

Server

Создано системой Doxygen 1.9.1



1 Алфавитный указатель классов	1
1.1 Классы	1
2 Список файлов	3
2.1 Файлы	3
3 Классы	5
3.1 Класс <code>WorkWithClient</code>	5
3.1.1 Подробное описание	6
3.1.2 Конструктор(ы)	6
3.1.2.1 <code>WorkWithClient()</code>	6
3.1.2.2 <code>~WorkWithClient()</code>	7
3.1.3 Методы	7
3.1.3.1 <code>checID()</code>	7
3.1.3.2 <code>checparol()</code>	7
3.1.3.3 <code>count()</code>	8
3.1.3.4 <code>errrecord()</code>	8
3.1.3.5 <code>inttohex()</code>	9
3.1.3.6 <code>makesalt()</code>	9
4 Файлы	11
4.1 Файл <code>main.cpp</code>	11
4.1.1 Подробное описание	11
4.2 Файл <code>mdfile.cpp</code>	12
4.2.1 Подробное описание	12
4.3 Файл <code>mdfile.h</code>	13
4.3.1 Подробное описание	14
Предметный указатель	15



# Глава 1

## Алфавитный указатель классов

### 1.1 Классы

Классы с их кратким описанием.

<a href="#">WorkWithClient</a>	
Класс для работы с данными клиентов . . . . .	<a href="#">5</a>



## Глава 2

# Список файлов

### 2.1 Файлы

Полный список документированных файлов.

<a href="#">main.cpp</a>	Главный модуль проекта . . . . .	11
<a href="#">mdfile.cpp</a>	Модуль реализации методов класса <a href="#">WorkWithClient</a> . . . . .	12
<a href="#">mdfile.h</a>	Заголовочный файл проекта . . . . .	13





## Глава 3

# Классы

### 3.1 Класс WorkWithClient

Класс для работы с данными клиентов

```
#include <mdfile.h>
```

Открытые члены

- `WorkWithClient ()=delete`  
запрет конструктора без параметров
- `WorkWithClient (const std::string bazep_adr, const std::string ferr_adr, const int n_port)`  
Конструктор
- `~WorkWithClient ()`  
Деструктор
- `std::string makesalt ()`  
Формирование случайного числа
- `bool checID (const std::string newid)`  
Проверка ID клиента
- `bool checparol (const std::string parol_to_chek)`  
Проверка пароля клиента
- `uint64_t count (const uint32_t len, const uint64_t *data)`  
Перемножение значений вектора
- `void errecored (const std::string krit, const std::string reazon)`  
Запись ошибок в файл

Закрытые члены

- `std::string inttohex (const uint64_t decnum)`  
Преобразование числа из десятичной системы в шестнадцатеричную

## Закрытые данные

- `std::ofstream ferr`  
файл для записи ошибок
- `std::string allclients`  
список всех клиентов и их паролей
- `std::string clientpar`  
пароль идентифицированного клиента
- `std::string salt = ""`  
переменная для хранения созданного случайного числа

### 3.1.1 Подробное описание

Класс для работы с данными клиентов

В классе реализованы функции для идентификации и аутификации клиентов, обработки присылаемых ими значений

### 3.1.2 Конструктор(ы)

#### 3.1.2.1 WorkWithClient()

```
WorkWithClient::WorkWithClient (
    const std::string bazep_adr,
    const std::string ferr_adr,
    const int n_port )
```

#### Конструктор

Конструктор открывает, считывает и закрывает файл базы клиентов; открывает файл для записи ошибок и проверяет выбранный порт.

#### Аргументы

in	bazep_adr	Адрес файла с базой клиентов. Адрес может быть полным путем или относительным путем для подкаталогов папки сервера.
in	ferr_adr	Адрес файла для записи ошибок. Адрес может быть полным путем или относительным путем для подкаталогов папки сервера.
in	n_port	Номер порта, выбранного для сервера.

#### Предупреждения

Порты со значением от 1024 зарезервированы системными приложениями и являются недопустимыми

## Исключения

string,если	произошла ошибка открытия или чтения файла базы клиентов, ошибка открытия файла записи ошибок или выбранный порт меньше 1025 или больше 65535.
-------------	--

## 3.1.2.2 ~WorkWithClient()

```
WorkWithClient::~WorkWithClient ( )
```

## Деструктор

Деструктор закрывает файл для записи ошибок.

## 3.1.3 Методы

## 3.1.3.1 checID()

```
bool WorkWithClient::checID (
    const std::string newid )
```

## Проверка ID клиента

Функция проверяет наличие переданного ID в списке клиентов и записывает в переменную clientpar соответствующий данному ID пароль.

## Возвращает

true, если такой ID есть в списке и false, если его нет

## 3.1.3.2 checparol()

```
bool WorkWithClient::checparol (
    const std::string parol_to_chek )
```

## Проверка пароля клиента

Функция соединяет сформированное случайное число с паролем из файла, хеширует полученную строку функцией MD5 и сравнивает полученную строку со строкой, переданной клиентом.

## Аргументы

in	parol_to_chek	Строка, переданная клиентом.
----	---------------	------------------------------

Возвращает

true, если строки совпадают и false, если нет

### 3.1.3.3 count()

```
uint64_t WorkWithClient::count (
    const uint32_t len,
    const uint64_t * data )
```

Перемножение значений вектора

Функция перемножает полученные значения и возвращает результат

Аргументы

in	len	Количество перемножаемых чисел.
in	data	Вектор содержащий перемножаемые числа.

Предупреждения

При переполнении вверх возвращается  $2^{64}-1$ , при переполнении вниз возвращается 0.

Возвращает

Произведение чисел

### 3.1.3.4 errrecord()

```
void WorkWithClient::errrecord (
    const std::string krit,
    const std::string reason )
```

Запись ошибок в файл

Функция производит запись возникающих ошибок в формате "'дата' 'критичность' – 'причина ошибки'"

Аргументы

in	krit	Критичность ошибки.
in	reason	Причина ошибки.

## 3.1.3.5 inttohex()

```
string WorkWithClient::inttohex (
    const uint64_t decnum ) [private]
```

Преобразование числа из десятичной системы в шестнадцатеричную

Функция преобразует число в строку шестнадцатеричных цифр.

Аргументы

in	decnum	Число для преобразования.
----	--------	---------------------------

Возвращает

Строка из шестнадцатеричных цифр

## 3.1.3.6 makesalt()

```
string WorkWithClient::makesalt ( )
```

Формирование случайного числа

Функция формирует строку длиной 16 со случайным шестнадцатеричным числом. Число может принимать значения от 1 до  $16^{16}-1$ . Нужная длина строки достигается дополнением слева цифрами "0".

Предупреждения

Для преобразования числа из десятичной системы в шестнадцатеричную используется функция inttohex.

Возвращает

Строка из 16 шестнадцатеричных цифр

Объявления и описания членов классов находятся в файлах:

- [mdfile.h](#)
- [mdfile.cpp](#)



# Глава 4

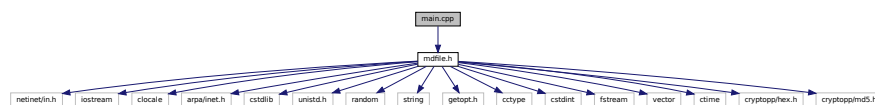
## Файлы

### 4.1 Файл main.cpp

Главный модуль проекта

```
#include "mdfile.h"
```

Граф включаемых заголовочных файлов для main.cpp:



### Функции

- int main (int argc, char \*argv[])

#### 4.1.1 Подробное описание

Главный модуль проекта

Автор

Нефедов А.А.

Дата

29.06.2023

Авторство

ИБСТ ПГУ

## Аргументы

opt	переменная для работы с параметрами командной строки
optarg	переменная для получения параметров командной строки

## 4.2 Файл mdfile.cpp

Модуль реализации методов класса [WorkWithClient](#).

```
#include "mdfile.h"
```

Граф включаемых заголовочных файлов для mdfile.cpp:



## Функции

- `bool multiplyWithOverflowCheck (uint64_t a, uint64_t b, uint64_t &result)`

### 4.2.1 Подробное описание

Модуль реализации методов класса [WorkWithClient](#).

Автор

Нефедов А.А.

Дата

29.06.2023

Авторство

ИБСТ ПГУ



## 4.3 Файл mdfile.h

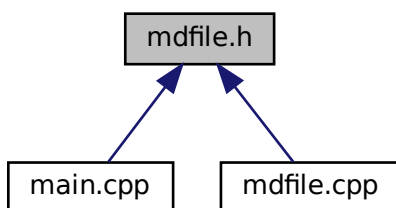
Заголовочный файл проекта

```
#include <netinet/in.h>
#include <iostream>
#include <clocale>
#include <arpa/inet.h>
#include <cstdlib>
#include <unistd.h>
#include <random>
#include <string>
#include <getopt.h>
#include <cctype>
#include <cstdint>
#include <fstream>
#include <vector>
#include <ctime>
#include <cryptopp/hex.h>
#include <cryptopp/md5.h>
```

Граф включаемых заголовочных файлов для mdfile.h:



Граф файлов, в которые включается этот файл:



Классы

- class [WorkWithClient](#)

Класс для работы с данными клиентов

Макросы

- `#define CRYPTOPP_ENABLE_NAMESPACE_WEAK 1`

#### 4.3.1 Подробное описание

Заголовочный файл проекта

Автор

Нефедов А.А.

Дата

29.06.2023

Авторство

ИБСТ ПГУ

# Предметный указатель

- ~WorkWithClient
  - WorkWithClient, [7](#)
- checID
  - WorkWithClient, [7](#)
- checparol
  - WorkWithClient, [7](#)
- count
  - WorkWithClient, [8](#)
- errrecord
  - WorkWithClient, [8](#)
- inttohex
  - WorkWithClient, [8](#)
- main.cpp, [11](#)
- makesalt
  - WorkWithClient, [9](#)
- mdfile.cpp, [12](#)
- mdfile.h, [13](#)
- WorkWithClient, [5](#)
  - ~WorkWithClient, [7](#)
  - checID, [7](#)
  - checparol, [7](#)
  - count, [8](#)
  - errrecord, [8](#)
  - inttohex, [8](#)
  - makesalt, [9](#)
  - WorkWithClient, [6](#)