

Задача по программированию сокетов 3: SMTP

Выполнение данной работы поможет вам лучше понимать принцип работы протокола SMTP. Вы также получите опыт в реализации стандартного протокола, используя язык Python.

Ваша задача – разработать простой почтовый клиент, который отправляет сообщения электронной почты произвольному получателю. Программа должна соединиться с почтовым сервером, используя протокол SMTP, и передать ему сообщение. В языке Python существует модуль, называемый `smtpplib`, в котором есть встроенные методы для отправки почты по протоколу SMTP, но мы не будем его использовать в данной лабораторной, потому что он скрывает все нюансы работы протокола SMTP и сокетов. В целях ограничения спама, некоторые почтовые серверы не принимают соединения TCP от произвольных источников. Для эксперимента, описанного ниже, вы можете попробовать использовать подключение как к почтовому серверу вашей организации, так и к популярному серверу веб-почты, такому, как AOL, например. Подключиться можете, используя два разных клиентских места – с домашнего компьютера или с рабочего.

Код

Ниже представлена заготовка для клиентской программы. Вам нужно ее завершить. Места, которые вы должны заполнить своим кодом, помечены как **#Начало вставки** и **#Конец вставки**. В каждом таком месте может содержаться одна или несколько строк вашего кода.

Дополнительное замечание

В некоторых случаях принимающий почтовый сервер может классифицировать вашу электронную почту как спам. Поэтому при просмотре сообщений, отправленных вашей клиентской программой, убедитесь, что вы проверили папку нежелательной почты.

Задание

Завершить код вашего SMTP-клиента и сделать скриншоты, подтверждающие получение почтового сообщения.

Шаблон кода почтового клиента на языке Python

```
from socket import *
msg = "\r\n Я люблю компьютерные сети!"
endmsg = "\r\n.\r\n"
# Выбираем почтовый сервер (например, сервер Google mail) и
# называем его mailserver
mailserver = #Начало вставки #Конец вставки
# Создаем сокет clientSocket и устанавливаем TCP-соединение с
# mailserver
#Начало вставки

#Конец вставки
recv = clientSocket.recv(1024)
print recv
if recv[:3] != '220':
    print 'код 220 от сервера не получен.'
# Отправляем команду HELO и выводим ответ сервера.
heloCommand = 'HELO Alice\r\n'
clientSocket.send(heloCommand)
recv1 = clientSocket.recv(1024)
print recv1
if recv1[:3] != '250':
    print 'код 250 от сервера не получен.'
# Отправляем команду MAIL FROM и выводим ответ сервера.
# Начало вставки

# Конец вставки
# Отправляем команду RCPT TO и выводим ответ сервера.
# Начало вставки

# Конец вставки
# Отправляем команду DATA и выводим ответ сервера.
# Начало вставки

# Конец вставки
# Отправляем данные сообщения.
# Начало вставки

# Конец вставки
# Сообщение завершается одинарной точкой.
# Начало вставки

# Конец вставки
# Отправляем команду QUIT и получаем ответ сервера.
# Начало вставки

# Конец вставки
```

Дополнительные задания

1. Многие почтовые серверы, такие как Google mail (адрес: smtp.gmail.com, порт: 587), требуют использования протоколов Transport Layer Security (TLS) или Secure Sockets Layer (SSL) для аутентификации клиентов и обеспечения безопасности. Добавьте команды TLS /SSL к уже имеющимся и реализуйте работу своего клиента, используя почтовый сервер Google mail с указанным выше адресом и портом.
2. Текущий SMTP-клиент работает только с текстом в теле сообщений электронной почты. Измените его таким образом, чтобы можно было отправлять письма и с текстом, и с изображениями.