虽然 HTTP/2 解决了很多之前旧版本的问题，但是它还是存在一个巨大的问题，主要是底层支撑的 TCP 协议造成的。

上文提到 HTTP/2 使用了多路复用，
一般
来说同一域名下只需要使用一个 TCP 连接。
但当这个连接中出现了丢包的情况，那就会导致 HTTP/2 的表现情况反倒不如 HTTP/1 了。

因为在出现丢包的情况下，整个 TCP 都要开始等待重传，也就导致了后面的所有数据都被阻塞了。但是对于 HTTP/1.1 来说，可以开启多个 T
CP 连接，出现这种情况反到只会影响其中一个连接，剩余的 TCP 连接还可以正常传输数据。 那么可能就会有人考虑到去修改 TCP 协议，其实这已经是一件不可能完成的任务了，因为 TCP 存在的时间实在太长，已经充斥在各种设备中，并且这个协议是由操作系统实现的，更新起来不大现实。

基于这个原因，Google 就自己架起炉灶搞了一个基于 UDP 协议的 QUIC 协议，并且使用在了 HTTP/3 上，HTTP/3 之前名为 HTTP-over-QUIC，从这个名字中我们也可以发现，HTTP/3 最大的改造就是使用了 QUIC。 QUIC 虽然基于 UDP，但是在原本的基础上新增了很多功能，接下来我们重点介绍几个 QUIC 功能。

 QUIC 功能 0RTT 通过使用类似 TCP 快速打开的技术，缓存当前会话的上下文，在下次恢复会话的时候，只需要将之前的缓存传递给服务端验证通过就可以进行传输了。0RTT 建连可以说是 QUIC 相比 HTTP/2 最大的性能优势。那什么是 0RTT 建连呢？ 这里面有两层含义:

 1、传输层 0RTT 就能建立连接。

 2、加密层 0RTT 就能建立加密连接

|  |  |
| --- | --- |
| test | test |
| test | test |
| test | test |