

TP1

Generated by Doxygen 1.8.11



# Contents

<b>1</b>	<b>Data Structure Index</b>	<b>1</b>
1.1	Data Structures . . . . .	1
<b>2</b>	<b>File Index</b>	<b>3</b>
2.1	File List . . . . .	3
<b>3</b>	<b>Data Structure Documentation</b>	<b>5</b>
3.1	Login Struct Reference . . . . .	5
3.1.1	Field Documentation . . . . .	5
3.1.1.1	name . . . . .	5
3.1.1.2	password . . . . .	5
3.2	sockbuff Struct Reference . . . . .	5
3.2.1	Field Documentation . . . . .	5
3.2.1.1	buffer . . . . .	5
3.2.1.2	buffer_size . . . . .	5
3.2.1.3	sockfd . . . . .	5

<b>4 File Documentation</b>	<b>7</b>
4.1 /home/torce/Desktop/JuanSoriano/TP1/src/baash/archfunc.c File Reference	7
4.1.1 Function Documentation	8
4.1.1.1 aArchivo(char *archivo)	8
4.1.1.2 buscarArchivo(char *file, char **carpetas, char *commandFile)	8
4.1.1.3 deArchivo(char filename[])	8
4.2 /home/torce/Desktop/JuanSoriano/TP1/src/baash/archfunc.h File Reference	8
4.2.1 Function Documentation	9
4.2.1.1 aArchivo(char *archivo)	9
4.2.1.2 buscarArchivo(char *file, char **carpetas, char *commandFile)	9
4.2.1.3 deArchivo(char filename[])	9
4.3 /home/torce/Desktop/JuanSoriano/TP1/src/baash/evalfunc.c File Reference	9
4.3.1 Function Documentation	9
4.3.1.1 evaluarBackground(char **argv)	9
4.3.1.2 evaluarPipe(char **argv, char **argv1, char **argv2)	9
4.3.1.3 evaluarRedir(char **argv, char *filename)	10
4.4 /home/torce/Desktop/JuanSoriano/TP1/src/baash/evalfunc.h File Reference	10
4.4.1 Function Documentation	11
4.4.1.1 evaluarBackground(char **argv)	11
4.4.1.2 evaluarPipe(char **argv, char **argv1, char **argv2)	11
4.4.1.3 evaluarRedir(char **argv, char *filename)	11
4.5 /home/torce/Desktop/JuanSoriano/TP1/src/baash/main.c File Reference	11
4.5.1 Function Documentation	11
4.5.1.1 ejecPipeline(char *argv1[], char *argv2[], char *folders[])	11
4.5.1.2 leerTeclado(char **argv, char *input)	11
4.5.1.3 main()	11
4.5.1.4 parsearPath(char **folders)	12
4.6 /home/torce/Desktop/JuanSoriano/TP1/src/client/main.c File Reference	12
4.6.1 Macro Definition Documentation	13
4.6.1.1 BUFF_SIZE	13

4.6.2	Function Documentation . . . . .	13
4.6.2.1	main(void) . . . . .	13
4.6.2.2	UDPcom(uint16_t UDPport, char *ipaddr, char *filename) . . . . .	13
4.6.2.3	writeto_socket(void *arg) . . . . .	14
4.6.3	Variable Documentation . . . . .	14
4.6.3.1	ipaddr . . . . .	14
4.6.3.2	ipaddrbuff . . . . .	14
4.6.3.3	portnr . . . . .	14
4.7	/home/torce/Desktop/JuanSoriano/TP1/src/server/main.c File Reference . . . . .	14
4.7.1	Macro Definition Documentation . . . . .	15
4.7.1.1	BUFF_SIZE . . . . .	15
4.7.1.2	LISTEN_PORT . . . . .	15
4.7.1.3	SNAME . . . . .	15
4.7.1.4	USER_NR . . . . .	15
4.7.2	Function Documentation . . . . .	15
4.7.2.1	main(void) . . . . .	15
4.7.2.2	UDPcom(uint16_t UDPport) . . . . .	15
4.7.2.3	verificar(char *user, char *password) . . . . .	15
	<b>Index</b>	<b>17</b>



# Chapter 1

## Data Structure Index

### 1.1 Data Structures

Here are the data structures with brief descriptions:

<b>Login</b>	5
<b>sockbuff</b>	5





## Chapter 2

# File Index

### 2.1 File List

Here is a list of all files with brief descriptions:

/home/torce/Desktop/JuanSoriano/TP1/src/baash/ <b>archfunc.c</b> . . . . .	7
/home/torce/Desktop/JuanSoriano/TP1/src/baash/ <b>archfunc.h</b> . . . . .	8
/home/torce/Desktop/JuanSoriano/TP1/src/baash/ <b>evalfunc.c</b> . . . . .	9
/home/torce/Desktop/JuanSoriano/TP1/src/baash/ <b>evalfunc.h</b> . . . . .	10
/home/torce/Desktop/JuanSoriano/TP1/src/baash/ <b>main.c</b> . . . . .	11
/home/torce/Desktop/JuanSoriano/TP1/src/client/ <b>main.c</b> . . . . .	12
/home/torce/Desktop/JuanSoriano/TP1/src/server/ <b>main.c</b> . . . . .	14



## Chapter 3

# Data Structure Documentation

### 3.1 Login Struct Reference

#### Data Fields

- char **name** [SNAME]
- char **password** [SNAME]

#### 3.1.1 Field Documentation

3.1.1.1 char name[SNAME]

3.1.1.2 char password[SNAME]

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

- /home/torce/Desktop/JuanSoriano/TP1/src/server/**main.c**

### 3.2 sockbuff Struct Reference

#### Data Fields

- int **sockfd**
- char \* **buffer**
- size\_t **buffer\_size**

#### 3.2.1 Field Documentation

3.2.1.1 char\* buffer

3.2.1.2 size\_t buffer\_size

3.2.1.3 int sockfd

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

- /home/torce/Desktop/JuanSoriano/TP1/src/client/**main.c**



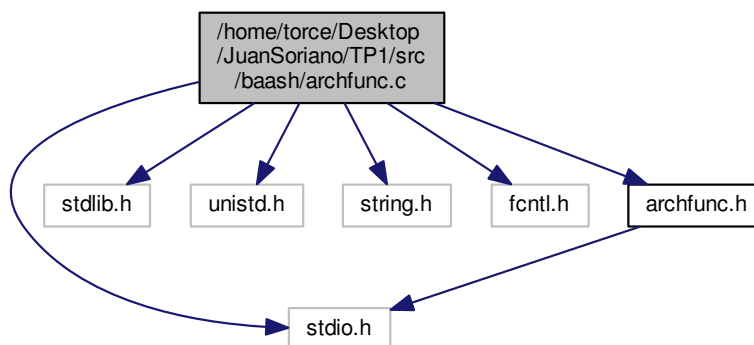
## Chapter 4

# File Documentation

### 4.1 /home/torce/Desktop/JuanSoriano/TP1/src/baash/archfunc.c File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <string.h>
#include <fcntl.h>
#include "archfunc.h"
```

Include dependency graph for archfunc.c:



### Functions

- void **buscarArchivo** (char \*file, char \*\*carpetas, char \*commandFile)
- void **aArchivo** (char \*archivo)
- void **deArchivo** (char filename[])

### 4.1.1 Function Documentation

4.1.1.1 void **aArchivo** ( char \* *archivo* )

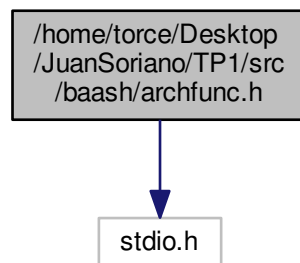
4.1.1.2 void **buscarArchivo** ( char \* *file*, char \*\* *carpetas*, char \* *commandFile* )

4.1.1.3 void **deArchivo** ( char *filename*[ ] )

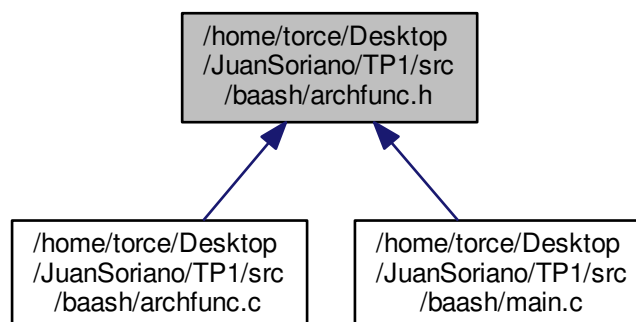
## 4.2 /home/torce/Desktop/JuanSoriano/TP1/src/baash/archfunc.h File Reference

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

Include dependency graph for archfunc.h:



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



### Functions

- void **buscarArchivo** (char \**file*, char \*\**carpetas*, char \**commandFile*)
- void **aArchivo** (char \**archivo*)
- void **deArchivo** (char *filename*[ ])

### 4.2.1 Function Documentation

4.2.1.1 void aArchivo ( char \* *archivo* )

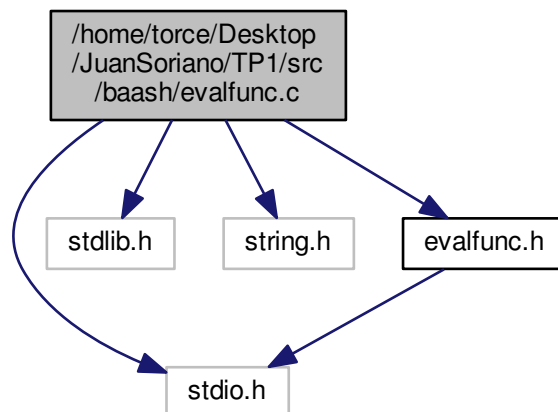
4.2.1.2 void buscarArchivo ( char \* *file*, char \*\* *carpetas*, char \* *commandFile* )

4.2.1.3 void deArchivo ( char *filename*[ ] )

## 4.3 /home/torce/Desktop/JuanSoriano/TP1/src/baash/evalfunc.c File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include "evalfunc.h"
```

Include dependency graph for evalfunc.c:



### Functions

- int **evaluarPipe** (char \*\*argv, char \*\*argv1, char \*\*argv2)
- int **evaluarBackground** (char \*\*argv)
- int **evaluarRedir** (char \*\*argv, char \*filename)

### 4.3.1 Function Documentation

4.3.1.1 int evaluarBackground ( char \*\* *argv* )

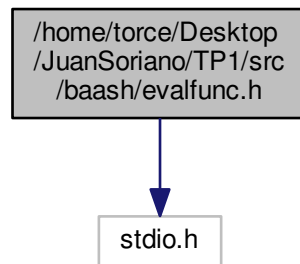
4.3.1.2 int evaluarPipe ( char \*\* *argv*, char \*\* *argv1*, char \*\* *argv2* )

4.3.1.3 `int evaluarRedir ( char ** argv, char * filename )`

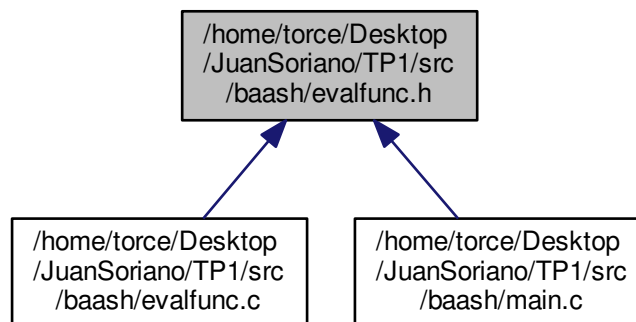
## 4.4 `/home/torce/Desktop/JuanSoriano/TP1/src/baash/evalfunc.h` File Reference

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

Include dependency graph for `evalfunc.h`:



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



## Functions

- `int evaluarPipe (char **argv, char **argv1, char **argv2)`
- `int evaluarBackground (char **argv)`
- `int evaluarRedir (char **argv, char *filename)`



### 4.4.1 Function Documentation

4.4.1.1 `int evaluarBackground ( char ** argv )`

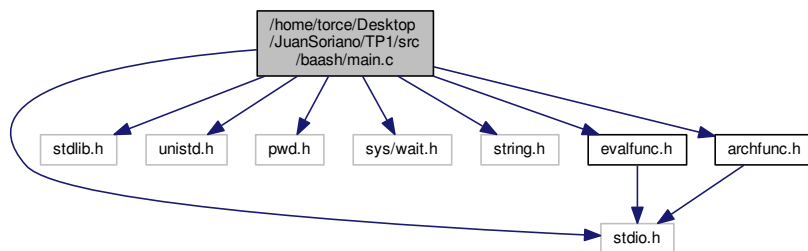
4.4.1.2 `int evaluarPipe ( char ** argv, char ** argv1, char ** argv2 )`

4.4.1.3 `int evaluarRedir ( char ** argv, char * filename )`

## 4.5 /home/torce/Desktop/JuanSoriano/TP1/src/baash/main.c File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <pwd.h>
#include <sys/wait.h>
#include <string.h>
#include "evalfunc.h"
#include "archfunc.h"
```

Include dependency graph for main.c:



## Functions

- void **ejecPipeline** (char \*argv1[], char \*argv2[], char \*folders[])
- int **parsearPath** (char \*\*folders)
- int **leerTeclado** (char \*\*argv, char \*input)
- int **main** ()

### 4.5.1 Function Documentation

4.5.1.1 `void ejecPipeline ( char * argv1[], char * argv2[], char * folders[] )`

4.5.1.2 `int leerTeclado ( char ** argv, char * input )`

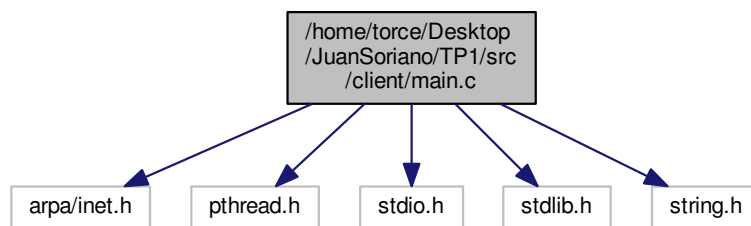
4.5.1.3 `int main ( void )`

#### 4.5.1.4 int parsearPath ( char \*\* folders )

## 4.6 /home/torce/Desktop/JuanSoriano/TP1/src/client/main.c File Reference

```
#include <arpa/inet.h>
#include <pthread.h>
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
```

Include dependency graph for main.c:



### Data Structures

- struct **sockbuff**

### Macros

- #define **BUFF\_SIZE** 1024

### Functions

- void \* **writeto\_socket** (void \*arg)
- int **UDPcom** (uint16\_t UDPport, char \***ipaddr**, char \*filename)
- int **main** (void)

### Variables

- uint16\_t **portnr**
- char **ipaddrbuff** [20]
- char \* **ipaddr** = NULL

## 4.6.1 Macro Definition Documentation

4.6.1.1 `#define BUFF_SIZE 1024`

## 4.6.2 Function Documentation

4.6.2.1 `int main ( void )`

4.6.2.2 `int UDPcom ( uint16_t UDPport, char * ipaddr, char * filename )`

genera una petición de descarga de un archivo al servidor por socket UDP

## Parameters

<i>UDPport</i>	es el puerto que el servidor le asigno al cliente
<i>ipaddr</i>	es la direccion ip
<i>filename</i>	es el nombre del archivo que se quiere descargar

## 4.6.2.3 void \* writeto\_socket ( void \* arg )

toma los datos ingresados por el teclado y parsea lo ingresado para ver si se trata de un comando o una descarga.

## Parameters

<i>arg</i>	es un puntero a la direccion en memoria del struct sockbuff
------------	---

## 4.6.3 Variable Documentation

## 4.6.3.1 char\* ipaddr = NULL

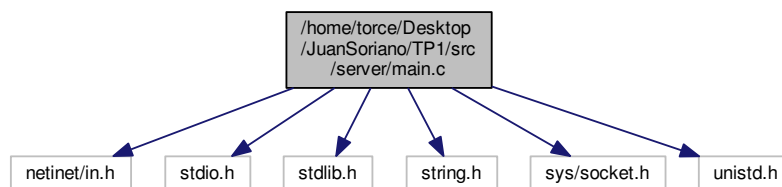
## 4.6.3.2 char ipaddrbuff[20]

## 4.6.3.3 uint16\_t portnr

## 4.7 /home/torce/Desktop/JuanSoriano/TP1/src/server/main.c File Reference

```
#include <netinet/in.h>
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <sys/socket.h>
#include <unistd.h>
```

Include dependency graph for main.c:



## Data Structures

- struct **Login**

## Macros

- `#define LISTEN_PORT 6020`
- `#define SNAME 20`
- `#define USER_NR 2`
- `#define BUFF_SIZE 1024`

## Functions

- `int verificar (char *user, char *password)`
- `void UDPcom (uint16_t UDPport)`
- `int main (void)`

### 4.7.1 Macro Definition Documentation

4.7.1.1 `#define BUFF_SIZE 1024`

4.7.1.2 `#define LISTEN_PORT 6020`

4.7.1.3 `#define SNAME 20`

4.7.1.4 `#define USER_NR 2`

### 4.7.2 Function Documentation

4.7.2.1 `int main ( void )`

4.7.2.2 `void UDPcom ( uint16_t UDPport )`

Realiza una conexion UDP para enviar un archivo de forma no segura al cliente

#### Parameters

<i>UDPport</i>	es el puerto UDP de conexion con el cliente
----------------	---

4.7.2.3 `int verificar ( char * user, char * password )`

Verifica del lado del server que los datos ingresados por el cliente correspondan a un usuario registrado

#### Parameters

<i>user</i>	el nombre de usuario ingresado
<i>password</i>	el password ingresado



# Index

/home/torce/Desktop/JuanSoriano/TP1/src/baash/archfunc.c, 7  
/home/torce/Desktop/JuanSoriano/TP1/src/baash/archfunc.h, 8  
/home/torce/Desktop/JuanSoriano/TP1/src/baash/evalfunc.c, 9  
/home/torce/Desktop/JuanSoriano/TP1/src/baash/evalfunc.h, 10  
/home/torce/Desktop/JuanSoriano/TP1/src/baash/main.c, 11  
/home/torce/Desktop/JuanSoriano/TP1/src/client/main.c, 12  
/home/torce/Desktop/JuanSoriano/TP1/src/server/main.c, 14

aArchivo  
  archfunc.c, 8  
  archfunc.h, 9

archfunc.c  
  aArchivo, 8  
  buscarArchivo, 8  
  deArchivo, 8

archfunc.h  
  aArchivo, 9  
  buscarArchivo, 9  
  deArchivo, 9

BUFF\_SIZE  
  client/main.c, 13  
  server/main.c, 15

baash/main.c  
  ejecPipeline, 11  
  leerTeclado, 11  
  main, 11  
  parsearPath, 11

buffer  
  sockbuff, 5

buffer\_size  
  sockbuff, 5

buscarArchivo  
  archfunc.c, 8  
  archfunc.h, 9

client/main.c  
  BUFF\_SIZE, 13  
  ipaddr, 14  
  ipaddrbuff, 14  
  main, 13  
  portnr, 14  
  UDPcom, 13

writeto\_socket, 14

deArchivo  
  archfunc.c, 8  
  archfunc.h, 9

ejecPipeline  
  baash/main.c, 11

evalfunc.c  
  evaluarBackground, 9  
  evaluarPipe, 9  
  evaluarRedir, 9

evalfunc.h  
  evaluarBackground, 11  
  evaluarPipe, 11  
  evaluarRedir, 11

evaluarBackground  
  evalfunc.c, 9  
  evalfunc.h, 11

evaluarPipe  
  evalfunc.c, 9  
  evalfunc.h, 11

evaluarRedir  
  evalfunc.c, 9  
  evalfunc.h, 11

ipaddr  
  client/main.c, 14

ipaddrbuff  
  client/main.c, 14

LISTEN\_PORT  
  server/main.c, 15

leerTeclado  
  baash/main.c, 11

Login, 5  
  name, 5  
  password, 5

main  
  baash/main.c, 11  
  client/main.c, 13  
  server/main.c, 15

name  
  Login, 5

parsearPath  
  baash/main.c, 11

password  
  Login, 5

- portnr
  - client/main.c, 14
- SNAME
  - server/main.c, 15
- server/main.c
  - BUFF\_SIZE, 15
  - LISTEN\_PORT, 15
  - main, 15
  - SNAME, 15
  - UDPcom, 15
  - USER\_NR, 15
  - verificar, 15
- sockbuff, 5
  - buffer, 5
  - buffer\_size, 5
  - sockfd, 5
- sockfd
  - sockbuff, 5
- UDPcom
  - client/main.c, 13
  - server/main.c, 15
- USER\_NR
  - server/main.c, 15
- verificar
  - server/main.c, 15
- writeto\_socket
  - client/main.c, 14