|  |  |
| --- | --- |
| 实验名称：TCP协议分析 | |
| 实验台号： | 实验时间：2023/11/28 |
| 实验小组： | |
| 实验目的：  •了解TCP报文段的结构；  •掌握TCP数据流追踪的方法；  •理解TCP三次握手的基本过程；  •理解TCP连接终止的基本过程；  •认识TCP重置；  •理解TCP可靠数据传输的基本原理。 | |
| 实验环境说明：  wireshark-4.0.10 | |
| 实验过程、步骤（可另附页、使用网络拓扑图等辅助说明）及结果：  IMG_256 源端口 第一个字段是 **源端口**（ source port ），它的长度为 16 位，表示报文 **发送方** 的端口号。 目的端口 第二个字段是 **目的端口**（ destination port ），它的长度为 16 位，表示报文 **接收方** 的端口号。 序号 **序号**（ sequence number ）字段，长度为 32 位，表示数据首字节的序号。在三次握手阶段，SYN 指令也是通过该字段，将本端选定的 **起始序号** 告诉接收方。 确认号 **确认号** （ acknowledgement number ）字段，长度为 32 位。它表示已确认收到的数据序号，它的值为：已收到数据最后一个字节的序号加一，即接收方期望进一步接收的数据序号。 头部长度 **头部长度** （ header length ）字段，长度为 4 位，表示 TCP 报文头部的长度，也可称为 **数据偏移** （ data offset ）。跟 IP 协议一样，TCP 头部长度字段也不是以字节为单位，而是以 32 位字（4字节）为单位。 保留 **保留** （ reversed ）字段，长度为 3 位，保留未用。 标志位 **标志位**（ flags ），长度为 9 位，用于保存一些标志位。前面提到的 SYN ACK FIN 等指令，就是以标志位的形式保存在该字段中。这样的标志位，总共有 9 个：   * NS ，ECN显式拥塞通知，属于 TCP 扩展，略； * CWR ，同样属于 TCP 扩展，略； * ECE ，同样属于 TCP 扩展，略； * URG ，紧急数据指令，表示紧急指针有效，报文段包含高优先级数据； * ACK ，确认指令，表示确认号有效，对已接收数据进行确认； * PSH ，立即推送指令，指示接收方立即将数据提交给应用层，不用等缓冲区装满； * RST ，重置指令，表示出现严重错误，常用于拒绝非法报文段以及拒绝连接请求； * SYN ，序号同步指令，在 TCP 三次握手建立连接时，将本端选定序号告诉对端； * FIN ，连接关闭指令，用于告诉对端，本端数据已发送完毕，连接关闭；  窗口大小 **窗口大小**（ window size ）字段长度为 16 位，表示当前报文发送者接收窗口的大小，单位一般是 **字节** 。接收窗口表示接收方还能接收的数据大小，用于实现 TCP **流量控制** 机制. 校验和 **校验和**（ checksum ）字段长度为 16 位，保存报文段的校验和，用于纠错。跟 UDP 协议一样，TCP 整个报文段都会参与校验和计算。除此之外，TCP 还会在报文段前面拼接一个 IP 伪头部，同时参与校验和计算： 紧急指针 **紧急指针**（ urgent pointer ）字段长度为 16 位，仅在 URG 标志位开启时有效，它的值为紧急数据末字节，相对于当前报文段数据序号的偏移量。这意味着，该偏移量与序号字段相加即可得到紧急数据最后一个字节的序号。  URG 标志位和紧急指针一起为 TCP 提供了 **紧急模式** （ urgent mode ），以便在正常数据流中传输紧急数据。在套接字编程中，紧急数据经常被称为 **带外数据** （ out-of-band data ）。  因篇幅关系，紧急模式和带外数据本文不再展开介绍。对这个话题感兴趣的话，可以留意后续章节。 选项****选项****（ options ）字段包含一些可选记录，总长度最多可达 40 字节 **TCP数据流追踪**  选取http://www.hao123.com作为抓包地址  设置过滤规则  not udp and not ssl and not snmp and not icmp    **三次握手**  **mermaid-diagram-2023-12-03-221416**  **流程图**  **第一次握手**    SYN位为1指示要建立连接  同时seq会被随机赋一个初值（1654923206）  等待服务器响应  **第二次握手**    SYN和ACK位设置位1， ack设置为x+1（1654923207）seq被随机初始化为1728786954等待客户端的第三次握手  **第三次握手**    ACK为1 seq为x+1（1654923207）ack为（1728786955）完成第三次握手等待服务器传输数据    服务器和客户端之间开始传输数据  四次挥手  mermaid-diagram-2023-12-03-223842  示意图  当没有新的数据要发送的时候，就会释放 TCP 连接  客户端发送 FIN，用来关闭客户端到服务端的数据传送  当服务端收到 FIN 后，发送 ACK 给客户端  同时服务端向客户端发送 FIN，用来关闭服务端向客户端的数据传送  当客户端收到 FIN 后，发送一个 ACK 给 server 端    如图TCP 连接终止成功  **TCP重置**  TCP RESET是一种紧急的TCP数据包，通常在以下情况下使用：   * 当TCP连接的一方要强制终止连接时，TCP RESET包将被发送给连接的另一方，表示连接已被强制关闭。 * 当数据传输的选项被重置时，TCP RESET包可以将数据复位。 * 当TCP连接被阻塞或出现故障时，TCP RESET包可以使连接恢复正常。   TCP RESET包包含一个RST标志，一旦RST标志被设置，表明这是一个TCP RESET包，用于强制终止TCP连接。  **TCP 如何保证传输的可靠性？**  基于数据块传输：应用数据被分割成 TCP 认为最适合发送的数据块，再传输给网络层，数据块被称为报文段或段。  对失序数据包重新排序以及去重：TCP 为了保证不发生丢包，就给每个包一个序列号，有了序列号能够将接收到的数据根据序列号排序，并且去掉重复序列号的数据就可以实现数据包去重。  校验和 : TCP 将保持它首部和数据的检验和。这是一个端到端的检验和，目的是检测数据在传输过程中的任何变化。如果收到段的检验和有差错，TCP 将丢弃这个报文段和不确认收到此报文段。  超时重传 : 当发送方发送数据之后，它启动一个定时器，等待目的端确认收到这个报文段。接收端实体对已成功收到的包发回一个相应的确认信息（ACK）。如果发送端实体在合理的往返时延（RTT）内未收到确认消息，那么对应的数据包就被假设为已丢失并进行重传。  流量控制 : TCP 连接的每一方都有固定大小的缓冲空间，TCP 的接收端只允许发送端发送接收端缓冲区能接纳的数据。当接收方来不及处理发送方的数据，能提示发送方降低发送的速率，防止包丢失。TCP 使用的流量控制协议是可变大小的滑动窗口协议（TCP 利用滑动窗口实现流量控制）。  拥塞控制 : 当网络拥塞时，减少数据的发送。 | |
| 实验总结（遇到的问题及解决办法、体会）：  通过深入学习TCP协议，我更加深刻地理解了计算机网络中数据传输的复杂性和精密性。TCP作为一种可靠的传输协议，通过连接的建立、数据的传输和连接的终止等过程，确保了数据的可靠性和完整性。我认识到TCP协议在网络通信中的重要性，它不仅仅是一种协议，更是整个互联网通信的基石之一。  通过实际的分析操作，我学到了如何使用工具进行抓包分析。Wireshark等网络抓包工具让我可以直观地观察到TCP协议的各个阶段，包括握手、数据传输和连接的关闭。这让我对理论知识有了更为直观的认识，也提高了我对网络通信的实际应用的认识水平。  在实验中，我还意识到了网络通信中可能出现的一些问题，比如丢包、拥塞等，这些问题可能对通信的性能和效果产生重大影响。因此，学习如何分析和识别这些问题，以及采取相应的优化措施，对于成为一名优秀的网络工程师至关重要。 | |
| 器材、工具领用及归还负责人： | 实验记录人： |
| 实验执笔人： | 报告协助人：无 |
| 小组成员签名：无 | |
| 验收人： | 成绩评定： |