# LINUX系统下C语言库学习

## 1.#include<glob.h>

#include<stdio.h>

#include<string.h>

#include<stdlib.h>

#include <glob.h>

#define SIZE 10

#define COL\_SIZE 128

void find\_patter\_file(char\* pattern, char\*\* file\_name\_table,int\* returnSize, int size,int col\_size)

{

glob\_t rslt\_buff;

int i;

if (pattern == NULL || file\_name\_table == NULL) {

return ;

}

glob(pattern, GLOB\_NOSORT, NULL, &rslt\_buff);

if (rslt\_buff.gl\_pathc < 1)

{

\*returnSize = 0;

return;

}

for (i = 0; (i < size && i < rslt\_buff.gl\_pathc); i++) {

if (strlen(rslt\_buff.gl\_pathv[i]) < col\_size) {

memcpy(file\_name\_table[i], rslt\_buff.gl\_pathv[i], strlen(rslt\_buff.gl\_pathv[i]));

\*returnSize += 1;

}

}

}

int main()

{

char\*\* fileNameTable = NULL;

int returnSize = 0;

char patten[COL\_SIZE] = {0};

int i;

fileNameTable = (char\*\*)malloc(sizeof(char\*) \* SIZE);

for (i = 0; i < 10; i++) {

fileNameTable[i] = (char\*)malloc(sizeof(char) \* COL\_SIZE);

memset(fileNameTable[i], 0 , sizeof(char) \* COL\_SIZE);

}

sprintf(patten, "%s/\*load%d\_0x%x\*", "/usr1/y00456627/testcode",1,0x90005003);

find\_patter\_file(patten, fileNameTable, &returnSize, SIZE, COL\_SIZE);

for (i = 0; i < returnSize; i++) {

printf("%s\n", fileNameTable[i]);

}

}