Phone Firewall Reference Manual

v0.01

Generated by Doxygen 1.5.4

Thu Nov 6 10:42:10 2008

Contents

Chapter 1

Phone Firewall Data Structure Index

1.1 Phone Firewall Data Structures

Here are the data structures with brief descriptions:

Chapter 2

Phone Firewall File Index

2.1 Phone Firewall File List

Here is a list of all files with brief descriptions:

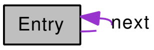
Chapter 3

Phone Firewall Data Structure Documentation

3.1 Entry Struct Reference

```
#include <libphonefirewall.h>
```

Collaboration diagram for Entry:



Data Fields

- int country_code
- int area_code
- unsigned long long number
- char ∗ name
- char ∗ reason
- int priority
- struct Entry ∗ next

3.1.1 Detailed Description

Definition at line 60 of file libphonefirewall.h.

3.1.2 Field Documentation

3.1.2.1 int Entry::country_code

Definition at line 61 of file libphonefirewall.h.

Referenced by check_entry(), evaluate_stmt(), find_entry(), and insert_into_list().

3.1.2.2 int Entry::area_code

Definition at line 62 of file libphonefirewall.h.

Referenced by check_entry(), evaluate_stmt(), find_entry(), and insert_into_list().

3.1.2.3 unsigned long long Entry::number

Definition at line 63 of file libphonefirewall.h.

Referenced by check_entry(), evaluate_stmt(), find_entry(), and insert_into_list().

3.1.2.4 char∗ Entry::name

Definition at line 64 of file libphonefirewall.h.

Referenced by find_entry(), and insert_into_list().

3.1.2.5 char∗ Entry::reason

Definition at line 65 of file libphonefirewall.h.

Referenced by find_entry(), and insert_into_list().

3.1.2.6 int Entry::priority

Definition at line 66 of file libphonefirewall.h.

Referenced by check_entry(), evaluate_stmt(), and find_entry().

3.1.2.7 struct Entry∗ Entry::next `[read]`

Definition at line 67 of file libphonefirewall.h.

Referenced by insert_into_list().

The documentation for this struct was generated from the following file:

- libphonefirewall.h

3.2 entry Struct Reference

Includes all informations for an entry.

```
#include <libphonefirewall.h>
```

3.2.1 Detailed Description

Includes all informations for an entry.

The struct which includes all information about entries (black- and whitelist).

The documentation for this struct was generated from the following file:

- libphonefirewall.h

Chapter 4

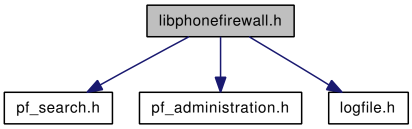
Phone Firewall File Documentation

4.1 libphonefirewall.h File Reference

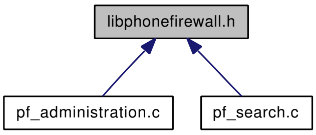
API of the phone firewall.

```
#include "pf_search.h"
#include "pf_administration.h"
#include "logfile.h"
```

Include dependency graph for libphonefirewall.h:



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



Data Structures

- struct Entry

Defines

- #define PRIO_ALL -999
- #define DB_FILE "db/phone-firewall.db"
- #define STMT_SIZE 1024
- #define MAX_LINE_LENGTH 512

- #define TB_COUNTRYCODE "countrycode"
- #define TB_AREACODE "areacode"
- #define TB_NUMBER "number"
- #define TB_NAME "name"
- #define TB_REASON "reason"
- #define TB_PRIORITY "priority"
- #define BLACKLIST_FLAG 0x01
- #define WHITELIST_FLAG 0x02
- #define COMPONENT_NAME "phone firewall"

4.1.1 Detailed Description

API of the phone firewall.

Author:

Alex Oberhauser

The header file of the Phone Firewall. Blocks or accepts incoming phone calls, so it's possible to prevent disturbing phone calls. Provides a API which can used by other application to build nice programs.

Definition in file libphonefirewall.h.

4.1.2 Define Documentation

4.1.2.1 #define BLACKLIST_FLAG 0x01

Definition at line 48 of file libphonefirewall.h.

Referenced by check_entry().

4.1.2.2 #define COMPONENT_NAME "phone firewall"

Definition at line 51 of file libphonefirewall.h.

4.1.2.3 #define DB_FILE "db/phone-firewall.db"

Definition at line 37 of file libphonefirewall.h.

Referenced by add_entry(), change_name(), change_number(), change_priority(), change_reason(), check_entry(), get_entry_by_name(), get_entry_by_number(), get_entry_by_reason(), and rm_entry().

4.1.2.4 #define MAX_LINE_LENGTH 512

Definition at line 39 of file libphonefirewall.h.

Referenced by add_entry(), change_name(), change_number(), change_priority(), change_reason(), check_entry(), get_entry_by_name(), get_entry_by_number(), get_entry_by_reason(), and rm_entry().

4.1.2.5 #define PRIO_ALL -999

Definition at line 36 of file libphonefirewall.h.

Referenced by add_entry(), and evaluate_stmt().

4.1.2.6 #define STMT_SIZE 1024

Definition at line 38 of file libphonefirewall.h.

Referenced by add_entry(), change_name(), change_number(), change_priority(), change_reason(), check_entry(), get_entry_by_name(), get_entry_by_number(), get_entry_by_reason(), and rm_entry().

4.1.2.7 #define TB_AREACODE "areacode"

Definition at line 42 of file libphonefirewall.h.

Referenced by add_entry(), change_name(), change_number(), change_priority(), change_reason(), check_entry(), evaluate_stmt(), find_entry(), get_entry_by_name(), get_entry_by_number(), get_entry_by_reason(), and rm_entry().

4.1.2.8 #define TB_COUNTRYCODE "countrycode"

Definition at line 41 of file libphonefirewall.h.

Referenced by add_entry(), change_name(), change_number(), change_priority(), change_reason(), check_entry(), evaluate_stmt(), find_entry(), get_entry_by_name(), get_entry_by_number(), get_entry_by_reason(), and rm_entry().

4.1.2.9 #define TB_NAME "name"

Definition at line 44 of file libphonefirewall.h.

Referenced by add_entry(), change_name(), find_entry(), get_entry_by_name(), get_entry_by_number(), and get_entry_by_reason().

4.1.2.10 #define TB_NUMBER "number"

Definition at line 43 of file libphonefirewall.h.

Referenced by add_entry(), change_name(), change_number(), change_priority(), change_reason(), check_entry(), evaluate_stmt(), find_entry(), get_entry_by_name(), get_entry_by_number(), get_entry_by_reason(), and rm_entry().

4.1.2.11 #define TB_PRIORITY "priority"

Definition at line 46 of file libphonefirewall.h.

Referenced by add_entry(), change_priority(), check_entry(), evaluate_stmt(), find_entry(), get_entry_by_name(), get_entry_by_number(), and get_entry_by_reason().

4.1.2.12 #define TB_REASON "reason"

Definition at line 45 of file libphonefirewall.h.

Referenced by add_entry(), change_reason(), find_entry(), get_entry_by_name(), get_entry_by_number(), and get_entry_by_reason().

4.1.2.13 #define WHITELIST_FLAG 0x02

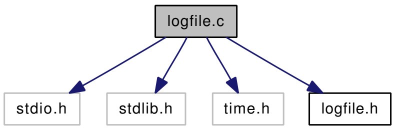
Definition at line 49 of file libphonefirewall.h.

Referenced by add_entry(), change_name(), change_number(), change_priority(), change_reason(), check_entry(), get_entry_by_name(), get_entry_by_number(), get_entry_by_reason(), and rm_entry().

4.2 logfile.c File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <time.h>
#include "logfile.h"
```

Include dependency graph for logfile.c:



Functions

- char ∗ asctime (const struct tm ∗timeptr)
- int write_logentry (char ∗msg, char ∗component, int flag)

4.2.1 Function Documentation

4.2.1.1 char∗ asctime (const struct tm ∗ *timeptr*)

Compounds a humand readable date and time string.

Parameters:

timeptr A pointer to the actual time.

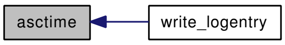
Returns:

The date and time as a string.

Definition at line 33 of file logfile.c.

Referenced by write_logentry().

Here is the caller graph for this function:



4.2.1.2 int write_logentry (char ∗ *msg*, char ∗ *component*, int *flag*)

Writes a logfile enty.

Parameters:

msg The message which should be written in the logfile.

component The program which calls the write_logentry function, for example "phonefirewall"

flag What message should be written. Use the defined flags.

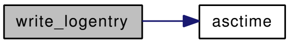
Returns:

-1 if something fails, otherwise 0

Definition at line 58 of file logfile.c.

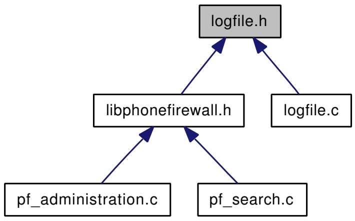
References asctime(), ERR_FLAG, INFO_FLAG, LOGFILE, MAX_ENTRY_LENGTH, UNKNOWN, and WARN_FLAG.

Here is the call graph for this function:



4.3 logfile.h File Reference

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



Defines

- #define LOGFILE "/tmp/phonefirewall.log"
- #define MAX_ENTRY_LENGTH 128
- #define UNKNOWN 0
- #define ERR_FLAG 1
- #define WARN_FLAG 2
- #define INFO_FLAG 3
- #define UNKNOWN_LOG(m) write_logentry(m, COMPONENT_NAME, UNKNOWN_FLAG)
- #define ERR_LOG(m) write_logentry(m, COMPONENT_NAME, ERR_FLAG)
- #define WARN_LOG(m) write_logentry(m, COMPONENT_NAME, WARN_FLAG)
- #define INFO_LOG(m) write_logentry(m, COMPONENT_NAME, INFO_FLAG)

Functions

- int write_logentry (char ∗msg, char ∗component, int flag)

4.3.1 Define Documentation

4.3.1.1 #define ERR_FLAG 1

Definition at line 25 of file logfile.h.

Referenced by write_logentry().

4.3.1.2 #define ERR_LOG(m) write_logentry(m, COMPONENT_NAME, ERR_FLAG)

Definition at line 30 of file logfile.h.

Referenced by add_entry(), change_name(), change_number(), change_priority(), change_reason(), check_entry(), get_entry_by_name(), get_entry_by_number(), get_entry_by_reason(), and rm_entry().

4.3.1.3 #define INFO_FLAG 3

Definition at line 27 of file logfile.h.

Referenced by write_logentry().

4.3.1.4 #define INFO_LOG(m) write_logentry(m, COMPONENT_NAME, INFO_FLAG)

Definition at line 32 of file logfile.h.

Referenced by check_entry().

4.3.1.5 #define LOGFILE "/tmp/phonefirewall.log"

Definition at line 21 of file logfile.h.

Referenced by write_logentry().

4.3.1.6 #define MAX_ENTRY_LENGTH 128

Definition at line 22 of file logfile.h.

Referenced by write_logentry().

4.3.1.7 #define UNKNOWN 0

Definition at line 24 of file logfile.h.

Referenced by write_logentry().

4.3.1.8 #define UNKNOWN_LOG(m) write_logentry(m, COMPONENT_NAME, UNKNOWN_FLAG)

Definition at line 29 of file logfile.h.

4.3.1.9 #define WARN_FLAG 2

Definition at line 26 of file logfile.h.

Referenced by write_logentry().

4.3.1.10 #define WARN_LOG(m) write_logentry(m, COMPONENT_NAME, WARN_FLAG)

Definition at line 31 of file logfile.h.

4.3.2 Function Documentation

4.3.2.1 int write_logentry (char ∗ *msg*, char ∗ *component*, int *flag*)

Writes a logfile enty.

Parameters:

msg The message which should be written in the logfile.

component The program which calls the write_logentry function, for example "phonefirewall"

flag What message should be written. Use the defined flags.

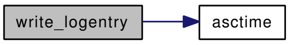
Returns:

-1 if something fails, otherwise 0

Definition at line 58 of file logfile.c.

References asctime(), ERR_FLAG, INFO_FLAG, LOGFILE, MAX_ENTRY_LENGTH, UNKNOWN, and WARN_FLAG.

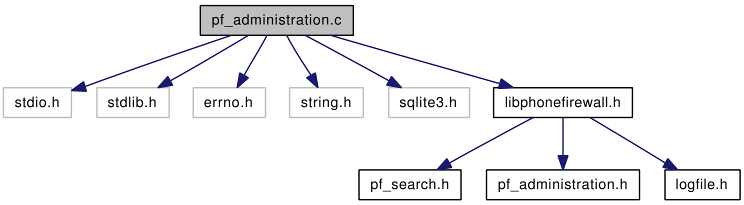
Here is the call graph for this function:



4.4 pf_administration.c File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <errno.h>
#include <string.h>
#include <sqlite3.h>
#include "libphonefirewall.h"
```

Include dependency graph for pf_administration.c:



Functions

- int evaluate_stmt (sqlite3_stmt ∗pp_stmt, struct Entry ∗p_entry)
- int add_entry (int country_code, int area_code, unsigned long long number, char ∗name, char ∗reason, int priority, int listflag)
- int rm_entry (int country_code, int area_code, unsigned long long number, int listflag)
- int check_entry (int country_code, int area_code, unsigned long long number, int priority, int listflag)
- int change_name (int country_code, int area_code, unsigned long long number, char ∗new_name, int listflag)
- int change_number (int country_code, int area_code, unsigned long long number, int new_country_-code, int new_area_code, unsigned long long new_number, int listflag)
- int change_reason (int country_code, int area_code, unsigned long long number, char ∗new_reason, int listflag)
- int change_priority (int country_code, int area_code, unsigned long long number, int new_priority, int listflag)

4.4.1 Function Documentation

4.4.1.1 int add_entry (int *country_code*, int *area_code*, unsigned long long *number*, char ∗ *name*, char ∗ *reason*, int *priority*, int *listflag*)

Add a number to the blacklist/whitelist. The number will be blocked after that.

Parameters:

country_code The country code (for example 39 for Italy, 43 for Austria, and so one)

area_code The area code which indicates your mobile operator.

number The telephone number of the person (without country and area code.

name The name of the person.

reason Why you have blocked this person.

priority Gives the entry a priority. 0 is standard. If the priority is higher the value will be also blocked/accepted if a higher priority is choosen.

listflag A flag, which indicates if you would use the blacklist (BLACKLIST_FLAG) or the whitelist (WHITELIST_FLAG).

The value "PRIO_ALL" stands for all priorities.

Returns:

If all goes well 0 (zero) otherwise -1 and the errno variable will be set..

Definition at line 74 of file pf_administration.c.

References DB_FILE, ERR_LOG, MAX_LINE_LENGTH, PRIO_ALL, STMT_SIZE, TB_AREACODE, TB_COUNTRYCODE, TB_NAME, TB_NUMBER, TB_PRIORITY, TB_REASON, and WHITELIST_FLAG.

4.4.1.2 int change_name (int *country_code*, int *area_code*, unsigned long long *number*, char ∗ *new_name*, int *listflag*)

Changes the name of the entry. For the unique identification you need to enter the country code, area code and the number because this tripple identifies the entry.

Parameters:

country_code The country code (for example 39 for Italy, 43 for Austria, and so one)

area_code The area code which indicates your mobile operator.

number The telephone number of the person (without country and area code.

new_name The new name

listflag A flag, which indicates if you would use the blacklist (BLACKLIST_FLAG) or the whitelist (WHITELIST_FLAG).

Returns:

If the number was changed 1, otherwise 0.

Definition at line 227 of file pf_administration.c.

References DB_FILE, ERR_LOG, MAX_LINE_LENGTH, STMT_SIZE, TB_AREACODE, TB_COUNTRYCODE, TB_NAME, TB_NUMBER, and WHITELIST_FLAG.

4.4.1.3 int change_number (int *country_code*, int *area_code*, unsigned long long *number*, int *new_country_code*, int *new_area_code*, unsigned long long *new_number*, int *listflag*)

Changes the number of the entry. You need to enter the country code, area code and the number because this tripple identifies the entry.

Parameters:

country_code The country code (for example 39 for Italy, 43 for Austria, and so one)

area_code The area code which indicates your mobile operator.

number The telephone number of the person (without country and area code.

new_country_code The new country code.

new_area_code The new area code.

new_number The new number.

listflag A flag, which indicates if you would use the blacklist (BLACKLIST_FLAG) or the whitelist (WHITELIST_FLAG).

Returns:

If the number was changed 1, otherwise 0.

Definition at line 266 of file pf_administration.c.

References DB_FILE, ERR_LOG, MAX_LINE_LENGTH, STMT_SIZE, TB_AREACODE, TB_COUNTRYCODE, TB_NUMBER, and WHITELIST_FLAG.

4.4.1.4 int change_priority (int *country_code*, int *area_code*, unsigned long long *number*, int *new_priority*, int *listflag*)

Changes the priority of the entry. For the unique identification you need to enter the country code, area code and the number because this tripple identifies the entry.

Parameters:

country_code The country code (for example 39 for Italy, 43 for Austria, and so one)

area_code The area code which indicates your mobile operator.

number The telephone number of the person (without country and area code.

new_priority The new priority.

listflag A flag, which indicates if you would use the blacklist (BLACKLIST_FLAG) or the whitelist (WHITELIST_FLAG).

Returns:

If the number was changed 1, otherwise 0.

Definition at line 346 of file pf_administration.c.

References DB_FILE, ERR_LOG, MAX_LINE_LENGTH, STMT_SIZE, TB_AREACODE, TB_COUNTRYCODE, TB_NUMBER, TB_PRIORITY, and WHITELIST_FLAG.

4.4.1.5 int change_reason (int *country_code*, int *area_code*, unsigned long long *number*, char ∗ *new_reason*, int *listflag*)

Changes the reason of the entry. For the unique identification you need to enter the country code, area code and the number because this tripple identifies the entry.

Parameters:

country_code The country code (for example 39 for Italy, 43 for Austria, and so one)

area_code The area code which indicates your mobile operator.

number The telephone number of the person (without country and area code.

new_reason The new reason.

listflag A flag, which indicates if you would use the blacklist (BLACKLIST_FLAG) or the whitelist (WHITELIST_FLAG).

Returns:

If the number was changed 1, otherwise 0.

Definition at line 307 of file pf_administration.c.

References DB_FILE, ERR_LOG, MAX_LINE_LENGTH, STMT_SIZE, TB_AREACODE, TB_COUNTRYCODE, TB_NUMBER, TB_REASON, and WHITELIST_FLAG.

4.4.1.6 int check_entry (int *country_code*, int *area_code*, unsigned long long *number*, int *priority*, int *listflag*)

Checks if a number is on the blacklist/whitelist.

Parameters:

country_code The country code (for example 39 for Italy, 43 for Austria, and so one)

area_code The area code which indicates your mobile operator.

number The telephone number of the person (without country and area code.

priority Gives the entry a priority. 0 is standard. If the priority is higher the value will be also blocked/accepted if a higher priority is choosen.

listflag A flag, which indicates if you would use the blacklist (BLACKLIST_FLAG) or the whitelist (WHITELIST_FLAG).

The value "PRIO_ALL" stands for all priorities.

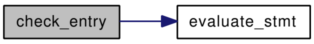
Returns:

If the number was found 1, otherwise 0.

Definition at line 155 of file pf_administration.c.

References Entry::area_code, BLACKLIST_FLAG, Entry::country_code, DB_FILE, ERR_LOG, evaluate_stmt(), INFO_LOG, MAX_LINE_LENGTH, Entry::number, Entry::priority, STMT_SIZE, TB_AREACODE, TB_COUNTRYCODE, TB_NUMBER, TB_PRIORITY, and WHITELIST_FLAG.

Here is the call graph for this function:



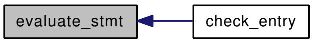
4.4.1.7 int evaluate_stmt (sqlite3_stmt * *pp_stmt*, struct Entry * *p_entry*)

Definition at line 27 of file pf_administration.c.

References Entry::area_code, Entry::country_code, Entry::number, PRIO_ALL, Entry::priority, TB_AREACODE, TB_COUNTRYCODE, TB_NUMBER, and TB_PRIORITY.

Referenced by check_entry().

Here is the caller graph for this function:



4.4.1.8 int rm_entry (int *country_code*, int *area_code*, unsigned long long *number*, int *listlfag*)

Removes a number from the blacklist/whitelist.

Parameters:

country_code The country code (for example 39 for Italy, 43 for Austria, and so one)

area_code The area code which indicates your mobile operator.

number The number which will be deleted.

listflag A flag, which indicates if you would use the blacklist (BLACKLIST_FLAG) or the whitelist (WHITELIST_FLAG).

Returns:

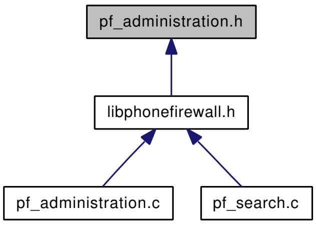
If all goes right 0, otherwise an error code.

Definition at line 119 of file pf_administration.c.

References DB_FILE, ERR_LOG, MAX_LINE_LENGTH, STMT_SIZE, TB_AREACODE, TB_-COUNTRYCODE, TB_NUMBER, and WHITELIST_FLAG.

4.5 pf_administration.h File Reference

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



Functions

- int add_entry (int country_code, int area_code, unsigned long long number, char ∗name, char ∗reason, int priority, int listflag)
- int rm_entry (int country_code, int area_code, unsigned long long number, int listlfag)
- int check_entry (int country_code, int area_code, unsigned long long number, int priority, int listflag)
- int change_name (int country_code, int area_code, unsigned long long number, char ∗new_name, int listflag)
- int change_number (int country_code, int area_code, unsigned long long number, int new_country_-code, int new_area_code, unsigned long long new_number, int listflag)
- int change_reason (int country_code, int area_code, unsigned long long number, char ∗new_reason, int listflag)
- int change_priority (int country_code, int area_code, unsigned long long number, int new_priority, int listflag)

4.5.1 Function Documentation

4.5.1.1 int add_entry (int *country_code*, int *area_code*, unsigned long long *number*, char ∗ *name*, char ∗ *reason*, int *priority*, int *listflag*)

Add a number to the blacklist/whitelist. The number will be blocked after that.

Parameters:

country_code The country code (for example 39 for Italy, 43 for Austria, and so one)

area_code The area code which indicates your mobile operator.

number The telephone number of the person (without country and area code.

name The name of the person.

reason Why you have blocked this person.

priority Gives the entry a priority. 0 is standard. If the priority is higher the value will be also blocked/accepted if a higher priority is choosen.

listflag A flag, which indicates if you would use the blacklist (BLACKLIST_FLAG) or the whitelist (WHITELIST_FLAG).

The value "PRIO_ALL" stands for all priorities.

Returns:

If all goes well 0 (zero) otherwise -1 and the errno variable will be set..

Definition at line 74 of file pf_administration.c.

References DB_FILE, ERR_LOG, MAX_LINE_LENGTH, PRIO_ALL, STMT_SIZE, TB_AREACODE, TB_COUNTRYCODE, TB_NAME, TB_NUMBER, TB_PRIORITY, TB_REASON, and WHITELIST_-FLAG.

4.5.1.2 int change_name (int *country_code*, int *area_code*, unsigned long long *number*, char ∗ *new_name*, int *listflag*)

Changes the name of the entry. For the unique identification you need to enter the country code, area code and the number because this tripple identifies the entry.

Parameters:

country_code The country code (for example 39 for Italy, 43 for Austria, and so one)

area_code The area code which indicates your mobile operator.

number The telephone number of the person (without country and area code.

new_name The new name

listflag A flag, which indicates if you would use the blacklist (BLACKLIST_FLAG) or the whitelist (WHITELIST_FLAG).

Returns:

If the number was changed 1, otherwise 0.

Definition at line 227 of file pf_administration.c.

References DB_FILE, ERR_LOG, MAX_LINE_LENGTH, STMT_SIZE, TB_AREACODE, TB_-COUNTRYCODE, TB_NAME, TB_NUMBER, and WHITELIST_FLAG.

4.5.1.3 int change_number (int *country_code*, int *area_code*, unsigned long long *number*, int *new_country_code*, int *new_area_code*, unsigned long long *new_number*, int *listflag*)

Changes the number of the entry. You need to enter the country code, area code and the number because this tripple identifies the entry.

Parameters:

country_code The country code (for example 39 for Italy, 43 for Austria, and so one)

area_code The area code which indicates your mobile operator.

number The telephone number of the person (without country and area code.

new_country_code The new country code.

new_area_code The new area code.

new_number The new number.

listflag A flag, which indicates if you would use the blacklist (BLACKLIST_FLAG) or the whitelist (WHITELIST_FLAG).

Returns:

If the number was changed 1, otherwise 0.

Definition at line 266 of file pf_administration.c.

References DB_FILE, ERR_LOG, MAX_LINE_LENGTH, STMT_SIZE, TB_AREACODE, TB_COUNTRYCODE, TB_NUMBER, and WHITELIST_FLAG.

4.5.1.4 int change_priority (int *country_code*, int *area_code*, unsigned long long *number*, int *new_priority*, int *listflag*)

Changes the priority of the entry. For the unique identification you need to enter the country code, area code and the number because this tripple identifies the entry.

Parameters:

country_code The country code (for example 39 for Italy, 43 for Austria, and so one)

area_code The area code which indicates your mobile operator.

number The telephone number of the person (without country and area code.

new_priority The new priority.

listflag A flag, which indicates if you would use the blacklist (BLACKLIST_FLAG) or the whitelist (WHITELIST_FLAG).

Returns:

If the number was changed 1, otherwise 0.

Definition at line 346 of file pf_administration.c.

References DB_FILE, ERR_LOG, MAX_LINE_LENGTH, STMT_SIZE, TB_AREACODE, TB_COUNTRYCODE, TB_NUMBER, TB_PRIORITY, and WHITELIST_FLAG.

4.5.1.5 int change_reason (int *country_code*, int *area_code*, unsigned long long *number*, char * *new_reason*, int *listflag*)

Changes the reason of the entry. For the unique identification you need to enter the country code, area code and the number because this tripple identifies the entry.

Parameters:

country_code The country code (for example 39 for Italy, 43 for Austria, and so one)

area_code The area code which indicates your mobile operator.

number The telephone number of the person (without country and area code.

new_reason The new reason.

listflag A flag, which indicates if you would use the blacklist (BLACKLIST_FLAG) or the whitelist (WHITELIST_FLAG).

Returns:

If the number was changed 1, otherwise 0.

Definition at line 307 of file pf_administration.c.

References DB_FILE, ERR_LOG, MAX_LINE_LENGTH, STMT_SIZE, TB_AREACODE, TB_COUNTRYCODE, TB_NUMBER, TB_REASON, and WHITELIST_FLAG.

4.5.1.6 int check_entry (int *country_code*, int *area_code*, unsigned long long *number*, int *priority*, int *listflag*)

Checks if a number is on the blacklist/whitelist.

Parameters:

country_code The country code (for example 39 for Italy, 43 for Austria, and so one)

area_code The area code which indicates your mobile operator.

number The telephone number of the person (without country and area code.

priority Gives the entry a priority. 0 is standard. If the priority is higher the value will be also blocked/accepted if a higher priority is choosen.

listflag A flag, which indicates if you would use the blacklist (BLACKLIST_FLAG) or the whitelist (WHITELIST_FLAG).

The value "PRIO_ALL" stands for all priorities.

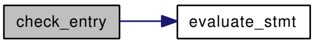
Returns:

If the number was found 1, otherwise 0.

Definition at line 155 of file pf_administration.c.

References Entry::area_code, BLACKLIST_FLAG, Entry::country_code, DB_FILE, ERR_LOG, evaluate_stmt(), INFO_LOG, MAX_LINE_LENGTH, Entry::number, Entry::priority, STMT_SIZE, TB_AREACODE, TB_COUNTRYCODE, TB_NUMBER, TB_PRIORITY, and WHITELIST_FLAG.

Here is the call graph for this function:



4.5.1.7 int rm_entry (int *country_code*, int *area_code*, unsigned long long *number*, int *listlfag*)

Removes a number from the blacklist/whitelist.

Parameters:

country_code The country code (for example 39 for Italy, 43 for Austria, and so one)

area_code The area code which indicates your mobile operator.

number The number which will be deleted.

listflag A flag, which indicates if you would use the blacklist (BLACKLIST_FLAG) or the whitelist (WHITELIST_FLAG).

Returns:

If all goes right 0, otherwise an error code.

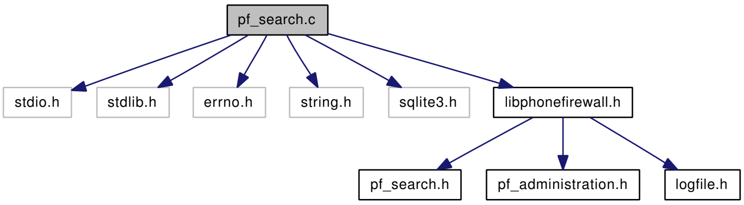
Definition at line 119 of file pf_administration.c.

References DB_FILE, ERR_LOG, MAX_LINE_LENGTH, STMT_SIZE, TB_AREACODE, TB_COUNTRYCODE, TB_NUMBER, and WHITELIST_FLAG.

4.6 pf_search.c File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <errno.h>
#include <string.h>
#include <sqlite3.h>
#include "libphonefirewall.h"
```

Include dependency graph for pf_search.c:



Defines

- #define ASCII_PERCENT_CHAR 37

Functions

- struct Entry ∗ insert_into_list (struct Entry ∗p_root, struct Entry ∗p_entry)
- struct Entry ∗ find_entry (sqlite3_stmt ∗pp_stmt)
- struct Entry ∗ get_entry_by_name (char ∗name, int listflag)
- struct Entry ∗ get_entry_by_number (int country_code, int area_code, unsigned long long number, int listflag)
- struct Entry ∗ get_entry_by_reason (char ∗reason, int listflag)

4.6.1 Define Documentation

4.6.1.1 #define ASCII_PERCENT_CHAR 37

Definition at line 27 of file pf_search.c.

Referenced by get_entry_by_name(), get_entry_by_number(), and get_entry_by_reason().

4.6.2 Function Documentation

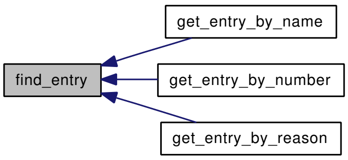
4.6.2.1 struct Entry∗ find_entry (sqlite3_stmt ∗ *pp_stmt*) `[read]`

Definition at line 76 of file pf_search.c.

References Entry::area_code, Entry::country_code, Entry::name, Entry::number, Entry::priority, Entry::reason, TB_AREACODE, TB_COUNTRYCODE, TB_NAME, TB_NUMBER, TB_PRIORITY, and TB_REASON.

Referenced by get_entry_by_name(), get_entry_by_number(), and get_entry_by_reason().

Here is the caller graph for this function:



4.6.2.2 struct Entry∗ get_entry_by_name (char ∗ *name*, int *listflag*) [read]

Search a entry by name.

Parameters:

name The name of the person which is blocked.

listflag A flag, which indicates if you would use the blacklist (BLACKLIST_FLAG) or the whitelist (WHITELIST_FLAG).

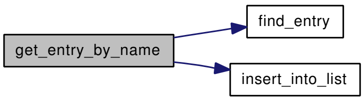
Returns:

entry Returns the entries which are found in a linked list.

Definition at line 112 of file pf_search.c.

References ASCII_PERCENT_CHAR, DB_FILE, ERR_LOG, find_entry(), insert_into_list(), MAX_-LINE_LENGTH, STMT_SIZE, TB_AREACODE, TB_COUNTRYCODE, TB_NAME, TB_NUMBER, TB_PRIORITY, TB_REASON, and WHITELIST_FLAG.

Here is the call graph for this function:



4.6.2.3 struct Entry∗ get_entry_by_number (int *country_code*, int *area_code*, unsigned long long *number*, int *listflag*) [read]

Search a entries by number (country code + area code + number).

Parameters:

country_code The country code (for example 39 for Italy, 43 for Austria, and so one)

area_code The area code which indicates your mobile operator.

number The telephone number of the person (without country and area code.

listflag A flag, which indicates if you would use the blacklist (BLACKLIST_FLAG) or the whitelist (WHITELIST_FLAG).

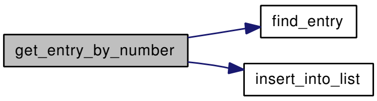
Returns:

entry Returns the entries which are found in a linked list.

Definition at line 158 of file pf_search.c.

References ASCII_PERCENT_CHAR, DB_FILE, ERR_LOG, find_entry(), insert_into_list(), MAX_-LINE_LENGTH, STMT_SIZE, TB_AREACODE, TB_COUNTRYCODE, TB_NAME, TB_NUMBER, TB_PRIORITY, TB_REASON, and WHITELIST_FLAG.

Here is the call graph for this function:



4.6.2.4 struct Entry∗ get_entry_by_reason (char ∗ *reason*, int *listflag*) [read]

Search a entries by reason.

Parameters:

reason The reason why a person is blocked/accpeted.

listflag A flag, which indicates if you would use the blacklist (BLACKLIST_FLAG) or the whitelist (WHITELIST_FLAG).

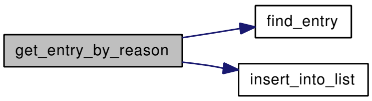
Returns:

entry Returns the entries which are found in a linked list.

Definition at line 206 of file pf_search.c.

References ASCII_PERCENT_CHAR, DB_FILE, ERR_LOG, find_entry(), insert_into_list(), MAX_-LINE_LENGTH, STMT_SIZE, TB_AREACODE, TB_COUNTRYCODE, TB_NAME, TB_NUMBER, TB_PRIORITY, TB_REASON, and WHITELIST_FLAG.

Here is the call graph for this function:



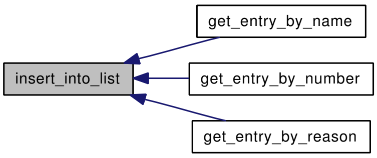
4.6.2.5 struct Entry∗ insert_into_list (struct Entry ∗ *p_root*, struct Entry ∗ *p_entry*) [read]

Definition at line 29 of file pf_search.c.

References Entry::area_code, Entry::country_code, Entry::name, Entry::next, Entry::number, and Entry::reason.

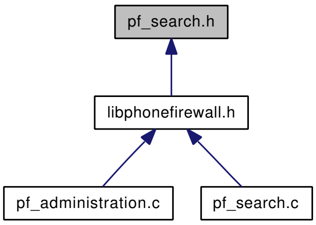
Referenced by get_entry_by_name(), get_entry_by_number(), and get_entry_by_reason().

Here is the caller graph for this function:



4.7 pf_search.h File Reference

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



Functions

- struct Entry ∗ get_entry_by_name (char ∗name, int listflag)
- struct Entry ∗ get_entry_by_number (int country_code, int area_code, unsigned long long number, int listflag)
- struct Entry ∗ get_entry_by_reason (char ∗reason, int listflag)

4.7.1 Function Documentation

4.7.1.1 struct Entry∗ get_entry_by_name (char ∗ *name*, int *listflag*) [read]

Search a entry by name.

Parameters:

name The name of the person which is blocked.

listflag A flag, which indicates if you would use the blacklist (BLACKLIST_FLAG) or the whitelist (WHITELIST_FLAG).

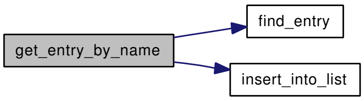
Returns:

entry Returns the entries which are found in a linked list.

Definition at line 112 of file pf_search.c.

References ASCII_PERCENT_CHAR, DB_FILE, ERR_LOG, find_entry(), insert_into_list(), MAX_-LINE_LENGTH, STMT_SIZE, TB_AREACODE, TB_COUNTRYCODE, TB_NAME, TB_NUMBER, TB_PRIORITY, TB_REASON, and WHITELIST_FLAG.

Here is the call graph for this function:



4.7.1.2 struct Entry∗ get_entry_by_number (int *country_code*, int *area_code*, unsigned long long *number*, int *listflag*) [read]

Search a entries by number (country code + area code + number).

Parameters:

country_code The country code (for example 39 for Italy, 43 for Austria, and so one)

area_code The area code which indicates your mobile operator.

number The telephone number of the person (without country and area code.

listflag A flag, which indicates if you would use the blacklist (BLACKLIST_FLAG) or the whitelist (WHITELIST_FLAG).

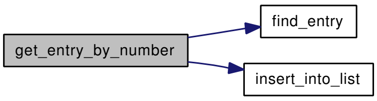
Returns:

entry Returns the entries which are found in a linked list.

Definition at line 158 of file pf_search.c.

References ASCII_PERCENT_CHAR, DB_FILE, ERR_LOG, find_entry(), insert_into_list(), MAX_-LINE_LENGTH, STMT_SIZE, TB_AREACODE, TB_COUNTRYCODE, TB_NAME, TB_NUMBER, TB_PRIORITY, TB_REASON, and WHITELIST_FLAG.

Here is the call graph for this function:



4.7.1.3 struct Entry∗ get_entry_by_reason (char ∗ *reason*, int *listflag*) [read]

Search a entries by reason.

Parameters:

reason The reason why a person is blocked/accpeted.

listflag A flag, which indicates if you would use the blacklist (BLACKLIST_FLAG) or the whitelist (WHITELIST_FLAG).

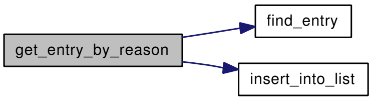
Returns:

entry Returns the entries which are found in a linked list.

Definition at line 206 of file pf_search.c.

References ASCII_PERCENT_CHAR, DB_FILE, ERR_LOG, find_entry(), insert_into_list(), MAX_-LINE_LENGTH, STMT_SIZE, TB_AREACODE, TB_COUNTRYCODE, TB_NAME, TB_NUMBER, TB_PRIORITY, TB_REASON, and WHITELIST_FLAG.

Here is the call graph for this function:



Index