

fix-hostfiles

1.20

Generated by Doxygen 1.10.0

1 File Index	1
1.1 File List	1
2 File Documentation	3
2.1 actions.h	3
2.2 src/fix-hosts.h File Reference	3
2.2.1 Detailed Description	4
2.2.2 Function Documentation	4
2.2.2.1 addDnsName()	4
2.2.2.2 dnsFlush()	5
2.2.2.3 updateHostsFiles()	5
2.2.2.4 usage()	5
2.3 fix-hosts.h	6
2.4 src/system-actions.h File Reference	6
2.4.1 Detailed Description	7
2.4.2 Function Documentation	8
2.4.2.1 booleanQuery()	8
2.4.2.2 checkProcess()	8
2.4.2.3 copyFile()	8
2.4.2.4 copyFile2()	9
2.4.2.5 displayProcess()	9
2.4.2.6 fileExists()	9
2.4.2.7 fileInfo()	10
2.4.2.8 handleError()	10
2.4.2.9 lsFiles()	10
2.4.2.10 validateDNSname()	11
2.5 system-actions.h	11
Index	13

Chapter 1

File Index

1.1 File List

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

src/ actions.h	3
src/ fix-hosts.h	
Modify /etc/hosts for use with hblock(1)	3
src/ system-actions.h	
Common functions and system actions	6

Chapter 2

File Documentation

2.1 actions.h

```
00001 #ifndef ACTIONS_H
00002 #define ACTIONS_H
00003
00008 typedef enum {
00009     ACTION_PREP,
00010     ACTION_RESTORE,
00011     ACTION_ADD,
00012     ACTION_FLUSH,
00013     ACTION_INVALID
00014 } Action;
00015
00016 #endif /* ACTIONS_H */
```

2.2 src/fix-hosts.h File Reference

Modify /etc/hosts for use with hblock(1)

```
#include "system-actions.h"
#include "actions.h"
#include <dirent.h>
#include <fcntl.h>
#include <fnmatch.h>
#include <grp.h>
#include <libgen.h>
#include <limits.h>
#include <pwd.h>
#include <stdio.h>
#include <stdbool.h>
#include <stdlib.h>
#include <string.h>
#include <sys/stat.h>
#include <sys/types.h>
#include <sys/utsname.h>
#include <time.h>
#include <unistd.h>
```

Macros

- `#define ETC "/etc/"`
- `#define HOSTS ETC "hosts"`
- `#define HOSTS_ORIG ETC "hosts-ORIG"`
- `#define HOSTFILES "hosts*"`
- `#define HBLOCK "hblock"`
- `#define ALLOWLIST ETC HBLOCK "/allow.list"`
- `#define HBLOCK_DIR ETC HBLOCK`

Functions

- void `usage` (const char *program)
Display usage.
- int `updateHostsFiles` (const char *src, const char *dst, Action action)
Modify /etc/hosts.
- int `addDnsName` (const char *hblock_dir, const char *dns_name, const char *allow_file)
Add DNS name to hblock allow list.
- int `dnsFlush` (void)
Flush the DNS cache and restart mDNSResponder daemon.

2.2.1 Detailed Description

Modify /etc/hosts for use with hblock(1)

Author

Robert Primmer (<https://github.com/rprimer>)

hblock(1) is a shell script, available on [homebrew](#), that blocks ads, beacons and malware sites. It does this by editing /etc/hosts and setting the IP address for such sites to 0.0.0.0. The issue is that hblock sometimes adds sites to /etc/hosts that are needed. This executable fixes such issues by adding good DNS hosts to the exclusion list (/etc/hblock/allow.list) and removing the corresponding entry from /etc/hosts. It will also optionally flush the DNS cache and restart the mDNSResponder daemon.

Version

1.1

Date

2024-03-22

2.2.2 Function Documentation

2.2.2.1 addDnsName()

```
int addDnsName (
    const char * hblock_dir,
    const char * dns_name,
    const char * allow_file )
```

Add DNS name to hblock allow list.

Parameters

<i>hblock_dir</i>	Directory holding hblock allow list
<i>dns_name</i>	DNS name to add
<i>allow_file</i>	Name of allow list file

Returns

int Return status

2.2.2.2 dnsFlush()

```
int dnsFlush (
    void )
```

Flush the DNS cache and restart mDNSResponder daemon.

Returns

int Return status

2.2.2.3 updateHostsFiles()

```
int updateHostsFiles (
    const char * src,
    const char * dst,
    Action action )
```

Modify /etc/hosts.

Parameters

<i>src</i>	Source file for action, typically hosts or hosts-ORIG
<i>dst</i>	Destination for action, typically hosts or hosts-ORIG
<i>action</i>	Action being performed. ACTION_PREP is action of interest.

Returns

int Return status

2.2.2.4 usage()

```
void usage (
    const char * program )
```

Display usage.

Parameters

<i>program</i>	Basename of calling program
----------------	-----------------------------

2.3 fix-hosts.h

[Go to the documentation of this file.](#)

```

00001
00017 #ifndef FIX_HOSTS_H
00018 #define FIX_HOSTS_H
00019
00020 #include "system-actions.h"
00021 #include "actions.h"
00022
00023 #include <dirent.h>
00024 #include <fcntl.h>
00025 #include <fnmatch.h>
00026 #include <grp.h>
00027 #include <libgen.h>
00028 #include <limits.h>
00029 #include <pwd.h>
00030 #include <stdio.h>
00031 #include <stdbool.h>
00032 #include <stdlib.h>
00033 #include <string.h>
00034 #include <sys/stat.h>
00035 #include <sys/types.h>
00036 #include <sys/utsname.h>
00037 #include <time.h>
00038 #include <unistd.h>
00039
00040 #ifdef DEBUG
00041 #define ETC "./etc/"
00042 #else
00043 #define ETC "/etc/"
00044 #endif // DEBUG
00045
00046 #define HOSTS ETC "hosts"
00047 #define HOSTS_ORIG ETC "hosts-ORIG"
00048 #define HOSTFILES "hosts*"
00049
00050 #define HBLOCK "hblock"
00051 #define ALLOWLIST ETC HBLOCK "/allow.list"
00052 #define HBLOCK_DIR ETC HBLOCK
00053
00059 void usage(const char *program);
00060
00069 int updateHostsFiles(const char *src, const char *dst, Action action);
00070
00079 int addDnsName(const char *hblock_dir, const char *dns_name, const char *allow_file);
00080
00086 int dnsFlush(void);
00087
00088 #endif // FIX_HOSTS_H

```

2.4 src/system-actions.h File Reference

Common functions and system actions.

```

#include <dirent.h>
#include <errno.h>
#include <fcntl.h>
#include <fnmatch.h>
#include <grp.h>
#include <libgen.h>
#include <limits.h>
#include <pwd.h>

```

```
#include <regex.h>
#include <stdarg.h>
#include <stdbool.h>
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <sys/stat.h>
#include <sys/types.h>
#include <sys/wait.h>
#include <time.h>
#include <unistd.h>
```

Macros

- #define **HANDLE_ERROR**(fmt, ...) handleError(true, __FILE__, __func__, __LINE__, fmt, ##__VA_ARGS__↵
__)
- #define **REPORT_ERROR**(fmt, ...) handleError(false, __FILE__, __func__, __LINE__, fmt, ##__VA_ARGS__↵
__)

Functions

- void **handleError** (bool fatal, char *file, const char *func, int line, const char *fmt,...)
Common error handling routine.
- int **booleanQuery** (const char *prompt)
Query user for yes or no.
- int **fileExists** (const char *filename)
Check for file existence.
- int **copyFile** (const char *src, const char *dest)
Make a copy of a file. Uses libC functions.
- int **copyFile2** (const char *src, const char *dest)
Make a copy of a file. Uses system calls.
- int **lsFiles** (const char *dirname, const char *files)
List files in a directory.
- int **fileInfo** (const char *filepath)
Display information about a file.
- int **checkProcess** (const char *process_name, bool display_pids)
Check if a process is currently running.
- int **displayProcess** (const char *process_name)
Display info on a running process.
- int **validateDNSname** (const char *dns_name)
DNS name must start & end with a letter or a number and can only contain letters, numbers, and hyphens.

2.4.1 Detailed Description

Common functions and system actions.

Author

Robert Primmer (<https://github.com/rprimer>)

Version

1.3

Date

2024-04-02

2.4.2 Function Documentation

2.4.2.1 booleanQuery()

```
int booleanQuery (
    const char * prompt )
```

Query user for yes or no.

Parameters

<i>prompt</i>	Message to be displayed to user.
---------------	----------------------------------

Returns

int Return true if user entered y or Y.

2.4.2.2 checkProcess()

```
int checkProcess (
    const char * process_name,
    bool display_pids )
```

Check if a process is currently running.

Parameters

<i>process_name</i>	Process to look for.
<i>display_pids</i>	Optionally display PIDs of running process.

Returns

int Return status.

2.4.2.3 copyFile()

```
int copyFile (
    const char * src,
    const char * dest )
```

Make a copy of a file. Uses libC functions.

Parameters

<i>src</i>	File to be copied.
<i>dest</i>	Filename of copy.

Returns

int Return status.

2.4.2.4 copyFile2()

```
int copyFile2 (
    const char * src,
    const char * dest )
```

Make a copy of a file. Uses system calls.

Parameters

<i>src</i>	File to be copied.
<i>dest</i>	Filename of copy.

Returns

int Return status.

2.4.2.5 displayProcess()

```
int displayProcess (
    const char * process_name )
```

Display info on a running process.

Parameters

<i>process_name</i>	Process to look for.
---------------------	----------------------

Returns

int Return status.

2.4.2.6 fileExists()

```
int fileExists (
    const char * filename )
```

Check for file existence.

Parameters

<i>filename</i>	File to check.
-----------------	----------------

Returns

int Return true if file exists.

2.4.2.7 fileInfo()

```
int fileInfo (
    const char * filepath )
```

Display information about a file.

Parameters

<i>filepath</i>	File to stat(2).
-----------------	------------------

Returns

int Return status.

2.4.2.8 handleError()

```
void handleError (
    bool fatal,
    char * file,
    const char * func,
    int line,
    const char * fmt,
    ... )
```

Common error handling routine.

Parameters

<i>fatal</i>	If true, exit program, else returns to the caller.
<i>file</i>	C filename (translation unit) of caller.
<i>func</i>	Function name of caller.
<i>line</i>	Line number in translation unit.
<i>fmt</i>	Optional parameters can be provided (va_list).

2.4.2.9 lsFiles()

```
int lsFiles (
```

```
const char * dirname,
const char * files )
```

List files in a directory.

Parameters

<i>dirname</i>	Directory housing files.
<i>files</i>	Files to list.

Returns

int Return status.

2.4.2.10 validateDNSName()

```
int validateDNSName (
    const char * dns_name )
```

DNS name must start & end with a letter or a number and can only contain letters, numbers, and hyphens.

Parameters

<i>dns_name</i>	DNS name to check.
-----------------	--------------------

Returns

int Return status.

2.5 system-actions.h

[Go to the documentation of this file.](#)

```
00001
00009 #ifndef SYSTEM_ACTIONS_H
00010 #define SYSTEM_ACTIONS_H
00011
00012 #include <dirent.h>
00013 #include <errno.h>
00014 #include <fcntl.h>
00015 #include <fnmatch.h>
00016 #include <grp.h>
00017 #include <libgen.h>
00018 #include <limits.h>
00019 #include <pwd.h>
00020 #include <regex.h>
00021 #include <stdarg.h>
00022 #include <stdbool.h>
00023 #include <stdio.h>
00024 #include <stdlib.h>
00025 #include <string.h>
00026 #include <sys/stat.h>
00027 #include <sys/types.h>
00028 #include <sys/wait.h>
00029 #include <time.h>
00030 #include <unistd.h>
00031
00032 // #__VA_ARGS__ is a GNU extension that still works if __VA_ARGS__ is empty,
00033 // which supports calling the macro with just a string or with additional format arguments.
00034 // Modern compilers support this so I didn't want to clutter the code with a bunch of
```

```
00035 // #ifdef __GNUC__ conditionals just for the sake of some ancient compiler from long long ago.
00036 // __func__ was introduced in C99.
00037 #define HANDLE_ERROR(fmt, ...) handleError(true, __FILE__, __func__, __LINE__, fmt, ##__VA_ARGS__)
00038 #define REPORT_ERROR(fmt, ...) handleError(false, __FILE__, __func__, __LINE__, fmt, ##__VA_ARGS__)
00039
00049 void handleError(bool fatal, char *file, const char *func, int line, const char *fmt, ...);
00050
00057 int booleanQuery(const char *prompt);
00058
00065 int fileExists(const char *filename);
00066
00074 int copyFile(const char *src, const char *dest);
00075
00083 int copyFile2(const char *src, const char *dest);
00084
00092 int lsFiles(const char *dirname, const char *files);
00093
00100 int fileInfo(const char *filepath);
00101
00109 int checkProcess(const char *process_name, bool display_pids);
00110
00117 int displayProcess(const char *process_name);
00118
00125 int validateDNSname(const char *dns_name);
00126
00127 #endif /* SYSTEM_ACTIONS_H */
```


Index

- addDnsName
 - [fix-hosts.h, 4](#)
- booleanQuery
 - [system-actions.h, 8](#)
- checkProcess
 - [system-actions.h, 8](#)
- copyFile
 - [system-actions.h, 8](#)
- copyFile2
 - [system-actions.h, 9](#)
- displayProcess
 - [system-actions.h, 9](#)
- dnsFlush
 - [fix-hosts.h, 5](#)
- fileExists
 - [system-actions.h, 9](#)
- fileInfo
 - [system-actions.h, 10](#)
- fix-hosts.h
 - [addDnsName, 4](#)
 - [dnsFlush, 5](#)
 - [updateHostsFiles, 5](#)
 - [usage, 5](#)
- handleError
 - [system-actions.h, 10](#)
- IsFiles
 - [system-actions.h, 10](#)
- src/actions.h, [3](#)
- src/fix-hosts.h, [3, 6](#)
- src/system-actions.h, [6, 11](#)
- system-actions.h
 - [booleanQuery, 8](#)
 - [checkProcess, 8](#)
 - [copyFile, 8](#)
 - [copyFile2, 9](#)
 - [displayProcess, 9](#)
 - [fileExists, 9](#)
 - [fileInfo, 10](#)
 - [handleError, 10](#)
 - [IsFiles, 10](#)
 - [validateDnsName, 11](#)
- updateHostsFiles
 - [fix-hosts.h, 5](#)
- usage
 - [fix-hosts.h, 5](#)
- validateDnsName
 - [system-actions.h, 11](#)