OTPNitro

Generated by Doxygen 1.8.3.1

Fri Jun 27 2014 19:48:41

Contents

Chapter 1

Class Index

1.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

Chapter 2

File Index

2.1 File List

Here is a list of all files with brief descriptions:

Chapter 3

Class Documentation

3.1 Config Class Reference

Configuration, path and pages management.

`#include <config.h>`

Public Member Functions

- Config (void)
  Config constructor.
- int getChars ()
  Returns the max PATH length.
- int getPages ()
  Returns the number of generated pages.
- char * getPath ()
  Returns the current PATH.
- void setChars (int)
  Set the max PATH length.
- void setPages (int)
  Set the num of pages to be generated.
- void setPath (char *)
  Set a new PATH to be used.
- void saveConfig ()
  Save the config to the default PATH
  If the config file doesnt exist it will create it with default values.

3.1.1 Detailed Description

Configuration, path and pages management.

3.1.2 Constructor & Destructor Documentation

3.1.2.1 Config::Config (void)

Config constructor.

Returns

Config object

Default PATH

Win32: The PATH is %APPDATA%\.otpnitro\

Unix: The PATH is $HOME/.otpnitro/

Files

The config file is always otpnitro.ini in the PATH root.

The pages are stored on the PAGES folder from the PATH root.

3.1.3 Member Function Documentation

3.1.3.1 int Config::getChars ( )

Returns the max PATH length.

Returns

MAX_CHARS

3.1.3.2 int Config::getPages ( )

Returns the number of generated pages.

Returns

MAX_PAGES

3.1.3.3 char * Config::getPath ( )

Returns the current PATH.

3.1.3.4 void Config::saveConfig (void)

Save the config to the default PATH

If the config file doesnt exist it will create it with default values.

3.1.3.5 void Config::setChars (int *chars*)

Set the max PATH length.

Parameters

chars	MAX_CHARS (int)

3.1.3.6 void Config::setPages (int *pages*)

Set the num of pages to be generated.

Parameters

pages	MAX_PAGES (int)

3.1.3.7 void Config::setPath (char * *path*)

Set a new PATH to be used.

The documentation for this class was generated from the following files:

- config.h
- config.cpp

3.2 Crypto Class Reference

Crypt and decrypt class.

`#include <crypto.h>`

Public Member Functions

- string decrypt (string, string)
  Decrypt a text (crypted) string.
- string encrypt (string, string)
  Crypt a text string.

3.2.1 Detailed Description

Crypt and decrypt class.

3.2.2 Member Function Documentation

3.2.2.1 string Crypto::decrypt (string *crypted,* string *out*)

Decrypt a text (crypted) string.

Parameters

crypted	Original (crypted) string
out	The ciphered text to sum(26)

Returns

The decrypted string

This function also replaces all "JQ" ocurrences from the decrypted text to spaces

3.2.2.2 string Crypto::encrypt (string *in,* string *out*)

Crypt a text string.

Parameters

in	Original (not crypted) string
out	The ciphered text to sum(26)

Returns

The crypted string

This function also remove the newline chars and replace all spaces to the "JQ" from the original text before to be crypted

The documentation for this class was generated from the following files:

- crypto.h
- crypto.cpp

3.3 Page Class Reference

Page operations class.

`#include <page.h>`

Public Member Functions

- Page (void)
  The constructor sets all parameters from the Config object.
- bool generate (string)
  Generate a complete Book using Page::get() and write it to the disk.
- bool write (int, string, string)
- int next (string)
  Returns the next unused page num for a book.
- bool burn (int, string)
  Secure page file delete.
- string read (int, string)
  Get the ciphertext page from a given book.
- string get ()
  Generate ciphertext page using the Rand class.
- string list ()
  Returns a list of valid books.

3.3.1 Detailed Description

Page operations class.

3.3.2 Constructor & Destructor Documentation

3.3.2.1 Page::Page (void)

The constructor sets all parameters from the Config object.

Returns

Page object

3.3.3 Member Function Documentation

3.3.3.1 bool Page::burn (int *page*, string *id*)

Secure page file delete.

Parameters

page	Page num
id	Book ID

Returns

(bool) true == ok

3.3.3.2 bool Page::generate (string *id*)

Generate a complete Book using Page::get() and write it to the disk.

Parameters

id	New book ID

Returns

(bool) true == ok

3.3.3.3 string Page::get ( )

Generate ciphertext page using the Rand class.

Returns

ciphertext

3.3.3.4 string Page::list ( )

Returns a list of valid books.

Returns

files

3.3.3.5 int Page::next (string *id*)

Returns the next unused page num for a book.

Parameters

id	Book ID

Returns

(int)pagenum

3.3.3.6 string Page::read (int *page*, string *id*)

Get the ciphertext page from a given book.

Parameters

page	Page num
id	Book ID

Returns

ciphertext

3.3.3.7 bool Page::write (int *page*, string *id*, string *ciphertext*)

The documentation for this class was generated from the following files:

- page.h
- page.cpp

3.4 Rand Class Reference

This class provides all secure random rutines.

`#include <rand.h>`

Public Member Functions

- Rand ()

  The Rand constructor generate a new random seed.
- unsigned long getTicks ()

  This function get the tick number from the CPU.
- void setSeed (float)

  Random seed setter.
- float getSeed ()

  Random sed getter.
- float genSeed ()
- char getChar ()
- char getLetter ()

  Get a random [A-Z] char.
- int getNumber (int)

  Get a random number.

3.4.1 Detailed Description

This class provides all secure random rutines.

3.4.2 Constructor & Destructor Documentation

3.4.2.1 Rand::Rand ( )

The Rand constructor generate a new random seed.

Returns

Rand object

3.4.3 Member Function Documentation

3.4.3.1 float Rand::genSeed ( )

Parameters

Generate	a new seed using some black magic

Returns

(float)seed

seed = (float)((usecs.tv_usec + getpid()) ^ (int(Rand::getTicks()) << 16) / 10000);

3.4.3.2 char Rand::getChar ( )

Parameters

Get	a random char

Returns

(char)rnd

3.4.3.3 char Rand::getLetter ( )

Get a random [A-Z] char.

Returns

(char)rnd

3.4.3.4 int Rand::getNumber (int *a*)

Get a random number.

Parameters

a	number len

Returns

(int)rnd

3.4.3.5 float Rand::getSeed ( )

Random sed getter.

Returns

(float)seed

3.4.3.6 unsigned long Rand::getTicks ( )

This function get the tick number from the CPU.

Returns

(ulong)tsc

In ix86 and amd64 uses RDTSC to get the low ticks value.

In ARMv6 and ARMv7 currently uses a gettimeofday()

3.4.3.7 void Rand::setSeed (float *a*)

Random seed setter.

Parameters

a	The new seed

The documentation for this class was generated from the following files:

- rand.h
- rand.cpp

3.5 Text Class Reference

Text and encoding related functions.

`#include <text.h>`

Public Member Functions

- void replaceAll (string &, const string &, const string &)

  Replace all characters from a string.
- void create (int, string, string, string)

  Set the Text object parameters.
- string print (int)

  Print a human friendly Text object.
- string encodeB26 (unsigned char *, long)

  Returns the text encoded in Base26.
- void decodeB26 (unsigned char *, string)

  Returns the text decoded in Base26.
- void parse (string)

  Parse and set the Text object with a formatted (headed) message.

Public Attributes

- string msg

  Message string.
- string book

  Book ID.
- string from

  Sender identificative string (3 letters, for example: AVD)
- int page

  Page num (int)

3.5.1 Detailed Description

Text and encoding related functions.

3.5.2 Member Function Documentation

3.5.2.1 void Text::create (int *page*, string *book*, string *from*, string *msg*)

Set the Text object parameters.

Parameters

page	Page num
book	Book ID
from	A sender id for identification
msg	The cleartext message

3.5.2.2 void Text::decodeB26 (unsigned char * *output*, string *text*)

Returns the text decoded in Base26.

Parameters

output	Text decoded
text	Encoded text

3.5.2.3 string Text::encodeB26 (unsigned char * *text*, long *len*)

Returns the text encoded in Base26.

Parameters

text	Text to encode
len	Text len

Returns

Encoded text (string)

3.5.2.4 void Text::parse (string *text*)

Parse and set the Text object with a formatted (headed) message.

Parameters

text	The formatted message string

3.5.2.5 string Text::print (int *spa*)

Print a human friendly Text object.

Parameters

spa	Space num

Returns

sout.str()

3.5.2.6 void Text::replaceAll (string & , const string & , const string &)

Replace all characters from a string.

replaceAll(str, from, to)

Parameters

str	The string pointer
from	Character to be replaced
to	Replace character

3.5.3 Member Data Documentation

3.5.3.1 string Text::book

Book ID.

3.5.3.2 string Text::from

Sender identificative string (3 letters, for example: AVD)

3.5.3.3 string Text::msg

Message string.

3.5.3.4 int Text::page

Page num (int)

The documentation for this class was generated from the following files:

- text.h
- text.cpp

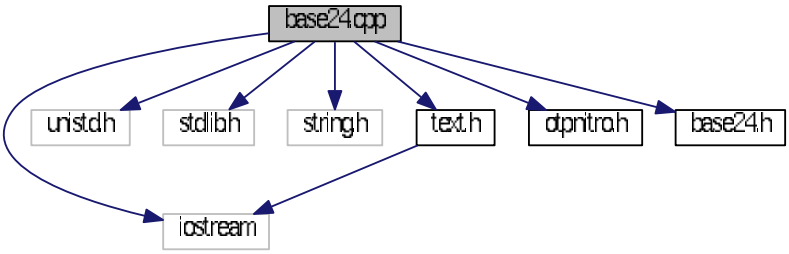
Chapter 4

File Documentation

4.1 base24.cpp File Reference

```
#include <iostream>
#include <unistd.h>
#include <stdlib.h>
#include <string.h>
#include "text.h"
#include "otpnitro.h"
#include "base24.h"
```

Include dependency graph for base24.cpp:



Functions

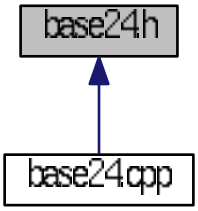
- int main (int argc, char **argv)

4.1.1 Function Documentation

4.1.1.1 int main (int *argc,* char ** *argv*)

4.2 base24.h File Reference

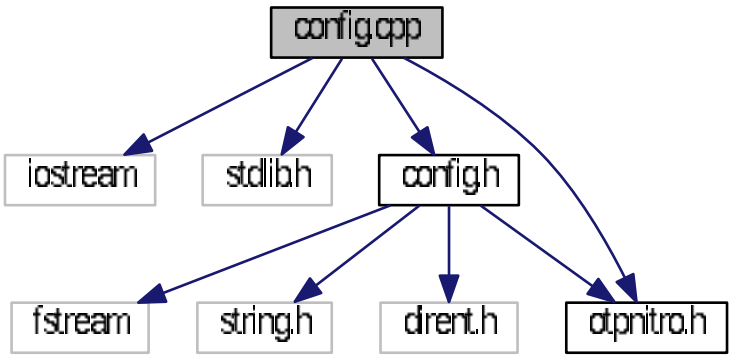
This graph shows which files directly or indirectly include this file:



4.3 config.cpp File Reference

```
#include <iostream>
#include <stdlib.h>
#include "config.h"
#include "otpnitro.h"
```

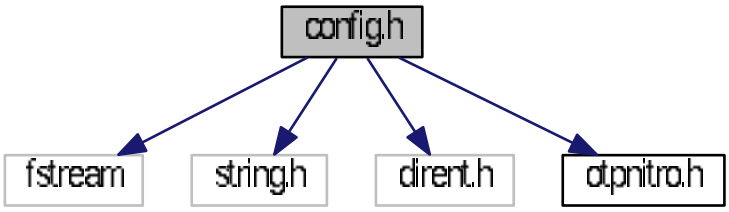
Include dependency graph for config.cpp:



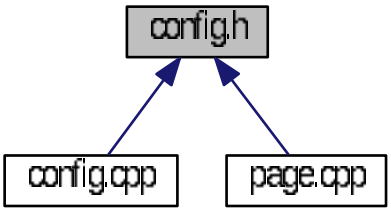
4.4 config.h File Reference

```
#include <fstream>
#include <string.h>
#include <dirent.h>
#include "otpnitro.h"
```

Include dependency graph for config.h:



This graph shows which files directly or indirectly include this file:



Classes

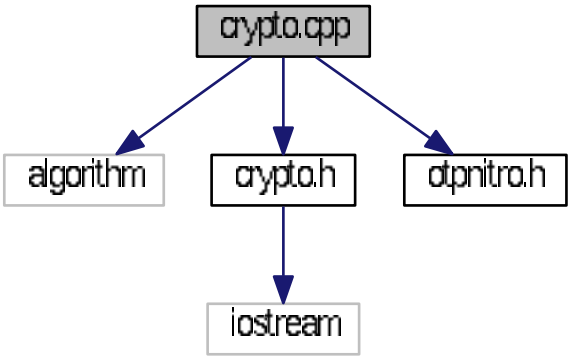
- class Config

  Configuration, path and pages management.

4.5 crypto.cpp File Reference

```
#include <algorithm>
#include "crypto.h"
#include "otpnitro.h"
```

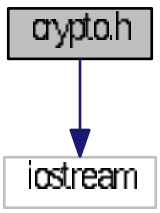
Include dependency graph for crypto.cpp:



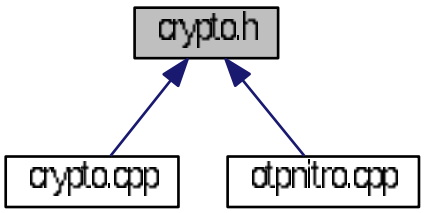
4.6 crypto.h File Reference

```
#include <iostream>
```

Include dependency graph for crypto.h:



This graph shows which files directly or indirectly include this file:



Classes

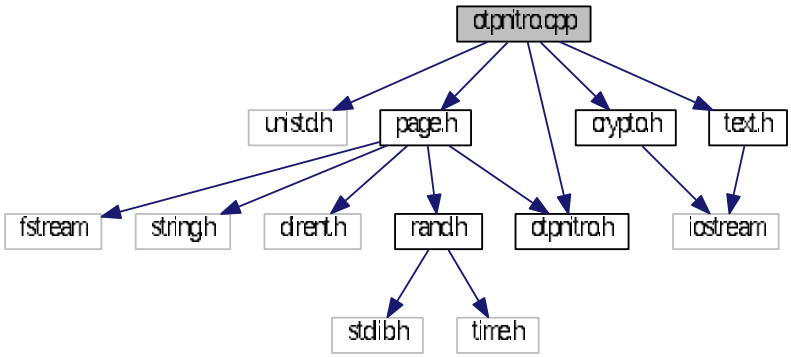
- class Crypto

  Crypt and decrypt class.

4.7 otpnitro.cpp File Reference

```
#include <unistd.h>
#include "page.h"
#include "crypto.h"
#include "text.h"
#include "otpnitro.h"
```

Include dependency graph for otpnitro.cpp:



Functions

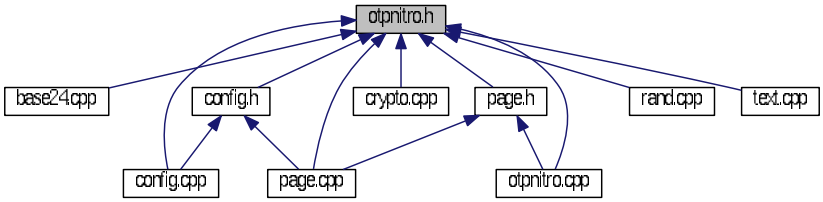
- int main (int argc, char **argv)

4.7.1 Function Documentation

4.7.1.1 int main (int *argc*, char ** *argv*)

4.8 otpnitro.h File Reference

This graph shows which files directly or indirectly include this file:



Macros

- #define VERSION 0.4
- #define SPACING 5
- #define MAX_PATH 256

4.8.1 Macro Definition Documentation

4.8.1.1 #define MAX_PATH 256

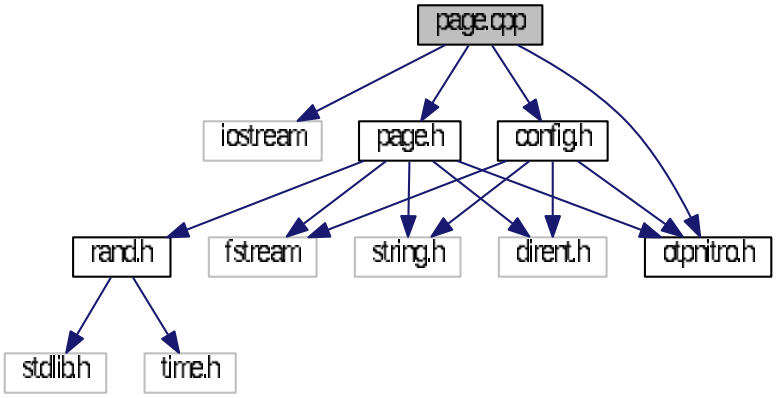
4.8.1.2 #define SPACING 5

4.8.1.3 #define VERSION 0.4

4.9 page.cpp File Reference

```
#include <iostream>
#include "page.h"
#include "config.h"
#include "otpnitro.h"
```

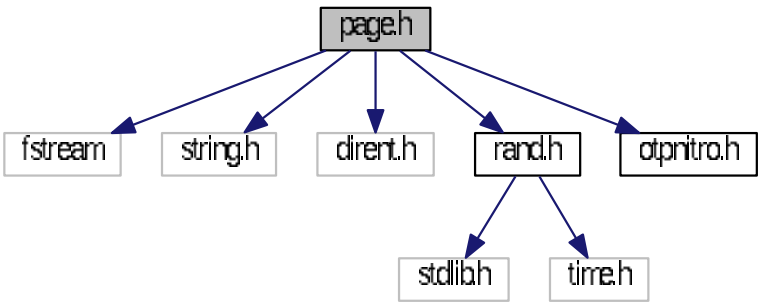
Include dependency graph for page.cpp:



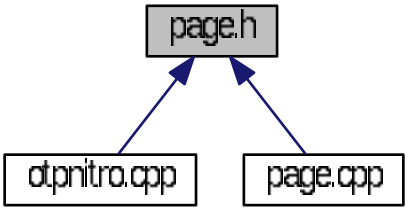
4.10 page.h File Reference

```
#include <fstream>
#include <string.h>
#include <dirent.h>
#include "rand.h"
#include "otpnitro.h"
```

Include dependency graph for page.h:



This graph shows which files directly or indirectly include this file:



Classes

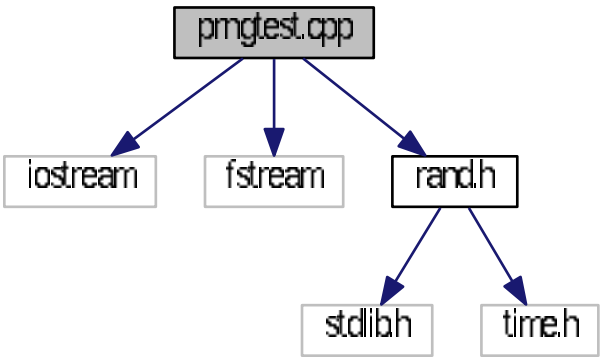
- class Page

  Page operations class.

4.11 prngtest.cpp File Reference

```
#include <iostream>
#include <fstream>
#include "rand.h"
```

Include dependency graph for prngtest.cpp:



Macros

- #define LOOPCNT 1000000

Functions

- int main ()

4.11.1 Macro Definition Documentation

4.11.1.1 #define LOOPCNT 1000000

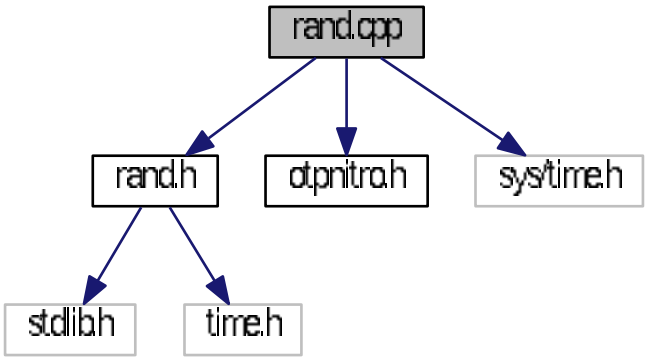
4.11.2 Function Documentation

4.11.2.1 int main ( )

4.12 rand.cpp File Reference

```
#include "rand.h"
#include "otpnitro.h"
#include <sys/time.h>
```

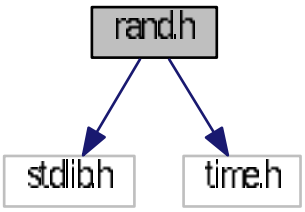
Include dependency graph for rand.cpp:



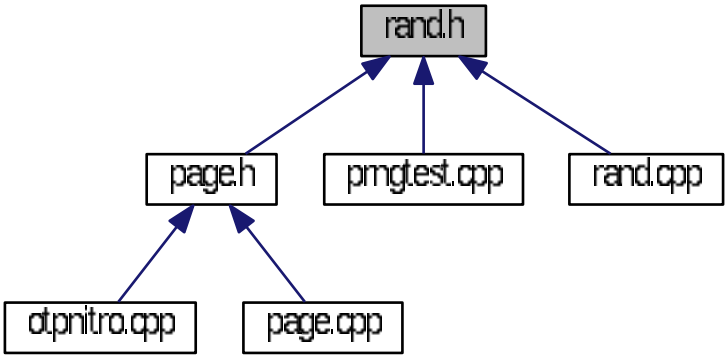
4.13 rand.h File Reference

```
#include <stdlib.h>
#include <time.h>
```

Include dependency graph for rand.h:



This graph shows which files directly or indirectly include this file:



Classes

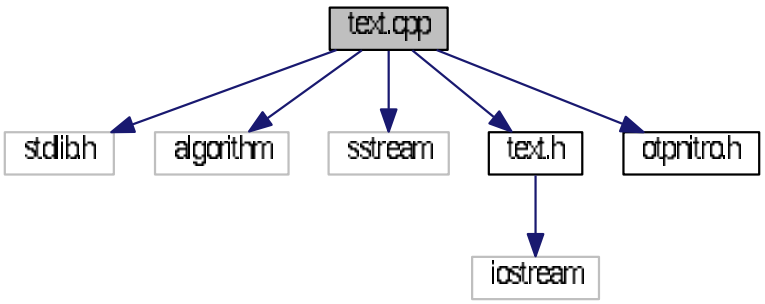
- class Rand

  This class provides all secure random rutines.

4.14 text.cpp File Reference

```
#include <stdlib.h>
#include <algorithm>
#include <sstream>
#include "text.h"
#include "otpnitro.h"
```

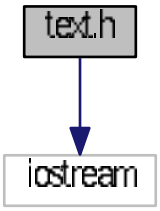
Include dependency graph for text.cpp:



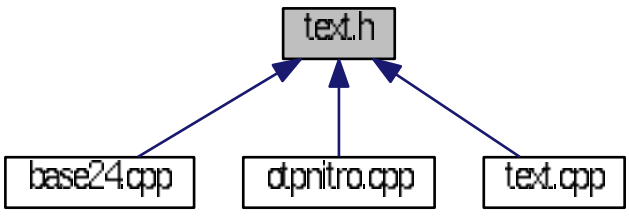
4.15 text.h File Reference

```
#include <iostream>
```

Include dependency graph for text.h:



This graph shows which files directly or indirectly include this file:



Classes

- class Text

  Text and encoding related functions.

Index