# Linux C程序的编译过程

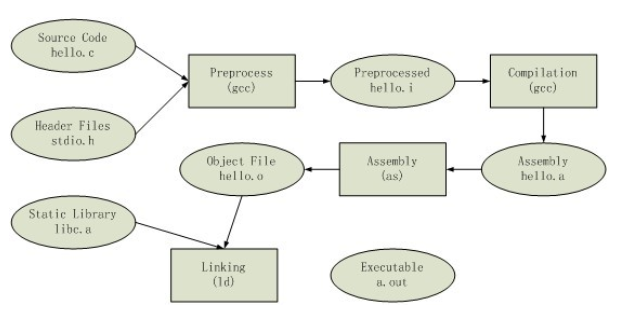
通常我们使用gcc来生成可执行程序，命令为：gcc hello.c，默认生成可执行文件a.out 其实编译（包括链接）的命令：gcc hello.c 可分解为如下4个大的步骤：

预处理**(**Preprocessing**)**

编译**(**Compilation**)**

汇编**(**Assembly**)**

链接**(**Linking**)**



下面举个例子来描述编译过程。

**1.编写c代码，并输入以下代码，生成文件hello.c**

#include <stdio.h>

#define DISPLAY "hello c!"

int main**(**void**)**

**{**

    printf**(**"%s\n"**,** DISPLAY **);**

**return** 0**;**

**}**

**2.预编译(Preprocessing)**

会对各种预处理指令（#include #define #ifdef 等#开始的代码行）进行处理，删除注释和多余的空白字符，生成一份新的代码 。

**[**root@wahoo test**]**#gcc **-**E hello**.**c **-**o hello**.**i

E 参数 通知gcc对目标文件进行预编译，这里是对文件hello**.**c文件

o 参数 是对命令输出结果进行导入操作，这里是把 gcc **-**E hello**.**c 操作结果输出到文件hello**.**i（命名要自定义）中进行保存

这个命令执行完后我们目录下多了一个文件hello.i，你可以查阅一下文件的内容。



**3.编译(Compilation)**

对代码进行语法、语义分析和错误判断，生成汇编代码文件。

**[**root@wahoo test**]**#gcc **-**S hello**.**i **-**o hello**.**s

S 参数 通知gcc对目标文件进行编译，这里是对文件hello**.**i文件

通过这一步我们知道 C语言跟汇编的 关系，至于他们之前是如何进行转换的，大家可以进行更深入的学习与探讨。 此时目录下多了一个hello.s文件，内容如图



**4.汇编(Assembly)**

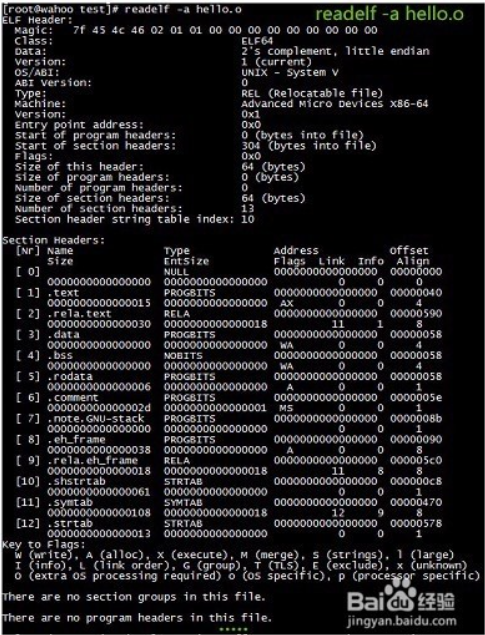
把汇编代码转换与计算机可认识的二进制文件，要知道计算机只认识0和1呢。

**[**root@wahoo test**]**#gcc **-**c hello**.**s **-**o hello**.**o

c 参数 通知gcc对目标文件执行指令转换操作

此步骤我们得到文件hello.o 。大家也同样打开文件查看一下，这个文件里面几乎没几个字符大家能看懂，这就对了，但大家可以通过这种方法将其转化为我们可读的形式。

**[**root@wahoo test**]**#readelf **-**a hello**.**o



**5.链接(Linking/Build)**

通俗的讲就是把多个\*.o文件合并成一个可执行文件，二进制指令文件。

**[**root@wahoo test**]**#gcc hello**.**o **-**o hello

这里我们就得到了一个可以直接在系统下执行的文件hello。

我们也可以对这个文件进行readelf操作，也可以进行二进制指令转汇编的操作。

**[**root@wahoo test**]**#objdump **-**d hello



**6.程序运行**

**[**root@wahoo test**]**#**./**hello

hello c**!**

