

Linux动态链接库编程入门PDF转换可能丢失图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/296/2021_2022_Linux_E5_8A_A8_E6_80_c103_296255.htm 动态链接库编程入门 动态链接库

是一种通用的软件组件技术，是多种操作系统中提供基本服务的方式。比如Win32内核就是3个DLL文件构成。这种技术在Linux操作系统下也有对应的实现，就是Linux标准对象Standard Object，对应的文件扩展名为.so。下面通过一个简单的例子开始介绍Linux标准对象。我们的标准对象文件含有一个函数，不需要声明export导出符号，只需要编译器设置即可。如下：

```
#include #include void show() { printf("Standard Object by gashero\n"). }
```

保存为myso.c文件，按照如下编译：

```
$ gcc -fPIC -shared -o libmyso.so myso.c
```

执行生成一个libmyso.so文件，按照Linux标准对象的命名惯例，应该在库名称之前加上"lib"前缀，尽管不是必须的。编译开关-fPIC代表函数符号可以重定向，-shared代表编译结果是一个标准对象。不同于Win32DLL，Linux标准对象中的所有函数都是直接导出的，都可以被调用程序所访问。下面我们编写调用程序：

```
#include int main() { printf("Invoke my so\n"). show(). return 0. }
```

保存为invoke.c，按照如下gcc开关编译：

```
$ gcc -o test invoke.c ./libmyso.so
```

编译生成test可执行文件。如上编译条件的最后一条需要是所调用的标准对象文件名，注意必须含有路径。如果只是使用libmyso.so，则必须确保这个文件在可访问的PATH下面。本例所使用的文件名"./libmyso.so"是当前路径下的，使用了相对路径。如下测试结果：

```
$. /test Invoke my so Standard Object by gashero
```

100Test 下载频道开通，各类考试题

目直接下载。详细请访问 www.100test.com