实验三 Socket 编程实现网络通信  
自02 彭程 2020011075

1. 实验目的

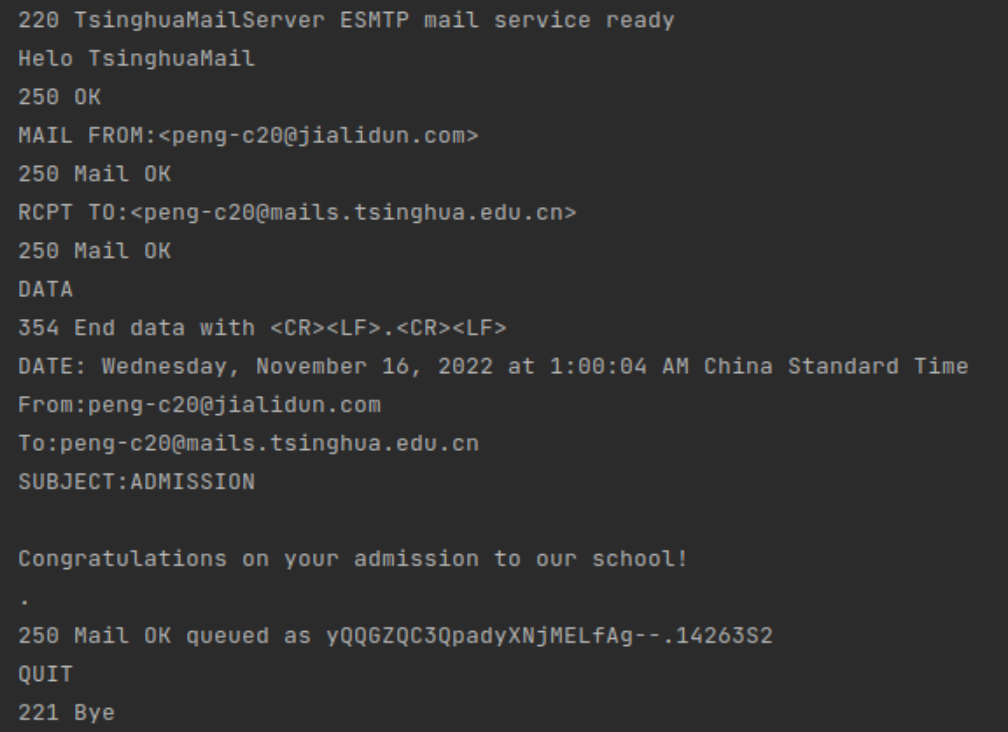
掌握 Socket 编程思想，并实现简单的 Socket 应用的连接通信过程。

1. 实验环境

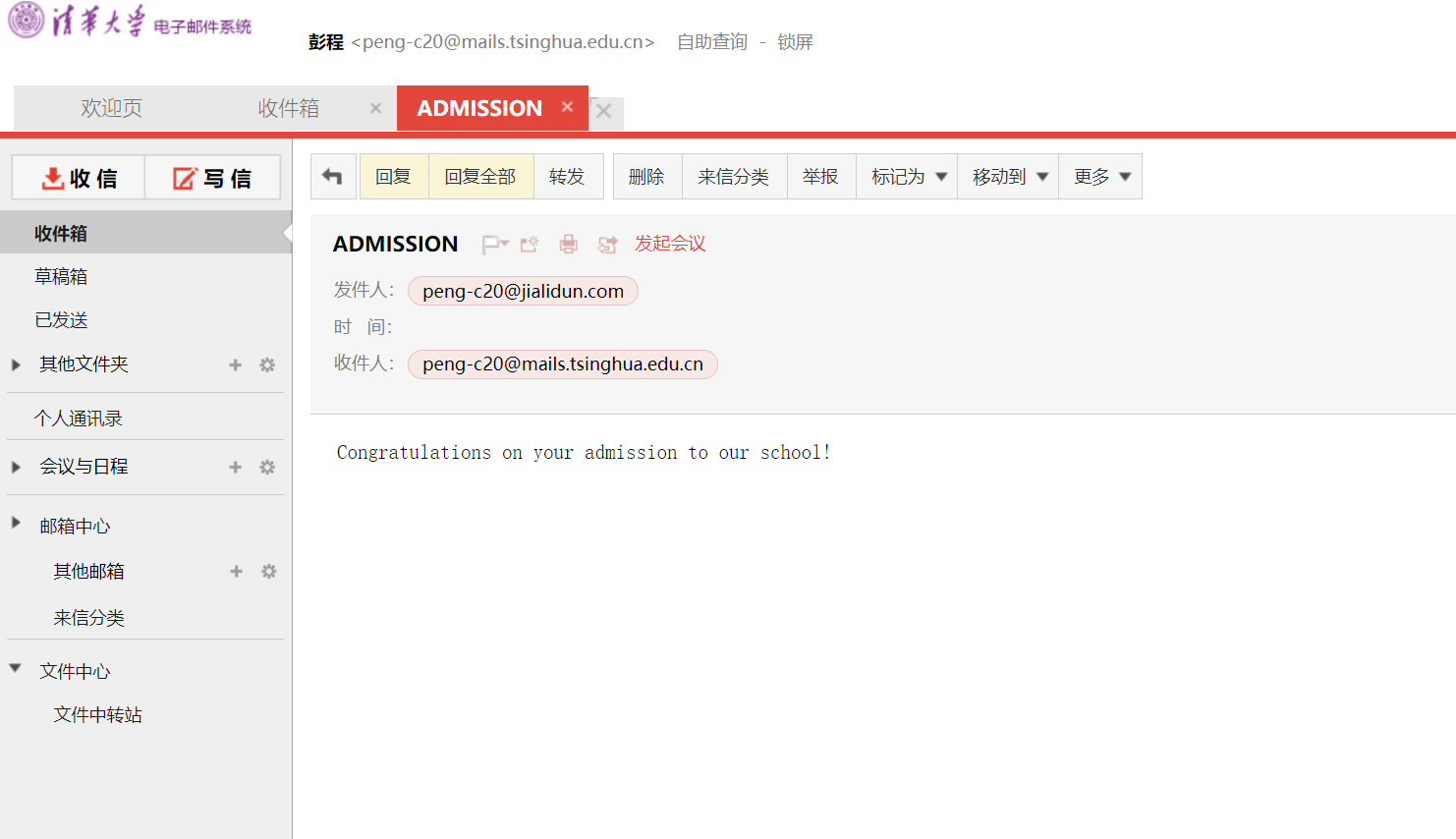
助教提供的虚拟机（Java环境），自己的主机（python环境）。

1. 实验内容
2. 不带服务器身份验证的邮件发送客户端实验（Simple SMTP）

控制台输出截图：

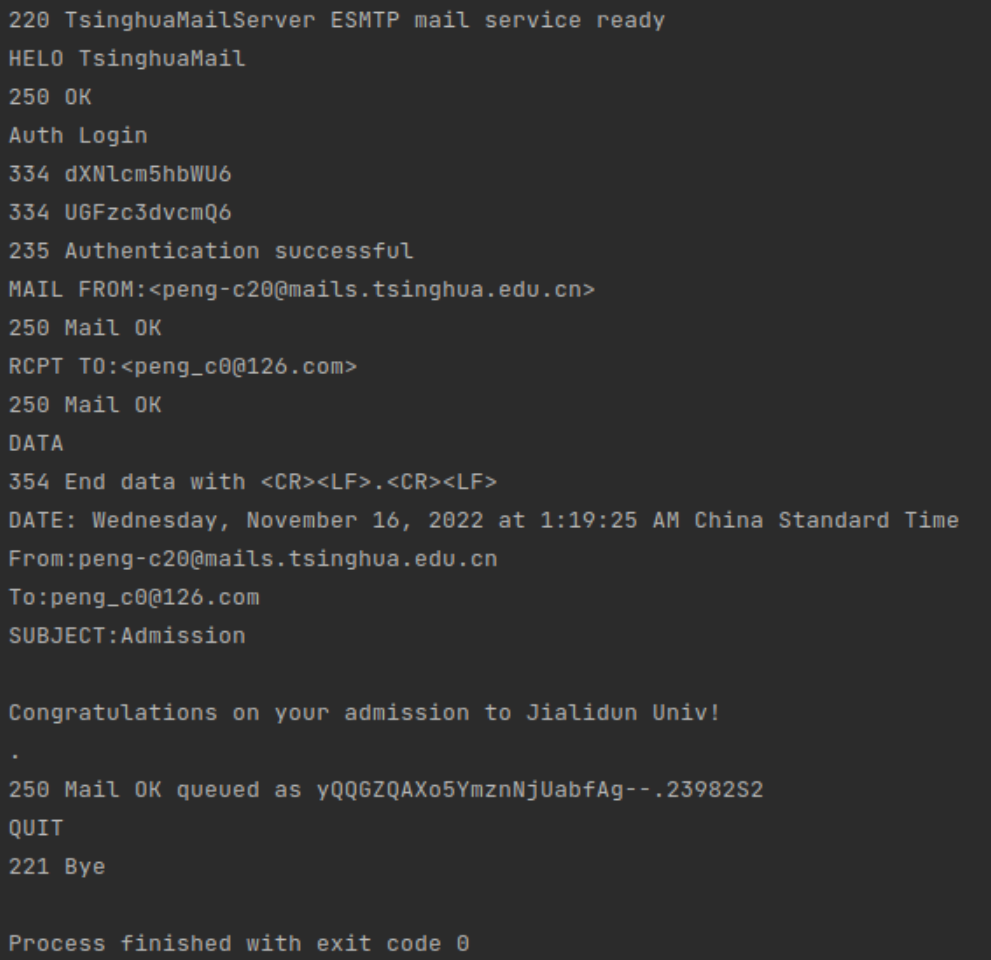


清华邮箱截图：



1. 带服务器身份验证的邮件发送客户端实验（Auth SMTP）

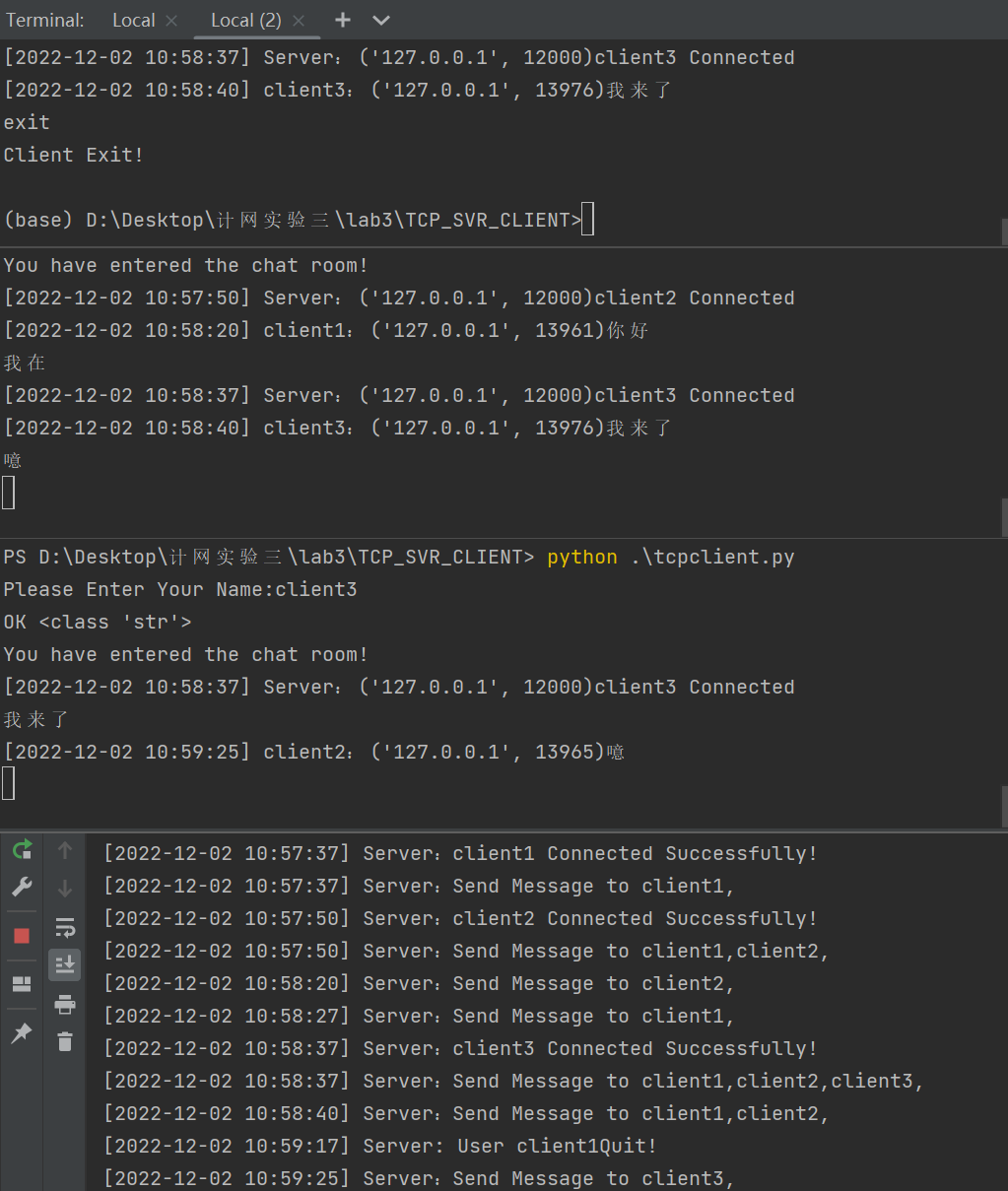
控制台输出截图：



校外邮箱截图：



3．自主编写通过 socket 收发信息的客户端和服务器端程序



上图展示了此任务实现的功能：

1. 分为客户端和服务器两部分。
2. 服务器仅起到连接多个客户端、中转消息的作用，本身不能参与聊天。  
   服务器必须可以同时接受多个客户端（至少三个）进行群聊。（图中上方的三个方框为三个用户端，下方为服务器）
3. 客户端应当能够服务器连接到服务器，连接到服务器后，可以发送消息  
   和接收消息。每当任何一个客户端发送消息，必须保证所有连接到同一  
   服务器的其他客户端均能接收到消息。（由图中可以看到，均收到了消息）
4. 程序需要从标准输入中读取消息内容；收到的其他人的消息，需要在标  
   准输出中打印出来。打印其他人发送的消息时，应当至少包括发送者的  
   IP 和端口、发送消息的时间、消息的内容。（由图中可以看到基本信息得到打印）  
   5、服务器应当具有基本的鲁棒性，例如某个客户端突然与服务器断开连接，  
   你的服务器不能崩溃。（图中client1最后退出，2、3仍然能通信）

实现方式：

本实验主要用thread带来的多线程方式实现。

1. 实验思考
2. Simple SMTP 和常用的 E-mail 客户端在功能结构上的比较。

功能： Simple SMTP 可以随意地编写发件人的邮件地址，因此可以发送伪邮件，而且实际收件人和邮件编写中写的收件人可以不相同， 实际发件人和邮件编写中写的发件人也可以不相同（在清华邮箱里会标注“真实的发送地址和宣称的发件人地址不同，请谨慎审视邮件内容的真实性”）；而 E-mail 客户端一般收件人与实际收件人应相同，发件人的邮件地址一般默认为自己的邮件地址不能修改， 同时邮件发送比较容易简单。

结构： Simple SMTP 结构层次分明，在发送邮件的过程中需要遵循 RFC 协议  
一步一步和服务器交互，最终才可以完成发送；而 E-mail 客户端可以直接使用  
可视化的图形界面，更加方便地修改邮件的各项信息，而不需要发送代码指令即  
可完成邮件的发送。

1. 使用 TCP 和 UDP 各自的优缺点比较。

|  |  |  |
| --- | --- | --- |
|  | TCP | UDP |
| 优点 | TCP 是面向连接的传输，提供了可靠的数据传递，通过流量控制、序号、确认和定时器等技术，TCP 确保正确地、按序地将数据从发送进程交付到接收进程，此外还提供了拥塞控制。 | UDP 是无连接的传输，是一种尽力而为的服务。不用建立连接（减少了时延）；简单（发送方和接收方不用保存连接状态）；更小的报文段头文件开销；无拥塞控制，可以尽快发送。 |
| 缺点 | TCP 发送速度慢，效率低，占用 资源多，易被攻击。 | UDP 的报文段可能会丢失、误码、乱序等。 无拥塞控制，可能造成网络拥堵 |

1. 针对程序中出现的问题及解决方法，写出实验体会。

在做第一问时，我遇到了“553 authentication is required”的报错，这是因为我编写的发件方是“@jialidun.edu.cn” 可能存在安全机制使得我不能这样发送，所以我重新编了一个邮箱后缀解决了问题。

在实现TCP聊天室时，刚开始并不会处理接发信息之间的关系，后来参考指导说明选择了多线程的方式，对信息处理和网络的处理方式有了更深的认识。  
 在本次实验中，我对 SMTP 协议和套接字编程有了更加深入的理解，实现了许多有趣、有用的实际功能， 我收获颇丰。