

实验三 Socket 编程实现网络通信 实验报告

姓名：李宇飏

班级：自05

学号：2020011645

SMTP

实验内容

1. 根据 SMTP 协议（RFC2821），补全无身份验证的 SMTP 程序；
2. 根据 SMTP 拓展协议（RFC2554），补全含有身份验证的 SMTP 程序。

实验过程

补全 SimpleSender.java

补全代码请见 CodeBase/SMTP/SimpleSender.java 标注 `// DONE: x` 处。编译并运行，命令行输出见 CodeBase/SMTP/SimpleSender.out。

验证在邮箱 `liyuyang20@mails.tainghua.edu.cn` 收到了对应邮件：

回复 回复全部 转发 删除 来信分类 这不是垃圾邮件 标记为 移动到 更多

The N-th Electronic Mail From Aiden to THU

发件人: Fake Aid... <i@fake-aidenli.net>

时 间:

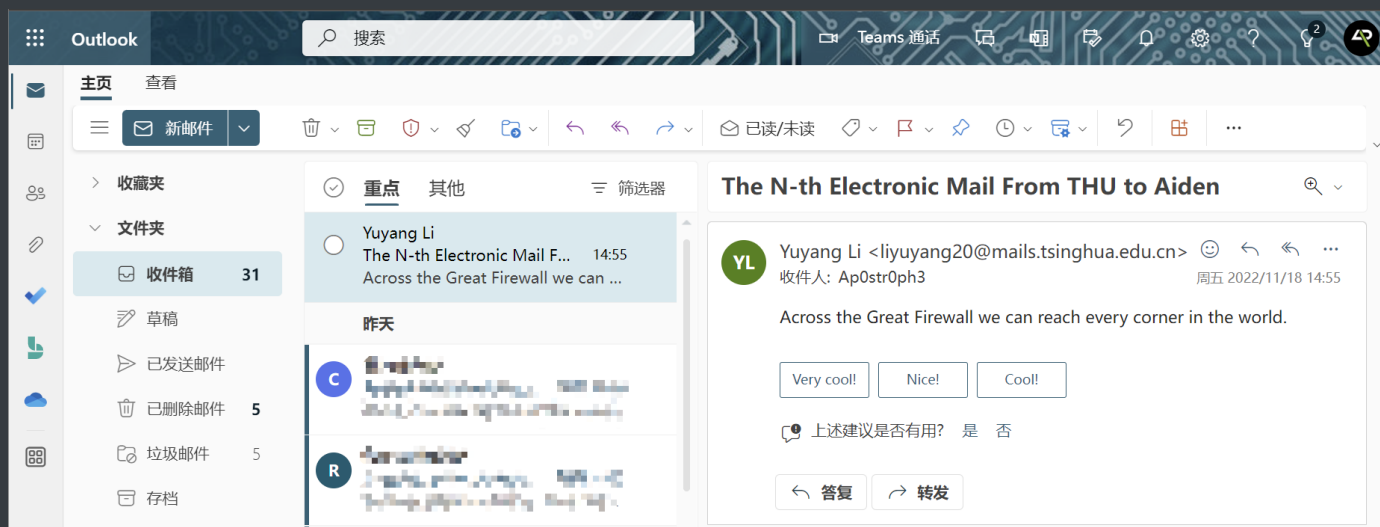
收件人: Yuyang Li <liyuyang20@mails.tsinghua.edu.cn>

Across the Great Firewall we can reach every corner in the world.

补全 AuthSender.java

补全代码请见 `CodeBase/SMTP/AuthSender.java` 标注 `// DONE: x` 处。编译并运行，命令行输出见 `CodeBase/SMTP/AuthSender.out`，密码用 `*` 代替。

验证在邮箱 `i@aidenli.net` 收到了对应邮件：



TCP 聊天室

实验内容

1. 使用 Python 编写基于 Socket 的多人文字聊天室
2. 测试聊天室功能，包括：连接、收发消息、断连处理

实验过程

编写聊天室

首先编写客户端、服务器类，由于二者具有一些相似的功能（比如创建接收/发送信息线程、记录 Log 等），首先编写父类 `ChatPeer`，然后派生 `ChatClient`，`ChatServer`。

服务器主要功能为：

1. 创建 `welcomeSocket`，监听客户端连接；
2. 维护 `clientPool`，将连接的客户端的客户端 `accept` 为 `serverSocket`；
3. 通过每个 `serverSocket` 监听并从客户端接收消息，并转发给其他所有连接；
4. 对于发送失败的 `serverSocket`，关闭与客户端的连接。

客户端主要功能为：

1. 创建 `clientSocket`，连接服务器；
2. 通过 `clientSocket` 监听并从服务器接收消息，并打印内容，同时显示发送方 IP、端口、时间；
3. 通过 `clientSocket` 向服务器发送消息；

此外，客户端、服务器均应当：

1. 充分记录 log，以 `INFO`，`DEBUG`，`ERROR` 等级别区分；
2. 妥当处理所有潜在的 `exception` 抛出；
3. 妥当地新建、维护和关闭线程，使前端非阻塞。

为了前端非阻塞，需要采用多线程/异步方法。为简洁记，本次采用 `threading.Thread` 通过多线程实现。具体地，

- `welcomeSocket` 及每个 `serverSocket` 单独新建线程，前者在服务器端程序结束时释放，后者在该 `socket` 连接失败后释放；
- `clientSocket` 单独新建线程，并在结束连接时释放；
- 客户端、服务端在发送每条消息时新建线程发送，在发送结束后释放。

此外，本次实验还是先了一些额外功能，包括但不限于：

1. 消息类型，目前支持：文本消息 `MSG`，控制消息 `CTL`（用于上线、下线通知）。
2. 用户上线、下线通知；
3. 用户端命令：如可以通过 `:q` 断开连接，通过 `::` 转义（如 `::q` 代表发送文本 `":q"`）。

源码见 `CodeBase/TCP-Chat`。

测试聊天室

聊天室不包含额外依赖，可以直接使用，在 Python 3.8-3.9 测试正常。

运行服务器：

```
python run.py --mode server --name Server --server_port PORT --log  
log/server.log
```

运行客户端：

```
python run.py --mode server --name Alice --server_addr ADDR --server_port  
PORT --log log/alice.log
```

其中，

- `ADDR`，`PORT` 为服务器对外开放服务的地址与端口（由 `welcomeSocket` 绑定）
- 服务端不需要指定 `server_addr`
- 而应当允许防火墙在 `server_port` 端口的 TCP 出入连接。

在服务器上启动服务器，以 himalia.adeinli.net:12333 为例，分别从 2 台设备的 3 个 shell 启动客户端连接服务器，测试：

- 1. 连接服务器；
- 2. 收发消息；
- 3. 手动退出程序；
- 4. 客户端程序中断，不影响服务端；重新连接并回到会话。

Alice、Bob（同一个出口 IP，不同端口）

bash

(networks) **aiden@Aiden-WS**:/mnt/d/Dev/Projects/ComputerNetworkLabs/Lab-3/Codebase/TCP-Chat\$ python run.py --mode client --name Bob --server_addr himalia.aidenli.net --server_port 12333 --log log/bob.log

[2022-11-30 10:55:46] Alice(59.66.156.26:22428)

Hi Bob

Hi Alice

[2022-11-30 10:55:51] System

Charlie(183.172.185.185) is now online!

Here comes Charlie!

[2022-11-30 10:56:52] Charlie(183.172.185.185:49937)

Hey guys, OpenAI has just released a new model called davinci-003 for GPT-3!

Wow, cool, gonna check it out!

[2022-11-30 10:57:03] System

Charlie(183.172.185.185) is now offline!

[2022-11-30 10:57:07] System

Alice(59.66.156.26) is now online!

[2022-11-30 10:57:17] Alice(59.66.156.26:22432)

Just got disconnected by accident... Did I miss anything?

Charlie said that OpenAI has just released a new model called davinci-003 for GPT-3! Go check that out!

[2022-11-30 10:57:54] Alice(59.66.156.26:22432)

Wow, thanks bro, gonna launch my playground!

[2022-11-30 10:57:57] System

Alice(59.66.156.26) is now offline!

:

:q

See you next time! Press ENTER to exit...

(networks) **aiden@Aiden-WS**:/mnt/d/Dev/Projects/ComputerNetworkLabs/Lab-3/Codebase/TCP-Chat\$

PROBLEMS

OUTPUT

DEBUG CONSOLE

JUPYTER

TERMINAL

(base) **aiden@Aiden-WS**:/mnt/d/Dev/Projects/ComputerNetworkLabs/Lab-3/Codebase/TCP-Chat\$ python run.py --mode client --name Alice --server_addr himalia.aidenli.net --server_port 12333 --log log/alica.log

[2022-11-30 10:55:43] System

Bob(59.66.156.26) is now online!

Hi Bob

[2022-11-30 10:55:49] Bob(59.66.156.26:22430)

Hi Alice

[2022-11-30 10:55:51] System

Charlie(183.172.185.185) is now online!

[2022-11-30 10:55:59] Bob(59.66.156.26:22430)

Here comes Charlie!

^CTraceback (most recent call last):

File "/mnt/d/Dev/Projects/ComputerNetworkLabs/Lab-3/Codebase/TCP-Chat/run.py", line 51, in <module>

line = input("")

KeyboardInterrupt

^CException ignored in: <module 'threading' from '/home/aiden/anaconda3/lib/python3.9/threading.py'>

Traceback (most recent call last):

File "/home/aiden/anaconda3/lib/python3.9/threading.py", line 1477, in _shutdown

lock.acquire()

KeyboardInterrupt:

(base) **aiden@Aiden-WS**:/mnt/d/Dev/Projects/ComputerNetworkLabs/Lab-3/Codebase/TCP-Chat\$ python run.py --mode client --name Alice --server_addr himalia.aidenli.net --server_port 12333 --log log/alica.log

Just got disconnected by accident... Did I miss anything?

[2022-11-30 10:57:38] Bob(59.66.156.26:22430)

Charlie said that OpenAI has just released a new model called davinci-003 for GPT-3! Go check that out!

Wow, thanks bro, gonna launch my playground!

:q

See you next time! Press ENTER to exit...

(base) **aiden@Aiden-WS**:/mnt/d/Dev/Projects/ComputerNetworkLabs/Lab-3/Codebase/TCP-Chat\$

Charlie（不同出口 IP）

```
aiden@ApOstr0Book:~/dev/THU-Computer-Network-Labs/Lab-3/Codebase/TCP-Chat
15% | 4.0 GB | 0.0
~/dev/THU-Computer-Network-Labs/Lab-3/Codebase/TCP-Chat on main !5
python run.py --mode client --name Charlie --server_addr himalia.aidenli.net --server_port 12333 --log log/charlie.log
[2022-11-30 10:55:59] Bob(59.66.156.26:22430)
Here comes Charlie!

Hey guys, OpenAI has just released a new model called davinci-003 for GPT-3!
[2022-11-30 10:56:59] Bob(59.66.156.26:22430)
Wow, cool, gonna check it out!

:q
See you next time! Press ENTER to exit...

~/dev/THU-Computer-Network-Labs/Lab-3/Codebase/TCP-Chat on main !5
```

Log 数据

客户端，以 Alice、Bob 的 log 为例：

```
alice.log
log
1 2022-11-30 02:55:42,496 - INFO - CLIENT Alice(ID: a4689fbd) initialized.
2 2022-11-30 02:55:42,553 - INFO - Connected to server at himalia.aidenli.net:12333
3 2022-11-30 02:55:42,553 - INFO - Sent message to ('43.132.130.243', 12333).
4 2022-11-30 02:55:44,236 - INFO - Received message from ('43.132.130.243', 12333).
5 2022-11-30 02:55:46,841 - INFO - Sent message to ('43.132.130.243', 12333).
6 2022-11-30 02:55:49,742 - INFO - Received message from ('43.132.130.243', 12333).
7 2022-11-30 02:55:52,286 - INFO - Received message from ('43.132.130.243', 12333).
8 2022-11-30 02:56:00,074 - INFO - Received message from ('43.132.130.243', 12333).
9 2022-11-30 02:57:07,662 - INFO - CLIENT Alice(ID: d80cfac6) initialized.
10 2022-11-30 02:57:07,738 - INFO - Connected to server at himalia.aidenli.net:12333
11 2022-11-30 02:57:07,740 - INFO - Sent message to ('43.132.130.243', 12333).
12 2022-11-30 02:57:17,751 - INFO - Received message from ('43.132.130.243', 12333).
13 2022-11-30 02:57:38,962 - INFO - Received message from ('43.132.130.243', 12333).
14 2022-11-30 02:57:55,296 - INFO - Sent message to ('43.132.130.243', 12333).
15 2022-11-30 02:57:57,985 - INFO - Sent message to ('43.132.130.243', 12333).
16 2022-11-30 02:57:57,985 - INFO - CLIENT Alice(d80cfac6) stopped.
17 2022-11-30 02:57:57,985 - INFO - Connection to server is now closed.
18

bob.log
log
1 2022-11-30 02:55:03,640 - INFO - CLIENT Bob(ID: aaf1d899) initialized.
2 2022-11-30 02:55:44,140 - INFO - CLIENT Bob(ID: 1b8ed4ca) initialized.
3 2022-11-30 02:55:44,188 - INFO - Connected to server at himalia.aidenli.net:12333
4 2022-11-30 02:55:44,189 - INFO - Sent message to ('43.132.130.243', 12333).
5 2022-11-30 02:55:46,892 - INFO - Received message from ('43.132.130.243', 12333).
6 2022-11-30 02:55:49,691 - INFO - Sent message to ('43.132.130.243', 12333).
7 2022-11-30 02:55:52,286 - INFO - Received message from ('43.132.130.243', 12333).
8 2022-11-30 02:56:00,026 - INFO - Sent message to ('43.132.130.243', 12333).
9 2022-11-30 02:56:52,591 - INFO - Received message from ('43.132.130.243', 12333).
10 2022-11-30 02:56:59,447 - INFO - Sent message to ('43.132.130.243', 12333).
11 2022-11-30 02:57:03,688 - INFO - Received message from ('43.132.130.243', 12333).
12 2022-11-30 02:57:07,788 - INFO - Received message from ('43.132.130.243', 12333).
13 2022-11-30 02:57:17,880 - INFO - Received message from ('43.132.130.243', 12333).
14 2022-11-30 02:57:38,914 - INFO - Sent message to ('43.132.130.243', 12333).
15 2022-11-30 02:57:55,346 - INFO - Received message from ('43.132.130.243', 12333).
16 2022-11-30 02:57:55,035 - INFO - Received message from ('43.132.130.243', 12333).
17 2022-11-30 02:58:04,043 - INFO - Sent message to ('43.132.130.243', 12333).
18 2022-11-30 02:58:04,044 - INFO - CLIENT Bob(1b8ed4ca) stopped.
19 2022-11-30 02:58:04,044 - INFO - Connection to server is now closed.
20
```

服务器端（节选）：

```
aiden@himalia: ~/dev/THU-C x + v
2022-11-30 10:56:52,161 - INFO - Sent message to ('59.66.156.26', 22428)
2022-11-30 10:56:52,161 - INFO - Sent message to ('59.66.156.26', 22430)
2022-11-30 10:56:59,064 - INFO - Received message from client, broad casting...
2022-11-30 10:56:59,065 - INFO - Received MSG message from Bob(59.66.156.26:22430). Message was sent at 2022-11-30 10:56:59.
2022-11-30 10:56:59,065 - INFO - Sent message to ('59.66.156.26', 22428)
2022-11-30 10:56:59,065 - INFO - Sent message to ('183.172.185.185', 49937)
2022-11-30 10:57:03,257 - INFO - Received message from client, broad casting...
2022-11-30 10:57:03,257 - INFO - Received CTL message from Charlie(183.172.185.185:49937). Message was sent at 2022-11-30 10:57:03.
2022-11-30 10:57:03,258 - INFO - Sent message to ('59.66.156.26', 22428)
2022-11-30 10:57:03,258 - INFO - Removed connection socket bf343604 from client pool.
2022-11-30 10:57:03,258 - INFO - Connection bf343604 is now closed.
2022-11-30 10:57:03,258 - INFO - Sent message to ('59.66.156.26', 22430)
2022-11-30 10:57:07,356 - INFO - Accepted connection from ('59.66.156.26', 22432)
2022-11-30 10:57:07,358 - INFO - Sent message to ('59.66.156.26', 22428)
2022-11-30 10:57:07,359 - INFO - Started to receive message from 59.66.156.26:22432 via connection f1984257.
2022-11-30 10:57:07,359 - INFO - Sent message to ('59.66.156.26', 22430)
2022-11-30 10:57:17,369 - INFO - Received message from client, broad casting...
2022-11-30 10:57:17,369 - INFO - Received MSG message from Alice(59.66.156.26:22432). Message was sent at 2022-11-30 10:57:17.
2022-11-30 10:57:17,370 - INFO - Sent message to ('59.66.156.26', 22428)
2022-11-30 10:57:17,370 - INFO - Sent message to ('59.66.156.26', 22430)
2022-11-30 10:57:38,532 - INFO - Received message from client, broad casting...
2022-11-30 10:57:38,533 - INFO - Received MSG message from Bob(59.66.156.26:22430). Message was sent at 2022-11-30 10:57:38.
2022-11-30 10:57:38,533 - INFO - Sent message to ('59.66.156.26', 22428)
2022-11-30 10:57:38,533 - INFO - Sent message to ('59.66.156.26', 22432)
2022-11-30 10:57:54,914 - INFO - Received message from client, broad casting...
2022-11-30 10:57:54,915 - INFO - Received MSG message from Alice(59.66.156.26:22432). Message was sent at 2022-11-30 10:57:54.
2022-11-30 10:57:54,915 - INFO - Sent message to ('59.66.156.26', 22428)
2022-11-30 10:57:54,915 - INFO - Sent message to ('59.66.156.26', 22430)
2022-11-30 10:57:57,604 - INFO - Received message from client, broad casting...
2022-11-30 10:57:57,604 - INFO - Received CTL message from Alice(59.66.156.26:22432). Message was sent at 2022-11-30 10:57:57.
2022-11-30 10:57:57,604 - INFO - Sent message to ('59.66.156.26', 22428)
2022-11-30 10:57:57,604 - INFO - Removed connection socket f1984257 from client pool.
2022-11-30 10:57:57,604 - INFO - Connection f1984257 is now closed.
2022-11-30 10:57:57,605 - INFO - Sent message to ('59.66.156.26', 22430)
2022-11-30 10:58:03,664 - INFO - Received message from client, broad casting...
2022-11-30 10:58:03,664 - INFO - Received CTL message from Bob(59.66.156.26:22430). Message was sent at 2022-11-30 10:58:03.
2022-11-30 10:58:03,664 - INFO - Removed connection socket a4e5de78 from client pool.
2022-11-30 10:58:03,664 - INFO - Connection a4e5de78 is now closed.
2022-11-30 10:58:03,665 - INFO - Sent message to ('59.66.156.26', 22428)
```

思考题与分析题

Simple SMTP 和常用的 E-mail 客户端在功能结构上的比较

Simple SMTP 仅仅是提供了一个发送邮件的功能，相比常见 E-Mail 客户端，还缺少若干功能，包括但不限于：

- 身份验证功能（在 AuthSMTP 里部分实现）
- 收件、通过 IMAP/POP3 拉取收件箱
- 草稿、已发件等其他结构功能
- 发送附件或富文本内容
- 使用 TLS 等加密数据

使用 TCP 和 UDP 各自的优缺点比较

优点	TCP	UDP
无需建立连接	否	是
可靠数据连接	是	否
拥塞机制/流量控制	有	无
支持超时重传	是	否
实时性 ↑	相比较低	相比较高
资源消耗 ↓	相比较高	相比较低

针对程序中出现的問題及解决方法，写出实验体会

实验中主要遇到了以下问题：

1. 若客户端意外断开连接，服务器端维护的 `serverSocket` 保持，在线程发送模式下，若不做异常处理，则会在下次发送时报错
解决方案：为线程设置 `Event()` 机制、`ExceptionHandler` 处理错误。
2. 客户端连接和下线时直接关闭连接，导致服务端需要通过异常处理的方式结束链接
解决方案：加入新消息类型 `CTL`，通过该类消息控制连接。

实验感想

通过此次实验，我感受到了网络通讯中，为了保障程序正常和消息收发正常，而需要考虑的状态机的复杂性。尽管此次实验中，程序的状态机较为简单，但在复杂场景中，有更多需要考量的因素和情况。

此外，我体会到了 TCP 向上提供可靠数据连接的易用性。即使 TCP 自身处理报文收发、确认、重传等机制十分复杂，但在基于 Socket 编程时，这些机制已经被封装好并向上提供可靠数据连接。

此外，此次实验没有考虑数据安全性，且客户端、服务端仅仅实现了基础操作功能，并采用手动的线程管理模式，可在实验结束后继续完善。