面试”获取面试一整套大礼包!!!

点我留言

收录于合集 #面试必备计算机网络 5

上一篇

程序员必考---http\https的连接过程

下一篇

梦想成真---Socket技术详解