



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 _____ «Теоретическая информатика и компьютерные технологии»

Лабораторная работа № 6
по курсу «Компьютерные сети»
«Разработка FTP- клиента»

Студент группы ИУ9-32Б Тараканов В. Д..

Преподаватель Посевин Д. П.

Москва 2024

1 Задание

Задача 1: Реализовать FTP-клиент на сервере.

Задача 2: Протестировать соединение GO FTP-клиента с установленным на сервере students.yss.su FTP-сервером со следующими параметрами доступа:

- ftp-host: students.yss.su
- login: ftpiu8
- passwd: 3Ru7yOTA

Задача 3: Реализовать следующую функциональность:

- загрузку файла;
- скачивание файла;
- создание директории;
- удаление файла;
- получение содержимого директории;
- переход в директорию;
- удаление пустой директории;
- удаление директории рекурсивно.

2 Результаты

Исходный код программы представлен в листингах 1.

Листинг 1: main.go

```
1 package main
2
3 import (
4     bytes "bytes"
5     "fmt"
6     "github.com/jlaffaye/ftp"
7     "io/ioutil"
8     "log"
9     "os"
10    "strings"
11    "time"
12 )
13
14 func main() {
```

```

15     c, err := ftp.Dial("students.yss.su:21", ftp.DialWithTimeout(5*
time.Second))
16     if err != nil {
17         log.Fatal(err)
18     }
19
20     err = c.Login("ftpiu8", "3Ru7yOTA")
21     if err != nil {
22         log.Fatal(err)
23     }
24
25     command := ""
26     path := ""
27     curDir := "./"
28     for command != "END" {
29         fmt.Scan(&command)
30         if command == "END" {
31             if err := c.Quit(); err != nil {
32                 log.Fatal(err)
33             }
34             break
35         } else if command == "GET" {
36             fmt.Println("type a path")
37             fmt.Scan(&path)
38             r, err := c.Retr(path)
39             if err != nil {
40                 panic(err)
41             }
42             defer r.Close()
43
44             buf, err := ioutil.ReadAll(r)
45             err = os.WriteFile(path, buf, 0666)
46             if err != nil {
47                 log.Println(err)
48             }
49             fmt.Println("Done")
50         } else if command == "PUSH" {
51             fmt.Println("type a path")
52             fmt.Scan(&path)
53             text, err := os.ReadFile(path)
54             if err != nil {
55                 log.Println(err)
56             }
57             byte := bytes.NewBuffer(text)
58             err = c.Stor(path, byte)
59             if err != nil {

```

```

60         fmt.Println(err)
61     }
62     fmt.Println("Done")
63 } else if command == "MKDIR" {
64     fmt.Println("type a path")
65     fmt.Scan(&path)
66     err = c.Mkdir(path)
67     if err != nil {
68         fmt.Println(err)
69     }
70     fmt.Println("Done")
71 } else if command == "CD" {
72     fmt.Println("type a path")
73     fmt.Scan(&path)
74     curDir += path + "/"
75     err = c.Chdir(path)
76     if path == ".." {
77         new_path := strings.Split(curDir, "/")
78         curDir = strings.Join(new_path[:len(new_path)-3], "/")
79     }
80     fmt.Println(curDir)
81     if err != nil {
82         fmt.Println(err)
83     }
84     fmt.Println("Done")
85 } else if command == "RMDIR" {
86     fmt.Println("type a path")
87     fmt.Scan(&path)
88     err = c.RemoveDir(path)
89     if err != nil {
90         fmt.Println(err)
91     }
92     fmt.Println("Done")
93 } else if command == "RM" {
94     fmt.Println("type a path")
95     fmt.Scan(&path)
96     err = c.Delete(path)
97     if err != nil {
98         fmt.Println(err)
99     }
100    fmt.Println("Done")
101 } else if command == "LS" {
102     list, err := c.List("./")
103     if err != nil {
104         fmt.Println(err)
105     }

```

```

106         for _, v := range list {
107             fmt.Println(v.Name)
108         }
109         fmt.Println("Done")
110     } else if command == "RMREC" {
111         fmt.Println("type a path")
112         fmt.Scan(&path)
113         err = c.RemoveDirRecur(path)
114         if err != nil {
115             fmt.Println(err)
116         }
117         fmt.Println("Done")
118     }
119 }
120
121 }
122

```