<https://blog.csdn.net/xirenwang/article/details/104139866>

程序的原意是输入一个字符串s1，经过字符检查后执行如下三条语句：

close(1);close(2);system(&s1);

我们的目的是要得到shell，官方wp给的是此处可以输入串：$0然后可以执行新的命令exec 1>&0以及其他以查看到flag

那么这条语句是什么意思呢？其中的0和1是指什么呢？

0和1是linux下的文件描述符。

在Linux中一切皆文件，文件描述符（file descriptor）是内核为了高效管理已被打开的文件所创建的索引，是一个非负整数（通常是小整数），用于指代被打开的文件，所有执行I/O操作的系统调用都通过文件描述符。程序刚刚启动的时候，0是标准输入，1是标准输出，2是标准错误。如果此时去打开一个新的文件，它的文件描述符会是3。

标准输入输出的指向是默认的，我们可以修改它们的指向，也即重定位

举例子，可以用exec 1>myoutput把标准输出重定向到myoutput文件中，也可以用exec 0<myinput把标准输入重定向到myinput文件中，而且，文件名字可以用&+文件描述符来代替。

那么问题到这里就解决了，三条语句中close(1);close(2)即把标准输出和标准错误输出关闭，然后我们可以执行 exec 1>&0，也就是把标准输出重定向到标准输入，因为默认打开一个终端后，0，1，2都指向同一个位置也就是当前终端，所以这条语句相当于重启了标准输出，此时就可以执行命令并且看得到输出了

补充:

$0是指第一个参数，argv[0]指向可执行文件的路径，也即表示sh，执行这条命令也就是打开了新的shell，这里也可以用s\h

这条命令其实属于exec操作文件描述符，本质上即exec+重定向命令，只不过操作的是文件描述符