To compile the client program run the following command:

```
g++ -o client Client2MultipleServers.cpp md5.cpp
```

To compile the server programs, open four command prompts and run the following command:

```
g++ -o DFS# DFS#.cpp –lpthread
```

```
(example: g++ -o DFS1 DFS1.cpp –lpthread)
```

Run the servers by running the following command:

```
./DFS1 10001
./DFS2 10002
./DFS3 10003
./DFS4 10004
```

To get the list of the files run the following command on the client side:

To run: ./client LIS filename.txt (all files which have been uploaded by this client are to be listed to get back the list)

```
./client LIS file1.txt file2.txt file3.txt
```

To put a file into the DFS servers run the following command:

```
./client PUT file2.txt
```

To get a file from the DFS servers run the following command:

```
./client2 GET file2.txt
```

(on get the file will be saved in the working directory with the name:
filename_fileBuiltfromDFS.txt

- During the PUT, the file will be divided into equal parts of 4 and will be encrypted and then will be written into the DFS servers.
- Encryption used:
  - Converting the password and the plain text into hex form
  - Bitwise – XOR of the hex forms of password and plain text
- Decryption:
  - Xor the cipher text with the hex form of password to get the plain text

    - Converting the hex form of the resultant plain text in above step to normal string form.

- Configuration Files:
    - DFS.conf:
        - The server side configuration files will have the username and password of the clients that can connect to the DFS.
    - DFC.conf
        - The client side configuration file will have the port numbers of the DFS servers and also the user name and password which the client uses to connect to the servers.
    - When there is a mismatch in these usernames and passwords, the connection between client and server will be unsuccessful.