#include <stdio.h>

#include <dirent.h>

#include<sys/types.h>

#include<unistd.h>

#include<sys/stat.h>

#include<fcntl.h>

#include <stdlib.h>

#include <string.h>

#include <limits.h>

//se define PATH\_MAX a 1024 (máximo numero de bytes en una ruta, incluyendo el caracter de terminacion nulo)

//BORRAR EL FICHERO FIFO

int main( int argc, char \*argv[] )

{

char rutaarchivo[NAME\_MAX], rutaarchivotipo[NAME\_MAX];

int estado;

if(setvbuf(stdout,NULL,\_IONBF,0)) {//NO SE LO QUE ES

perror("\nError en setvbuf");

}

DIR \*carpeta;

struct dirent \*entrada;

struct stat atributos;

if( argc != 2 ) {

printf( "Error en argumentos; debe proporcionar un directorio\n");

exit(EXIT\_FAILURE);

}

if( (carpeta= opendir( argv[1] )) == NULL ) {

perror( "opendir error " );

exit(EXIT\_FAILURE);

}

while( (entrada = readdir( carpeta)) != NULL ){

if(!strcmp(entrada->d\_name,".") ) continue;

if(!strcmp(entrada->d\_name,"..") ) continue;

sprintf(rutaarchivo,"%s/%s",argv[1], entrada->d\_name);

sprintf(rutaarchivotipo,"%s.tipo/%s.tipo",argv[1], entrada->d\_name);//esto pk??

if(stat(rutaarchivo,&atributos) < 0) {

perror("\nError en stat");

printf("\nError en stat sobre el archivo=%s\n", rutaarchivo);

exit(EXIT\_FAILURE);

}

if (fork()==0) {

int fd;

if( (fd=open(rutaarchivotipo,O\_WRONLY|O\_CREAT|O\_TRUNC,0700)) <0)

{ perror("Error en open");

printf("No se ha podido abrir %s\n", rutaarchivotipo);

exit(EXIT\_FAILURE);

}

dup2 (fd,STDOUT\_FILENO);//POR LO QUE veo la salida se va directo al nuevo archivo.tipo

printf( "uid:%i\n", (int) atributos.st\_uid);

printf( "SUID:%s\n",atributos.st\_mode & S\_ISUID?"true":"false");

execlp("file","file", rutaarchivo,NULL);

perror("Error en exec");

printf("No se ha ejecutado file sobre %s\n",rutaarchivo);

exit(EXIT\_FAILURE);

}//fin hijo

}

closedir(carpeta);

while (wait(&estado)!=-1);//nada

exit(EXIT\_SUCCESS);

}