



Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное
учреждение высшего образования
«Московский государственный технический университет имени
Н.Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н.Э. Баумана)

ФАКУЛЬТЕТ «Информатика и системы управления»

КАФЕДРА «Программное обеспечение ЭВМ и информационные технологии»

Отчет по лабораторной работе №5 по курсу «Компьютерные сети»

Тема Протокол SMTP. Реализация SMTP-клиента

Студент Кононенко С.С.

Группа ИУ7-73Б

Оценка (баллы) _____

Преподаватель Рогозин Н.О.

Задание

Написать SMTP-клиент, который

1. В качестве входных данных (аргументы командной строки) получает: адрес получателя, адрес отправителя, пароль.
2. Использует один из открытых SMTP-серверов для доставки MIME-сообщений, включая приложения, если они есть, в соответствии с вариантом.

Вариант №6, дополнительное задание №1. Доставка сообщений выполняется с регулярным интервалом. Интервал и тело сообщения, имя файла для прикрепления (опционально) вводятся с клавиатуры.

Программа

Работа выполнена на языке программирования **Go**, для отправки сообщений использована библиотека **gmail** (<https://github.com/go-gomail/gomail>), для выполнения работы с интервалом выбрана библиотека **gocron** (<https://github.com/go-co-op/gocron>). В качестве SMTP-сервера был выбран сервер **mail.ru** (<https://help.mail.ru/mail/mailer/popsmtplib>).

Листинг 1 – Код SMTP-клиента

```
1 package main
2
3 import (
4     "bufio"
5     "flag"
6     "fmt"
7     "log"
8     "os"
9     "strings"
10    "time"
11
12    "github.com/go-co-op/gocron"
13    "gopkg.in/gomail.v2"
14 )
15
16 var (
17     to, from, pass string
18     hasAttachment bool
19 )
20
21 func init() {
22     flag.StringVar(&to, "t", "somereceiveremail", "receiver's_email")
23     flag.StringVar(&from, "f", "somesenderemail", "sender's_email")
24     flag.StringVar(&pass, "p", "somesenderpass", "sender's_password")
25     flag.BoolVar(&hasAttachment, "a", false, "attach_file")
26 }
27
28 func main() {
29     flag.Parse()
30
31     var (
32         interval int
33         attachment, message string
34     )
35
36     fmt.Print("Input_sending_interval_(in_seconds):_")
```

```

37  fmt.Sprintf("%d", &interval)
38  fmt.Print("Input message: ")
39  reader := bufio.NewReader(os.Stdin)
40  message, _ = reader.ReadString('\n')
41  message = strings.Replace(message, "\n", "", -1)
42  m := formBaseMail(from, to, message)
43  if hasAttachment {
44      fmt.Print("Input attachment's name (absolute path or located in same directory as binary): ")
45      fmt.Sprintf("%s", &attachment)
46      m.Attach(attachment)
47  }
48
49  d := gomail.NewDialer("smtp.mail.ru", 465, from, pass)
50
51  s := gocron.NewScheduler(time.UTC)
52  s.Every(interval).Seconds().Do(func() {
53      if err := d.DialAndSend(m); err != nil {
54          log.Fatalln(err)
55      }
56  })
57  fmt.Println("Interval sending started. Press CTRL+C to exit")
58  s.StartBlocking()
59 }
60
61 func formBaseMail(from, to, body string) *gomail.Message {
62     m := gomail.NewMessage()
63
64     m.SetHeader("From", from)
65     m.SetHeader("To", to)
66     m.SetHeader("Subject", "Test mail")
67     m.SetBody("text/plain", body)
68
69     return m
70 }

```

Пример работы программы



```

~/Study/cn-7th-sem-labs/lab_05 [P] master 72 > ./build/smtclient.exe -t sergkononenko@mail.ru -f hackfeed@inbox.ru -p
Input sending interval (in seconds): 60
Input message: Hello from computer networks!
Interval sending started. Press CTRL+C to exit

```

Рисунок 1 – Запуск клиента без возможности добавить вложение

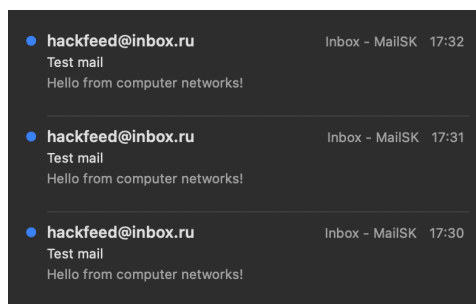


Рисунок 2 – Сообщения у получателя приходят с интервалом 60 секунд

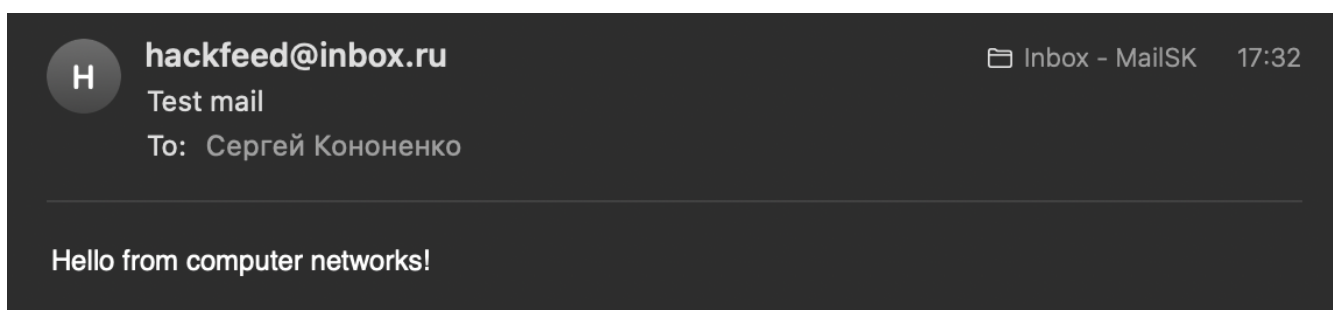


Рисунок 3 – Содержимое сообщения

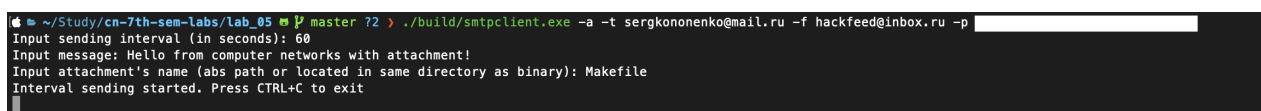


Рисунок 4 – Запуск клиента с возможностью добавить вложение

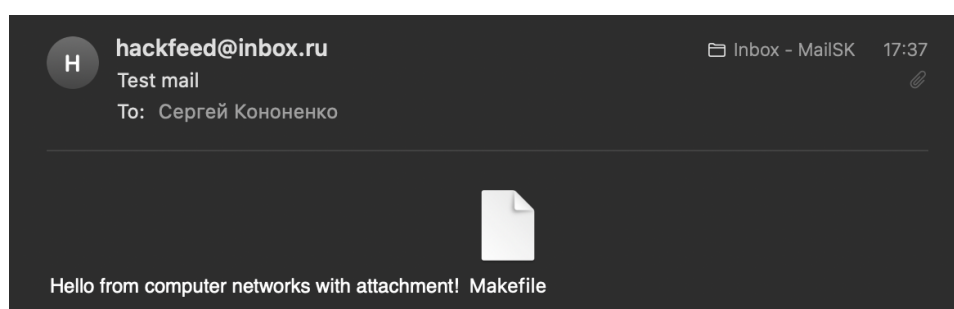


Рисунок 5 – Содержимое сообщения с вложением