ASCII Obfuscation

0.0.1

Generated by Doxygen 1.8.13

Contents

Chapter 1

Todo List

Global main (int argC, char ∗∗argV)

Add a check on return code for 'translate_into_obscure' function (and take action depending on it)

Global randomize_value (char input, char ∗output, unsigned int ∗offset)

We have to revise the following lines to ensure they do what they are supposed to

Chapter 2

File Index

2.1 File List

Here is a list of all files with brief descriptions:

Chapter 3

File Documentation

3.1 deprecated_functions.c File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <stdbool.h>
#include "deprecated_functions.h"
```

Functions

- void shift_letters (void)

  Shift letters from upper case to lower case or from lower case to upper case.
- void character_bit_shifting (void)

  Bit shifts a string of character to make it unreadable.

Variables

- const char input_table [ ] = "I think you know my point about inline if operations : it only obfuscates the code.\n"
- const char bit_shifted_char_table [ ]

3.1.1 Function Documentation

3.1.1.1 character_bit_shifting()

```
void character_bit_shifting (
            void  )
```

Bit shifts a string of character to make it unreadable.

Definition at line 89 of file deprecated_functions.c.

References input_table.

3.1.1.2 shift_letters()

```
void shift_letters (
             void  )
```

Shift letters from upper case to lower case or from lower case to upper case.

Definition at line 40 of file deprecated_functions.c.

References SHIFT_VAL.

3.1.2 Variable Documentation

3.1.2.1 bit_shifted_char_table

```
const char bit_shifted_char_table[]
```

Initial value:

```
=
{
    0x92, 0x40, 0xe8, 0xd0, 0xd2, 0xdc, 0xd6, 0x40, 0xf2, 0xde, 0xea, 0x40, 0xd6, 0xdc, 0xde, 0xee, 0x40,
      0xda, 0xf2,
    0x40, 0xe0, 0xde, 0xd2, 0xdc, 0xe8, 0x40, 0xc2, 0xc4, 0xde, 0xea, 0xe8, 0x40, 0xd2, 0xdc, 0xd8, 0xd2,
      0xdc, 0xca,
    0x40, 0xd2, 0xcc, 0x40, 0xde, 0xe0, 0xca, 0xe4, 0xc2, 0xe8, 0xd2, 0xde, 0xdc, 0xe6, 0x40, 0x74, 0x40,
      0xd2, 0xe8,
    0x40, 0xde, 0xdc, 0xd8, 0xf2, 0x40, 0xde, 0xc4, 0xcc, 0xea, 0xe6, 0xc6, 0xc2, 0xe8, 0xca, 0xe6, 0x40,
      0xe8, 0xd0,
    0xca, 0x40, 0xc6, 0xde, 0xc8, 0xca, 0x5c
}
```

Definition at line 18 of file deprecated_functions.c.

3.1.2.2 input_table

```
const char input_table[] = "I think you know my point about inline if operations :  it only
obfuscates the code.\n"
```

Definition at line 17 of file deprecated_functions.c.

Referenced by character_bit_shifting().

3.2 deprecated_functions.h File Reference

Macros

- #define SHIFT_VAL ('a'-'A')

Functions

- void shift_letters (void)

  Shift letters from upper case to lower case or from lower case to upper case.
- void character_bit_shifting (void)

  Bit shifts a string of character to make it unreadable.

3.2.1 Macro Definition Documentation

3.2.1.1 SHIFT_VAL

```
#define SHIFT_VAL ('a'-'A')
```

Definition at line 6 of file deprecated_functions.h.

Referenced by shift_letters().

3.2.2 Function Documentation

3.2.2.1 character_bit_shifting()

```
void character_bit_shifting (
            void  )
```

Bit shifts a string of character to make it unreadable.

Definition at line 89 of file deprecated_functions.c.

References input_table.

3.2.2.2 shift_letters()

```
void shift_letters (
            void  )
```

Shift letters from upper case to lower case or from lower case to upper case.

Definition at line 40 of file deprecated_functions.c.

References SHIFT_VAL.

3.3 functions.c File Reference

```
#include "functions.h"
#include "main.h"
#include <stdlib.h>
#include <iso646.h>
```

Functions

- int randomize_value (char input, char *output, unsigned int *offset)

  function implementation for ascii_obfuscation
- int translate_into_obscure (char *input, unsigned int input_length, unsigned char *output, unsigned int *p↩ _output_length)

  Translation loop function.

3.3.1 Function Documentation

3.3.1.1 randomize_value()

```
int randomize_value (
            char input,
            char * output,
            unsigned int * offset )
```

function implementation for ascii_obfuscation

function definition for `ascii_obfuscation`

functions.c

Note

Created by vince on 08/02/2020.

Todo We have to revise the following lines to ensure they do what they are supposed to

Definition at line 43 of file functions.c.

References RETURN_OK.

Referenced by TEST_randomize_value(), and translate_into_obscure().

3.3.1.2 translate_into_obscure()

```
int translate_into_obscure (
            char * input,
            unsigned int input_length,
            unsigned char * output,
            unsigned int * p_output_length )
```

Translation loop function.

Translation loop function Parameters :

- [in] input String that contains the original message
- [in] input_length Length of input string
- [out] output Buffer in which we would write the resulting string
- [out] p_output_length Pointer to length of output string Return : An int value :
  - RETURN_OK (0) if everything is OK
  - GENERIC_ERROR (-1) if process ends in error

If 'output' buffer is 'NULL' we might get in trouble trying to set values to a random memory location
=> We would better exit the function as soon as possible with an error code to inform the caller

If there is no input buffer passed in argument, use default string 'message_str'

Definition at line 86 of file functions.c.

References GENERIC_ERROR, message_str, randomize_value(), and RETURN_OK.

Referenced by main(), and TEST_randomize_value().

3.4 functions.h File Reference

Functions

- int randomize_value (char input, char *output, unsigned int *offset)
  function definition for `ascii_obfuscation`
- int translate_into_obscure (char *input, unsigned int input_length, unsigned char *output, unsigned int *p_output_length)
  Translation loop function.

3.4.1 Function Documentation

3.4.1.1 randomize_value()

```
int randomize_value (
            char input,
            char * output,
            unsigned int * offset )
```

function definition for `ascii_obfuscation`

functions.h

Note

Created by vince on 08/02/2020. Make 2 randomized letters out of 1 letter input

Parameters

in	*input*	Input character to be randomized
out	*output*	Buffer in which we would write the resulting 2 characters
out	*offset*	Offset of output characters in output buffer

Returns

An int value :

- RETURN_OK (0) if everything is OK
- GENERIC_ERROR (-1) if process ends in error

function definition for `ascii_obfuscation`

functions.c

Note

Created by vince on 08/02/2020.

Todo We have to revise the following lines to ensure they do what they are supposed to

Definition at line 43 of file functions.c.

References RETURN_OK.

Referenced by TEST_randomize_value(), and translate_into_obscure().

3.4.1.2 translate_into_obscure()

```
int translate_into_obscure (
            char * input,
            unsigned int input_length,
            unsigned char * output,
            unsigned int * p_output_length )
```

Translation loop function.

Parameters

in	*input*	String that contains the original message
in	*input_length*	Length of input string
out	*output*	Buffer in which we would write the resulting string
out	*p_output_length*	Pointer to length of output string

Returns

An int value :

- RETURN_OK (0) if everything is OK
- GENERIC_ERROR (-1) if process ends in error

Translation loop function Parameters :

- [in] input String that contains the original message
- [in] input_length Length of input string
- [out] output Buffer in which we would write the resulting string
- [out] p_output_length Pointer to length of output string Return : An int value :
  - RETURN_OK (0) if everything is OK
  - GENERIC_ERROR (-1) if process ends in error

If 'output' buffer is 'NULL' we might get in trouble trying to set values to a random memory location
=> We would better exit the function as soon as possible with an error code to inform the caller

If there is no input buffer passed in argument, use default string 'message_str'

Definition at line 86 of file functions.c.

References GENERIC_ERROR, message_str, randomize_value(), and RETURN_OK.

Referenced by main(), and TEST_randomize_value().

3.5 invert_ascii.c File Reference

```
#include <stdio.h>
```

Functions

- int main (void)

3.5.1 Function Documentation

3.5.1.1 main()

```
int main (
            void  )
```

Definition at line 3 of file invert_ascii.c.

3.6 main.c File Reference

```
#include <stdio.h>
#include <string.h>
#include "functions.h"
#include "main.h"
```

Functions

- int main (int argC, char **argV)
  Main program function.

Variables

- const char title_str [ ]
  Title string to ebe displayed in console.

3.6.1 Function Documentation

3.6.1.1 main()

```
int main (
            int argC,
            char ** argV )
```

Main program function.

Parameters

in	*argC*	Argument number
in	*argV*	Pointer onto input argument strings

Returns

RETURN_OK (0) in case of successful execution

Conditional behavior of program:

- if there are arguments in the program call, just translates the input string and exit (or print error message)
- if there is no argument, go to the infinite loop to use the program until asked to quit

Remarks

show how many arguments are passed to the program (following line is only here for debug)
- Just print the input string for now (translation part of the program is not finished yet)

- Perhaps keep a print of which option has been chosen in the end ('encode' or 'decode' for example)

Todo Add a check on return code for 'translate_into_obscure' function (and take action depending on it)

Definition at line 50 of file main.c.

References forever, RETURN_OK, title_str, and translate_into_obscure().

3.6.2 Variable Documentation

3.6.2.1 title_str

```
const char title_str[]
```

Initial value:

```
=
    "                                  _/  _/                _/            _/_/
                    _/       _/                      \n"
    "    _/_/_/     _/_/_/    _/_/_/                  _/_/    _/_/_/      _/      _/    _/    _/_/_/    _/_/_/
      _/_/_/   _/_/_/_/        _/_/    _/_/_/ \n"
    " _/    _/  _/_/      _/       _/  _/      _/    _/  _/    _/  _/_/_/_/  _/    _/  _/_/      _/
      _/    _/    _/      _/  _/    _/  _/    _/\n"
    "_/    _/      _/_/  _/       _/  _/      _/    _/  _/    _/    _/      _/    _/      _/_/  _/
      _/    _/    _/      _/  _/    _/  _/    _/ \n"
    " _/_/_/  _/_/_/      _/_/_/  _/  _/       _/_/    _/_/_/      _/        _/_/_/  _/_/_/      _/_/_/
      _/_/_/      _/_/  _/    _/_/    _/    _/  \n"
    "                                 _/_/_/_/
                                      \n"
```

Title string to ebe displayed in console.

ascii_obfuscation

Definition at line 20 of file main.c.

Referenced by main().

3.7 main.h File Reference

Macros

- #define UPPERCASE_LOWERCASE_SHIFT ('a'-'A')

  ascii_obfuscation program header file for specific definition share
- #define ever (;;)
- #define forever for ever

Enumerations

- enum error_codes { GENERIC_ERROR = -1, RETURN_OK = 0 }

  Define custom/personal error codes to avoid using magic numbers.

Variables

- const char vowels [6]

  Global utility elements for `ascii_obfuscation`
- const char consonants [20]

  Consonants table.
- const char message_str [0xFF]

  Default test message string for character manipulation.

3.7.1 Macro Definition Documentation

3.7.1.1 ever

```
#define ever (;;)
```

Definition at line 24 of file main.h.

3.7.1.2 forever

```
#define forever for ever
```

Definition at line 25 of file main.h.

Referenced by main().

3.7.1.3 UPPERCASE_LOWERCASE_SHIFT

```
#define UPPERCASE_LOWERCASE_SHIFT ('a'-'A')
```

`ascii_obfuscation` program header file for specific definition share

main.h

Note

Created by vince on 09/02/2020.difference between uppercase and lower case letters

Definition at line 21 of file main.h.

3.7.2 Enumeration Type Documentation

3.7.2.1 error_codes

```
enum error_codes
```

Define custom/personal error codes to avoid using magic numbers.

Enumerator

GENERIC_ERROR	General purpose error ; no particular details to share.
RETURN_OK	All good !

Definition at line 32 of file main.h.

3.7.3 Variable Documentation

3.7.3.1 consonants

```
const char consonants[20]
```

Consonants table.

Definition at line 20 of file utils.c.

3.7.3.2 message_str

```
const char message_str[0xFF]
```

Default test message string for character manipulation.

Definition at line 24 of file utils.c.

Referenced by translate_into_obscure().

3.7.3.3 vowels

```
const char vowels[6]
```

Global utility elements for `ascii_obfuscation`

utils.c

Note

Created by vince on 09/02/2020.vowels table

Definition at line 18 of file utils.c.

3.8 unit_tests.c File Reference

```
#include "unity.h"
#include "functions.h"
#include <string.h>
#include <iso646.h>
```

Functions

- void setUp (void)

  Unit test implementation for `ascii_obfuscation`
- void tearDown (void)
- void TEST_randomize_value (void)
- int main (void)

3.8.1 Function Documentation

3.8.1.1 main()

```
int main (
            void  )
```

Definition at line 63 of file unit_tests.c.

References TEST_randomize_value().

3.8.1.2 setUp()

```
void setUp (
            void  )
```

Unit test implementation for `ascii_obfuscation`

unit_tests.c

Note

Created by vince on 08/02/2020.

Definition at line 21 of file unit_tests.c.

3.8.1.3 tearDown()

```
void tearDown (
            void  )
```

Definition at line 26 of file unit_tests.c.

3.8.1.4 TEST_randomize_value()

```
void TEST_randomize_value (
            void  )
```

Definition at line 35 of file unit_tests.c.

References randomize_value(), and translate_into_obscure().

Referenced by main().

3.9 utils.c File Reference

```
#include "main.h"
```

Variables

- const char vowels [6] = { 'a', 'e', 'i', 'o', 'u', 'y' }

  Global utility elements for `ascii_obfuscation`
- const char consonants [20]

  Consonants table.
- const char message_str [0xFF]

  Default test message string for character manipulation.

3.9.1 Variable Documentation

3.9.1.1 consonants

```
const char consonants[20]
```

Initial value:

```
= { 'b', 'c', 'd', 'f', 'g', 'h', 'j', 'k', 'l',
                              'm', 'n', 'p', 'q', 'r', 's', 't', 'v', 'w',
                              'x', 'z' }
```

Consonants table.

Definition at line 20 of file utils.c.

3.9.1.2 message_str

```
const char message_str[0xFF]
```

Initial value:

```
=
        "I think you know my point about inline if operations : it only obfuscates the code.\0"
```

Default test message string for character manipulation.

Definition at line 24 of file utils.c.

Referenced by translate_into_obscure().

3.9.1.3 vowels

```
const char vowels[6] = { 'a', 'e', 'i', 'o', 'u', 'y' }
```

Global utility elements for `ascii_obfuscation`

utils.c

Note

Created by vince on 09/02/2020.vowels table

Definition at line 18 of file utils.c.

Index