

**计算机网络**

**课程实验报告**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 实验名称 | 可靠数据传输协议—GBN协议的设计与实现 | | | | | |
| 姓名 | 田一间 | | 院系 | 计算机学院 | | |
| 班级 | 1636101 | | 学号 | 1160300617 | | |
| 任课教师 | 李全龙 | | 指导教师 | 李全龙 | | |
| 实验地点 | 格物楼213 | | 实验时间 | 2018年11月3日 | | |
| 实验课表现 | 出勤、表现得分(10) |  | 实验报告  得分(40) |  | 实验总分 |  |
| 操作结果得分(50) |  |
| 教师评语 | | | | | | |
|  | | | | | | |

****

|  |
| --- |
| 实验目的： |
| 理解滑动窗口协议的基本原理；掌握 GBN 的工作原理；掌握基于UDP 设计并实现一个 GBN 协议的过程与技术。 |
| 实验内容： |
| 1) 基于UDP设计一个简单的GBN协议，实现单向可靠数据传输（服务器到客户的数据传输）。  2) 模拟引入数据包的丢失，验证所设计协议的有效性。  3) 改进所设计的 GBN 协议，支持双向数据传输。  4) 将所设计的 GBN 协议改进为 SR 协议。 |
| 实验过程： |
| 1. 设计GBN协议数据分组格式   发送端：    共1026字节，其中Seq是序列号，1个字节，取值为0~255；Data为1024字节，存储数据；最末尾为0，表示结束。  接收端：    接收端为ACK的数据帧，不需附带任何数据，ACK字段为1个字节，表示序列号数值，其余字节为0。   1. 设计滑动窗口   发送窗口大小W为10，序列号个数N为20，满足W+1 <= N。   1. 设计数据分组丢失验证模拟方法   接收端采用一定概率值进行丢包与丢失ACK的模拟，每次收到包时或要发送ACK时，生成随机数，判断其是否在概率范围内以决定是否发生丢失。     1. GBN协议实现   **客户端：**  使用阻塞模式接收数据包，当接收到一个包时，使用随机概率判断该包是否需要模拟丢失。若不需丢失，则判断该包是否是期待接收的数据包，若是，则构建该序列号的Ack数据帧，若不是，则构建期待序列号的数据帧。接着使用随机概率判断该包是否需要模拟Ack丢失，若是，直接返回，若不是，则返回该Ack数据帧。  关键代码：    **服务端**：  当接收到客户端的协议测试指令后，读取文件构建分组数据包。判断当前窗口是否可以继续发送，若可以发送，则发送相应数据包，窗口内部进行相应下标调整。  使用非阻塞方式接收客户端返回的Ack，线程延迟作为计时，则可通过一个计数器来作为计时器。没有接收到确认Ack，则计数器加1，超时时触发超时重传事件。接收到确认的Ack，则窗口进行滑动，进入下一轮。  关键代码：    超时重传函数：    Ack处理：     1. SR协议实现   SR协议在GBN协议的基础上进行设计，接收方对每个分组单独进行确认，设置缓存机制，缓存乱序到达的分组。发送方为每个分组设置定时器，只重传那些没有收到ACK的分组。  **客户端：**    **服务端：**    超时重传函数：    Ack处理：     1. 双向传输实现   客户端和服务器使用两个端口进行传输和接收，双线程并发进行，以达到全双工双向文件传输的效果。  **服务器端：**    **客户端：**    其中的 revdThread 和 sendThread 即为接收和发送进程，客户端和服务端各自构建接收和发送的socket，发送使用非阻塞的DatagramChannel，接收使用阻塞的DataSocket，初始化后，根据用户命令选择运行相应的线程即可。  **初始化示例(客户端)：** |
| 实验结果： |
| 采用演示截图、文字说明等方式，给出本次实验的实验结果。  **GBN**：    从运行结果可以看出，接收方pkt3发生丢失，于是一直重复发送ack2，服务器端等到 pkt3 超时，进行重传，从pkt3依次重传窗口中的 pkt。稍后，接收方又发生了pkt6丢失，一直发送ack5，等到服务器pkt6超时，再次重传。  **SR：**    从结果可以看到，接收方pkt2丢失，再收到来自发送方的包时，进行了缓存。发送方等到pkt2超时时，只重新发送了该包，而接收方收到该包后，窗口直接滑动到pkt10，因为pkt10也发生了丢包事件。  **双向SR：**    双向SR协议的传输如上图，可以看到实现效果为全双工，服务器与客户端同时进行文件的发送与接收，且均符合SR协议的原理，在这里不再进行分析。  **双向GBN协议**：      其效果展示如上图，在这里不再进行具体的原理分析。 |
| 问题讨论： |
| 对实验过程中的思考问题进行讨论或回答。   1. 发送端程序流程图   C:\Users\26241\Desktop\CBN服务端.jpg   1. 接收端程序流程图   C:\Users\26241\Desktop\GBN客户端.jpg   1. 协议典型交互过程   GBN：  接收方某一数据包丢失时，会重复发送期待的Ack序列号。发送方在该丢失包超时时，重传窗口内从该包开始的所有包。  SR：  接收方某一数据包丢失后，不发送该包Ack，会缓冲发送方发来的后续数据包。发送方在该丢失包超时时，仅重传该数据包。 |
| 心得体会： |
| 结合实验过程和结果给出实验的体会和收获。  通过这次实验，自己对于GBN协议与SR协议有了更清晰的认识。一直认为自己这块掌握的还行，但是在真正代码实现时，才发现自己还是没有真正的理解其过程。  所幸这次实验帮助自己弥补了这个不足，尽管过程十分艰难，但是能够正确的实现这两个协议确实让自己受益匪浅。  不仅如此，在单向传输的基础上，成功实现GBN协议与SR协议的客户端与服务器的双向同时传输，增加了自己的自信心。  在这里，感谢老师的知识教导，也感谢助教的指点，感谢帮助我的人。 |