

Forum:[Hack Forums](#)

Thread: [pipe redirection of child's stdin and stdout doesnt work](#)

mr.cracker - March 21, 2015, 1:05 am

I was trying to redirect child's STDIN and STDOUT to pipes using dup2().
child has : execl("/bin/sh")

I want to give commands from Parent.
Child executes it. Write o/p to stdout, which is redirected to pipe.
Then parent read from pipe, o/p to stdout of parent.

my code is,

```
Code #include<stdio.h>
#include<wait.h>
#include<unistd.h>
#include<stdlib.h>
#include<string.h>
```

```
int main(){
    pid_t pid;
    int n, len;
    int pipe1[2], pipe2[2];
    char buff[1024], data[4096];
```

```
    pipe(pipe1);
    pipe(pipe2);
```

```
    if( (pid = fork()) < 0){
        return 1;
    }else if(pid == 0){ //child
        close(pipe1[1]);    //child closes write end of pipe 1 ( fd1[1]), and
        close(pipe2[0]);    //child closes read end of pipe 2 (fd2[0]).
```

```
        dup2(pipe1[0], STDIN_FILENO); // pipe1[0] read pipe, which is opened in child. And it's
        companion
```

```
        // pipe1[1] write pipe is opened in parent.
```

```
        // whatever write into pipe1[1] will be received by stdin of child.
```

```
        dup2(pipe2[1], STDOUT_FILENO);
        execl("/bin/sh", "/bin/sh", (char *)0);
```

```
        exit(0);
    }
```

```
    //parent
```

```
    printf("parent pid: %d\n", (int)getpid());
    close(pipe1[0]);    //parent closes read end of pipe 1 ( fd1[0])
    close(pipe2[1]);    //parent closes write end of pipe 2 (fd2[1])
```

```

while(1){
printf("\nenter command: ");

//read pathname
if(fgets(buff, 1024, stdin) == NULL){
    printf("error in fgets");
    exit(1);
}

len = strlen(buff); // fgets() guarantees null byte at end
//if(buff[len - 1] == '\n')
// len--; //delete newline from fgets

//write pathname to ipc channel
if(write(pipe1[1], buff, len) != len){
    printf("write error in client\n");
}
printf("command written to child: %s\n", buff);

/* read from IPC, write to standard output */
n = read(pipe2[0], buff, 4096);
printf("am reading\n");
if(write(STDOUT_FILENO, buff, n) != n){
    printf("write error inside while in client\n");
}
}
}

```

it got stuck on here,
Code while((n = read(pipe2[0], buff, 4096)) > 0){I dont get any output.

Why child does not write into pipe2[1] ?
I think thats why system waits for some data in read() function.

Where i got mistake?
How can i make it work?

g4143 - March 21, 2015, 2:00 am

I'm pretty sure you can't redirect a terminal's stdin and stdout that way.

mr.cracker - March 21, 2015, 5:05 am

(Mar 21, 2015 02:00 AM)g4143 Wrote: [\[CITE\]](#) I'm pretty sure you can't redirect a terminal's stdin and stdout that way.

sorry, now its working, the problem was here.

Code if(buff[len - 1] == '\n')

len--; //delete newline from fgets

Since file stream is line buffered in terminal , i need to keep that '\n' to get output.

g4143 - March 21, 2015, 8:56 am

(Mar 21, 2015 05:05 AM)mr.cracker Wrote: [\[CITE\]](#) sorry, now its working, the problem was here.

Code if(buff[len - 1] == '\n')

len--; //delete newline from fgets

Since file stream is line buffered in terminal , i need to keep that '\n' to get output.

Since file stream is line buffered in the shell.

For some reason I thought you were attempting to do this.

Code `execl("/usr/bin/xterm", "/usr/bin/xterm", (char *)0);/*open a terminal as a child*/`

BTW, your program should have a return 0 at the end of main and it should `#include <sys/types.h>`.

mr.cracker - March 21, 2015, 9:11 am

(Mar 21, 2015 08:56 AM)g4143 Wrote: [\[CITE\]](#) For some reason I thought you were attempting to do this.

Code `execl("/usr/bin/xterm", "/usr/bin/xterm", (char *)0);/*open a terminal as a child*/`

i tried above code. It doesn't redirect stdin and stdout.!!!

Can you please tell me the reason ?

I have updated my code with working code above in my question.

now when i get a prompt i entered

"LLL" : that is a wrong command.

then i got,

Code `/bin/sh: 1: LLL: not found`

After that it asks for input continuously. I think the child ask for input. Can you just check my code once more. Just run it. It is a working code. and try a invalid input.

I could not find the reason.

g4143 - March 21, 2015, 9:36 am

(Mar 21, 2015 09:11 AM)mr.cracker Wrote: [\[CITE\]](#) i tried above code. It doesn't redirect stdin and stdout.!!!

Can you please tell me the reason ?

I have updated my code with working code above in my question.

now when i get a prompt i entered

"LLL" : that is a wrong command.

then i got,

Code `/bin/sh: 1: LLL: not found`

After that it asks for input continuously. I think the child ask for input. Can you just check my code once more. Just run it. It is a working code. and try a invalid input.

I could not find the reason.

Well is LLL a valid bash command or program on your system? Try opening a terminal and typing LLL at the prompt.

mr.cracker - March 21, 2015, 9:51 am

Can you please tell me, why below code doesn't redirect stdin and stdout ?

Code `execl("/usr/bin/xterm", "/usr/bin/xterm", (char *)0);/*open a terminal as a child*/`