

My Project

Generated by Doxygen 1.12.0

1 File Index	1
1.1 File List	1
2 File Documentation	3
2.1 change_id.c File Reference	3
2.1.1 Detailed Description	3
2.1.2 Function Documentation	3
2.1.2.1 change_group()	3
2.1.2.2 change_ids()	4
2.1.2.3 get_group_gid()	4
2.1.2.4 set_id_and_groups()	4
2.1.2.5 set_root()	5
2.2 int_to_str.c File Reference	5
2.2.1 Detailed Description	5
2.3 main.c File Reference	5
2.3.1 Detailed Description	6
2.3.2 Function Documentation	6
2.3.2.1 check_authorized()	6
2.3.2.2 check_password()	6
2.3.2.3 main()	7
2.3.2.4 run_command()	7
2.3.2.5 run_flags()	7
2.4 utilities.c File Reference	8
2.4.1 Detailed Description	8
2.4.2 Function Documentation	8
2.4.2.1 disable_echo()	8
2.4.2.2 enable_echo()	9
2.4.2.3 get_hash()	9
2.4.2.4 get_salt()	9
2.4.2.5 get_username()	9
Index	11

Chapter 1

File Index

1.1 File List

Here is a list of all documented files with brief descriptions:

change_id.c	Change all the ids	3
int_to_str.c	Convert an integer into a char *	5
main.c	Handle the main functions of the 'sudo' project	5
utilities.c	Some tool functions to use for hash and username parsing	8

Chapter 2

File Documentation

2.1 change_id.c File Reference

Change all the ids.

```
#include "../include/my_sudo.h"
```

Functions

- void `set_root` (void)
Set the root ids.
- int `set_id_and_groups` (char *uid, char *gid, int group_count, gid_t *groups_id)
Set the right ids.
- gid_t `get_group_gid` (const char *groupname)
Get the gid of a group with a given groupname.
- int `change_group` (char *groupname, char *username)
Set the correct gids.
- int `change_ids` (flags_t flags, char *username)
Main function to change the id.

2.1.1 Detailed Description

Change all the ids.

Necessary for the sudo function to work well

2.1.2 Function Documentation

2.1.2.1 change_group()

```
int change_group (  
    char * groupname,  
    char * username)
```

Set the correct gids.

Parameters

<i>*groupname</i>	The groupname to set the correct gid
<i>*username</i>	The username to get the correct groups name

Returns

int A status code

2.1.2.2 change_ids()

```
int change_ids (  
    flags_t flags,  
    char * username)
```

Main function to change the id.

Parameters

<i>flags</i>	The structure whith infos on the flags
<i>*username</i>	The username to get all the correct informations

Returns

void

2.1.2.3 get_group_gid()

```
gid_t get_group_gid (  
    const char * groupname)
```

Get the gid of a group with a given groupname.

Parameters

<i>*groupname</i>	The given groupname
-------------------	---------------------

Returns

gid_t The structure with all the informations

2.1.2.4 set_id_and_groups()

```
int set_id_and_groups (  
    char * uid,  
    char * gid,  
    int group_count,  
    gid_t * groups_id)
```

Set the right ids.

Parameters

<i>*uid</i>	uid to set
<i>*gid</i>	gid to set
<i>group_count</i>	The number of groups
<i>*groups_id</i>	A structure that contains infos on groups

Returns

int A status code

2.1.2.5 set_root()

```
void set_root (
    void )
```

Set the root ids.

Returns

void

2.2 int_to_str.c File Reference

Convert an integer into a char *.

```
#include "include/my_sudo.h"
```

Functions

- int **my_strlen** (char const *str)
- int **my_len_nbr** (int nb)
- char * **turns_int_to_string** (int nb)

2.2.1 Detailed Description

Convert an integer into a char *.

Contains the function to convert an integer paramter into a char* parameter

2.3 main.c File Reference

Handle the main functions of the 'sudo' project.

```
#include "../include/my_sudo.h"
```

Functions

- int `check_password` (char *hash, char *salt, char *username)
Get password from the stdin and compare its hash.
- int `run_command` (char **argv, int argc, flags_t flags)
Extract and execute the command from the user input.
- int `check_authorized` (char **argv, int argc, char *username, flags_t flags)
Check if the user is in the sudoers file and change its ids.
- int `run_flags` (char **argv, flags_t flags, int argc)
Extract the flags and their parameters to activate in the structure.
- int `main` (int argc, char **argv)
Check if the command has no errors and execute the main code.

2.3.1 Detailed Description

Handle the main functions of the 'sudo' project.

Displaying prompt to get user password and checking permissions in the sudoers file. Hash given password and check it.

2.3.2 Function Documentation

2.3.2.1 `check_authorized()`

```
int check_authorized (
    char ** argv,
    int argc,
    char * username,
    flags_t flags)
```

Check if the user is in the sudoers file and change its ids.

Parameters

<i>**argv</i>	The arguments array
<i>argc</i>	The size of the arguments array
<i>*username</i>	The username to know wich passwd hash to look for
<i>flags</i>	The structure which contains the activated flags and settings

Returns

int A status code to determine the behavior to adopt

2.3.2.2 `check_password()`

```
int check_password (
    char * hash,
    char * salt,
    char * username)
```

Get password from the stdin and compare its hash.

Parameters

<i>*hash</i>	The hash of the user password found in /etc/shadow
<i>*salt</i>	The salt of the user password found in /etc/shadow
<i>*username</i>	The username from who we take the password

Returns

int A status code for the other functions to determine the behavior

2.3.2.3 main()

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

Check if the command has no errors and execute the main code.

Parameters

<i>**argv</i>	The arguments array
<i>argc</i>	The size of the arguments array

Returns

int A status code to determine the behavior to adopt

2.3.2.4 run_command()

```
int run_command (  
    char ** argv,  
    int argc,  
    flags_t flags)
```

Extract and execute the command from the user input.

Parameters

<i>**argv</i>	The arguments array
<i>argc</i>	The size of the argv array
<i>flags</i>	The structure which contains the activated flags and settings

Returns

int A status code to determine the behavior to adopt

2.3.2.5 run_flags()

```
int run_flags (  
    char ** argv,  
    flags_t flags,  
    int argc)
```

Extract the flags and their parameters to activate in the structure.

Parameters

<i>**argv</i>	The arguments array
<i>flags</i>	The structure which contains the activated flags and settings
<i>argc</i>	The size of the arguments array

Returns

int A status code to determine the behavior to adopt

2.4 utilities.c File Reference

Some tool functions to use for hash and username parsing.

```
#include "../include/my_sudo.h"
```

Functions

- char * [get_username](#) (char *uid)
Check if all the conditions are ok to look for the username.
- char * [get_hash](#) (char *username)
Check if the conditions are ok to look for the hash.
- char * [get_salt](#) (char *hash)
Extract the salt from a hash.
- void [disable_echo](#) (void)
Disable the display of entered chars in the stdin.
- void [enable_echo](#) (void)
Enable the display of chars in the stdin.

2.4.1 Detailed Description

Some tool functions to use for hash and username parsing.

Contains the function to parse informations to use in the other file. Also has functions to handle stdin status (enable/disable)

2.4.2 Function Documentation

2.4.2.1 [disable_echo\(\)](#)

```
void disable_echo (
    void )
```

Disable the display of entered chars in the stdin.

Returns

void

2.4.2.2 enable_echo()

```
void enable_echo (
    void )
```

Enable the display of chars in the stdin.

Returns

void

2.4.2.3 get_hash()

```
char * get_hash (
    char * username)
```

Check if the conditions are ok to look for the hash.

Parameters

<i>*username</i>	The given username
------------------	--------------------

Returns

char* The hash

2.4.2.4 get_salt()

```
char * get_salt (
    char * hash)
```

Extract the salt from a hash.

Parameters

<i>*hash</i>	The hash
--------------	----------

Returns

char* The salt

2.4.2.5 get_username()

```
char * get_username (
    char * uid)
```

Check if all the conditions are ok to look for the username.

Parameters

<i>*uid</i>	The given uid
-------------	---------------

Returns

char* The hash

Index

- change_group
 - change_id.c, [3](#)
- change_id.c, [3](#)
 - change_group, [3](#)
 - change_ids, [4](#)
 - get_group_gid, [4](#)
 - set_id_and_groups, [4](#)
 - set_root, [5](#)
- change_ids
 - change_id.c, [4](#)
- check_authorized
 - main.c, [6](#)
- check_password
 - main.c, [6](#)
- disable_echo
 - utilities.c, [8](#)
- enable_echo
 - utilities.c, [8](#)
- get_group_gid
 - change_id.c, [4](#)
- get_hash
 - utilities.c, [9](#)
- get_salt
 - utilities.c, [9](#)
- get_username
 - utilities.c, [9](#)
- int_to_str.c, [5](#)
- main
 - main.c, [7](#)
- main.c, [5](#)
 - check_authorized, [6](#)
 - check_password, [6](#)
 - main, [7](#)
 - run_command, [7](#)
 - run_flags, [7](#)
- run_command
 - main.c, [7](#)
- run_flags
 - main.c, [7](#)
- set_id_and_groups
 - change_id.c, [4](#)
- set_root
 - change_id.c, [5](#)
- utilities.c, [8](#)
 - disable_echo, [8](#)
 - enable_echo, [8](#)
 - get_hash, [9](#)
 - get_salt, [9](#)
 - get_username, [9](#)