

uint64_t vectors processing server
1.0

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>errorhandler</code>	5
3.1.1 Подробное описание	5
3.1.2 Конструктор(ы)	5
3.1.2.1 <code>errorhandler()</code>	5
3.1.3 Методы	6
3.1.3.1 <code>logger()</code>	6
3.2 Класс <code>getdata</code>	6
3.2.1 Подробное описание	7
3.2.2 Конструктор(ы)	7
3.2.2.1 <code>getdata()</code>	7
3.2.3 Методы	7
3.2.3.1 <code>get()</code>	7
3.2.4 Данные класса	7
3.2.4.1 <code>lp</code>	8
3.3 Класс <code>md5h</code>	8
3.3.1 Подробное описание	8
3.3.2 Конструктор(ы)	8
3.3.2.1 <code>md5h()</code>	9
3.3.3 Методы	9
3.3.3.1 <code>hash()</code>	9
3.3.3.2 <code>salt_generator()</code>	10
3.3.4 Данные класса	10
3.3.4.1 <code>salt</code>	10
3.4 Класс <code>server</code>	10
3.4.1 Подробное описание	11
3.4.2 Конструктор(ы)	11
3.4.2.1 <code>server()</code>	11
3.4.3 Методы	12
3.4.3.1 <code>accepting_connection()</code>	12
3.4.3.2 <code>authentication()</code>	12
3.4.3.3 <code>handling()</code>	12
3.4.3.4 <code>start()</code>	12
3.4.4 Данные класса	13
3.4.4.1 <code>buf</code>	13
3.4.4.2 <code>lp</code>	13
3.4.4.3 <code>rc</code>	13
3.4.4.4 <code>sckt</code>	13

3.4.4.5 userdata	13
3.4.4.6 wrkr	13
4 Файлы	15
4.1 Файл errorhandler.cpp	15
4.2 Файл errorhandler.hpp	15
4.2.1 Подробное описание	16
4.3 Файл getdata.cpp	17
4.4 Файл getdata.hpp	17
4.4.1 Подробное описание	18
4.5 Файл main.cpp	18
4.5.1 Подробное описание	19
4.5.2 Макросы	19
4.5.2.1 CRYPTOPP_ENABLE_NAMESPACE_WEAK	19
4.5.3 Функции	19
4.5.3.1 help()	19
4.5.3.2 main()	19
4.6 Файл md5hash.cpp	20
4.6.1 Макросы	20
4.6.1.1 CRYPTOPP_ENABLE_NAMESPACE_WEAK	20
4.7 Файл md5hash.hpp	20
4.7.1 Макросы	21
4.7.1.1 CRYPTOPP_ENABLE_NAMESPACE_WEAK	21
4.8 Файл server.cpp	21
4.9 Файл server.hpp	21
4.9.1 Подробное описание	22
Предметный указатель	23

Глава 1

Алфавитный указатель классов

1.1 Классы

Классы с их кратким описанием.

errorhandler	Класс обработки ошибок	5
getdata	Класс, обрабатывающий файл с ID:PASS клиентов	6
md5h	Класс, обрабатывающий HASH MD5	8
server	Класс управления работой сервера	10

Глава 2

Список файлов

2.1 Файлы

Полный список файлов.

errorhandler.cpp	15
errorhandler.hpp	
Класс обработки ошибок с записью в журнал	15
getdata.cpp	17
getdata.hpp	
Класс, обрабатывающий файл с ID:PASS клиентов	17
main.cpp	
Функция вывода справки и точка входа в программу	18
md5hash.cpp	20
md5hash.hpp	20
server.cpp	21
server.hpp	
Класс, управляющий работой сервера	21

Глава 3

Классы

3.1 Класс errorhandler

Класс обработки ошибок

```
#include <errorhandler.hpp>
```

Открытые члены

- `errorhandler` (const std::string lpath, const char *why, const int exitCode, bool critical)
Конструктор класса

Закрытые члены

- void `logger` (const std::string lpath, const char *why, const std::string dtime, const std::string critical)
Функция записи ошибок в журнал

3.1.1 Подробное описание

Класс обработки ошибок

3.1.2 Конструктор(ы)

3.1.2.1 errorhandler()

```
errorhandler::errorhandler (  
    const std::string lpath,  
    const char * why,  
    const int exitCode,  
    bool critical )
```

Конструктор класса

Аргументы

lpath	Путь к файлу журнала
why	Содержание ошибки
exitCode	Код ошибки
critical	Критичность ошибки

3.1.3 Методы

3.1.3.1 logger()

```
void errorHandler::logger (
    const std::string lpath,
    const char * why,
    const std::string dtime,
    const std::string critical ) [private]
```

Функция записи ошибок в журнал

Аргументы

lpath	Путь к файлу журнала
why	Содержание ошибки
dtime	Время ошибки
critical	Критичность ошибки

Объявления и описания членов классов находятся в файлах:

- [errorhandler.hpp](#)
- [errorhandler.cpp](#)

3.2 Класс getdata

Класс, обрабатывающий файл с ID:PASS клиентов

```
#include <getdata.hpp>
```

Открытые члены

- [getdata](#) (const string lpath)
Конструктор класса
- map< string, string > [get](#) (string fpath)
Запись ID:PASS в словарь

Закрытые данные

- string `lp`
Путь к файлу с журналом

3.2.1 Подробное описание

Класс, обрабатывающий файл с ID:PASS клиентов

3.2.2 Конструктор(ы)

3.2.2.1 `getdata()`

```
getdata::getdata (  
    const string lpath )
```

Конструктор класса

Аргументы

<code>lpath</code>	Путь к файлу с журналом
--------------------	-------------------------

3.2.3 Методы

3.2.3.1 `get()`

```
map< string, string > getdata::get (  
    string fpath )
```

Запись ID:PASS в словарь

Аргументы

<code>fpath</code>	Путь к файлу с ID:PASS клиентов
--------------------	---------------------------------

Возвращает

userdata ID:PASS

3.2.4 Данные класса

3.2.4.1 lp

```
string getdata::lp [private]
```

Путь к файлу с журналом

Объявления и описания членов классов находятся в файлах:

- [getdata.hpp](#)
- [getdata.cpp](#)

3.3 Класс md5h

Класс, обрабатывающий HASH MD5.

```
#include <md5hash.hpp>
```

Открытые члены

- [md5h](#) ()
Конструктор класса
- string [hash](#) (const string password)
Получение хэша от пароля

Открытые атрибуты

- string [salt](#)
Соль

Закрытые члены

- string [salt_generator](#) ()
Генератор соли

3.3.1 Подробное описание

Класс, обрабатывающий HASH MD5.

3.3.2 Конструктор(ы)

3.3.2.1 md5h()

```
md5h::md5h ( )
```

Конструктор класса

3.3.3 Методы

3.3.3.1 hash()

```
string md5h::hash (  
    const string password )
```

Получение хэша от пароля

Аргументы

password	Пароль
----------	--------

Возвращает

digest Хэш от пароля

3.3.3.2 salt_generator()

```
string md5h::salt_generator ( ) [private]
```

Генератор соли

Возвращает

tmp_s Соль

3.3.4 Данные класса

3.3.4.1 salt

```
string md5h::salt
```

Соль

Объявления и описания членов классов находятся в файлах:

- [md5hash.hpp](#)
- [md5hash.cpp](#)

3.4 Класс server

Класс управления работой сервера

```
#include <server.hpp>
```

Открытые члены

- `server` (`std::string fpath`, `int port`, `std::string lpath`)
Конструктор класса
- `int start` (`int port`, `char *address`)
Функция запуска сервера и привязки порта
- `int accepting_connection` ()
Функция ожидания и принятия соединения с клиентом
- `bool authentication` ()
Функция аутентификации пользователя
- `bool handling` ()
Функция обработки векторов

Закрытые данные

- `std::string lp`
Путь к файлу с логом
- `int sckt`
Сокет
- `int wrkr`
Действующее соединение с клиентом
- `int rc`
Ответ от клиента
- `char * buf` = `new char[4096]`
Буфер
- `std::map< std::string, std::string > userdata`
Данные клиента ID:PASS.

3.4.1 Подробное описание

Класс управления работой сервера

3.4.2 Конструктор(ы)

3.4.2.1 server()

```
server::server (
    std::string fpath,
    int port,
    std::string lpath )
```

Конструктор класса

Аргументы

<code>fpath</code>	Путь к файлу с ID:PASS клиентов
<code>port</code>	Порт для приема соединения
<code>lpath</code>	Путь к файлу журнала

3.4.3 Методы

3.4.3.1 accepting_connection()

```
int server::accepting_connection ( )
```

Функция ожидания и принятия соединения с клиентом

Возвращает

bool Действующее соединение с клиентом

3.4.3.2 authentication()

```
bool server::authentication ( )
```

Функция аутентификации пользователя

Возвращает

True, если аутентификация удалась, False - если нет

3.4.3.3 handling()

```
bool server::handling ( )
```

Функция обработки векторов

Возвращает

True, если обработка удалась, False - если нет

3.4.3.4 start()

```
int server::start (
    int port,
    char * address )
```

Функция запуска сервера и привязки порта

Аргументы

port	Порт для приема соединения
address	Сетевой адрес для приема соединения

Возвращает

sckt Сокет

3.4.4 Данные класса

3.4.4.1 buf

```
char* server::buf = new char[4096] [private]
```

Буфер

3.4.4.2 lp

```
std::string server::lp [private]
```

Путь к файлу с логом

3.4.4.3 rc

```
int server::rc [private]
```

Ответ от клиента

3.4.4.4 sckt

```
int server::sckt [private]
```

Сокет

3.4.4.5 userdata

```
std::map<std::string, std::string> server::userdata [private]
```

Данные клиента ID:PASS.

3.4.4.6 wrkr

```
int server::wrkr [private]
```

Действующее соединение с клиентом

Объявления и описания членов классов находятся в файлах:

- [server.hpp](#)
- [server.cpp](#)

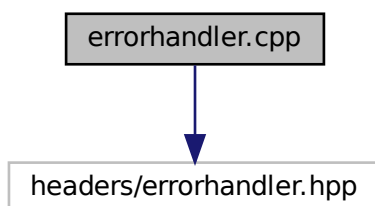
Глава 4

Файлы

4.1 Файл errorhandler.cpp

```
#include "headers/errorhandler.hpp"
```

Граф включаемых заголовочных файлов для errorhandler.cpp:



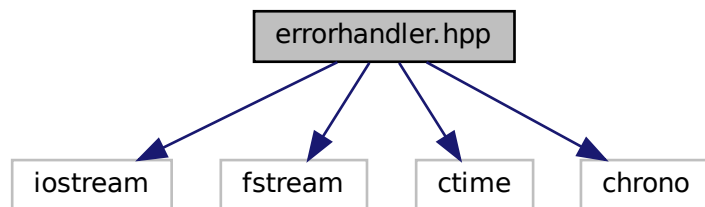
4.2 Файл errorhandler.hpp

Класс обработки ошибок с записью в журнал

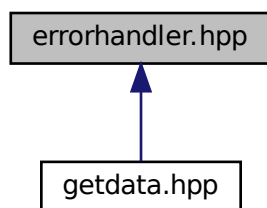
```
#include <iostream>
#include <fstream>
#include <ctime>
```

```
#include <chrono>
```

Граф включаемых заголовочных файлов для errorhandler.hpp:



Граф файлов, в которые включается этот файл:



Классы

- class `errorhandler`

Класс обработки ошибок

4.2.1 Подробное описание

Класс обработки ошибок с записью в журнал

Автор

Каспийский Н. К.

Версия

1.0

Дата

17.12.2023

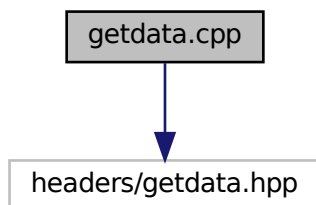
Авторство

ИБСТ ПГУ

4.3 Файл getdata.cpp

```
#include "headers/getdata.hpp"
```

Граф включаемых заголовочных файлов для getdata.cpp:

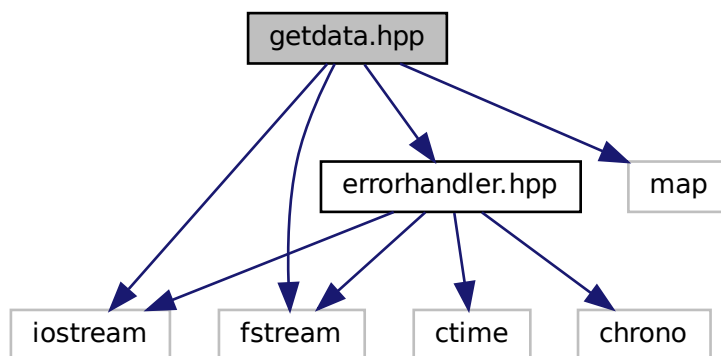


4.4 Файл getdata.hpp

Класс, обрабатывающий файл с ID:PASS клиентов

```
#include <iostream>
#include <fstream>
#include <map>
#include "errorhandler.hpp"
```

Граф включаемых заголовочных файлов для getdata.hpp:



Классы

- class [getdata](#)

Класс, обрабатывающий файл с ID:PASS клиентов

4.4.1 Подробное описание

Класс, обрабатывающий файл с ID:PASS клиентов

Автор

Каспийский Н. К.

Версия

1.0

Дата

17.12.2023

Авторство

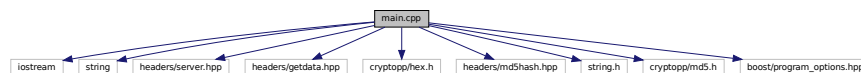
ИБСТ ПГУ

4.5 Файл main.cpp

Функция вывода справки и точка входа в программу

```
#include <iostream>
#include <string>
#include "headers/server.hpp"
#include "headers/getdata.hpp"
#include <cryptopp/hex.h>
#include "headers/md5hash.hpp"
#include <string.h>
#include <cryptopp/md5.h>
#include <boost/program_options.hpp>
```

Граф включаемых заголовочных файлов для main.cpp:



Макросы

- `#define CRYPTOPP_ENABLE_NAMESPACE_WEAK 1`

Функции

- `void help ()`
- `int main (int argc, const char *argv[])`

4.5.1 Подробное описание

Функция вывода справки и точка входа в программу

Автор

Каспийский Н. К.

Версия

1.0

Дата

17.12.2023

Авторство

ИБСТ ПГУ

4.5.2 Макросы

4.5.2.1 CRYPTOPP_ENABLE_NAMESPACE_WEAK

```
#define CRYPTOPP_ENABLE_NAMESPACE_WEAK 1
```

4.5.3 Функции

4.5.3.1 help()

```
void help ( )
```

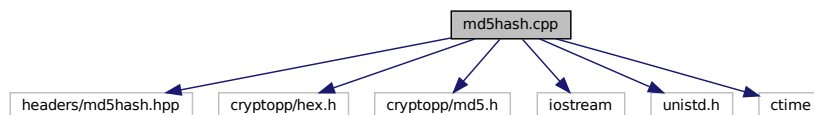
4.5.3.2 main()

```
int main (
    int argc,
    const char * argv[] )
```

4.6 Файл md5hash.cpp

```
#include "headers/md5hash.hpp"
#include <cryptopp/hex.h>
#include <cryptopp/md5.h>
#include <iostream>
#include <unistd.h>
#include <ctime>
```

Граф включаемых заголовочных файлов для md5hash.cpp:



Макросы

- `#define CRYPTOPP_ENABLE_NAMESPACE_WEAK 1`

4.6.1 Макросы

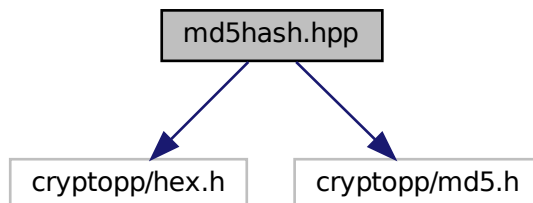
4.6.1.1 CRYPTOPP_ENABLE_NAMESPACE_WEAK

```
#define CRYPTOPP_ENABLE_NAMESPACE_WEAK 1
```

4.7 Файл md5hash.hpp

```
#include <cryptopp/hex.h>
#include <cryptopp/md5.h>
```

Граф включаемых заголовочных файлов для md5hash.hpp:



Классы

- class `md5h`
Класс, обрабатывающий HASH MD5.

Макросы

- `#define CRYPTOPP_ENABLE_NAMESPACE_WEAK 1`

4.7.1 Макросы

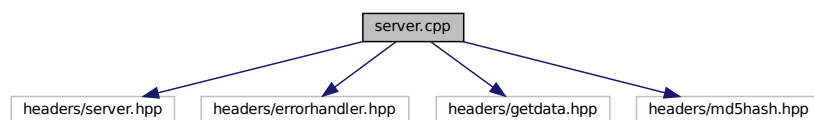
4.7.1.1 CRYPTOPP_ENABLE_NAMESPACE_WEAK

```
#define CRYPTOPP_ENABLE_NAMESPACE_WEAK 1
```

4.8 Файл server.cpp

```
#include "headers/server.hpp"
#include "headers/errorhandler.hpp"
#include "headers/getdata.hpp"
#include "headers/md5hash.hpp"
```

Граф включаемых заголовочных файлов для server.cpp:



4.9 Файл server.hpp

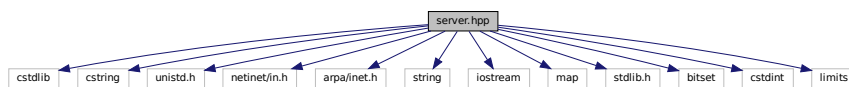
Класс, управляющий работой сервера

```
#include <cstdlib>
#include <cstring>
#include <unistd.h>
#include <netinet/in.h>
#include <arpa/inet.h>
#include <string>
#include <iostream>
#include <map>
#include <stdlib.h>
#include <bitset>
```

```
#include <cstdint>
```

```
#include <limits>
```

Граф включаемых заголовочных файлов для server.hpp:



Классы

- class [server](#)

Класс управления работой сервера

4.9.1 Подробное описание

Класс, управляющий работой сервера

Автор

Каспийский Н. К.

Версия

1.0

Дата

17.12.2023

Авторство

ИБСТ ПГУ

Предметный указатель

- accepting_connection
 - server, [12](#)
- authentication
 - server, [12](#)
- buf
 - server, [13](#)
- CRYPTOPP_ENABLE_NAMESPACE_WEAK
 - main.cpp, [19](#)
 - md5hash.cpp, [20](#)
 - md5hash.hpp, [21](#)
- errorhandler, [5](#)
 - errorhandler, [5](#)
 - logger, [6](#)
- errorhandler.cpp, [15](#)
- errorhandler.hpp, [15](#)
- get
 - getdata, [7](#)
- getdata, [6](#)
 - get, [7](#)
 - getdata, [7](#)
 - lp, [7](#)
- getdata.cpp, [17](#)
- getdata.hpp, [17](#)
- handling
 - server, [12](#)
- hash
 - md5h, [9](#)
- help
 - main.cpp, [19](#)
- logger
 - errorhandler, [6](#)
- lp
 - getdata, [7](#)
 - server, [13](#)
- main
 - main.cpp, [19](#)
- main.cpp, [18](#)
 - CRYPTOPP_ENABLE_NAMESPACE_WEAK, [19](#)
 - help, [19](#)
 - main, [19](#)
- md5h, [8](#)
 - hash, [9](#)
 - md5h, [8](#)
 - salt, [10](#)
 - salt_generator, [10](#)
- md5hash.cpp, [20](#)
 - CRYPTOPP_ENABLE_NAMESPACE_WEAK, [20](#)
- md5hash.hpp, [20](#)
 - CRYPTOPP_ENABLE_NAMESPACE_WEAK, [21](#)
- rc
 - server, [13](#)
- salt
 - md5h, [10](#)
- salt_generator
 - md5h, [10](#)
- sckt
 - server, [13](#)
- server, [10](#)
 - accepting_connection, [12](#)
 - authentication, [12](#)
 - buf, [13](#)
 - handling, [12](#)
 - lp, [13](#)
 - rc, [13](#)
 - sckt, [13](#)
 - server, [11](#)
 - start, [12](#)
 - userdata, [13](#)
 - wrkr, [13](#)
- server.cpp, [21](#)
- server.hpp, [21](#)
- start
 - server, [12](#)
- userdata
 - server, [13](#)
- wrkr
 - server, [13](#)