

Linux 运行的时候，是如何管理共享库(*.so)的？在 Linux 下面，共享库的寻找和加载是由 /lib/ld.so 实现的。ld.so 在标准路径(/lib, /usr/lib) 中寻找应用程序用到的共享库。

但是，如果需要用到的共享库在非标准路径，ld.so 怎么找到它呢？

目前，Linux 通用的做法是将非标准路径加入 /etc/ld.so.conf，然后运行 ldconfig 生成 /etc/ld.so.cache。ld.so 加载共享库的时候，会从 ld.so.cache 查找。

传统上，Linux 的先辈 Unix 还有一个环境变量：LD_LIBRARY_PATH 来处理非标准路径的共享库。ld.so 加载共享库的时候，也会查找这个变量所设置的路径。

```
LD_LIBRARY_PATH=$LD_LIBRARY_PATH:./lib
```

```
export LD_LIBRARY_PATH
```

1. 往/lib和/usr/lib里面加东西，是不用修改/etc/ld.so.conf的，但是完了之后要调一下 ldconfig，不然这个library 会找不到
2. 想往上面两个目录以外加东西的时候，一定要修改/etc/ld.so.conf，然后再调用ldconfig，不然也会找不到。
3. 如果想在这两个目录以外放lib，但是又不想在/etc/ld.so.conf中加东西（或者是没有权限加东西）。那也可以，就是export一个全局变 量LD_LIBRARY_PATH，然后运行程序的时候就会去这个目录中找library。一般来讲这只是一种临时的解决方案，在没有权限或临时需要的时候使用。
4. ldconfig做的这些东西都与运行程序时有关，跟编译时一点关系都没有。编译的时候还是该加-L就得加，不要混淆了。
5. 总之，就是不管做了什么关于library的变动后，最好都ldconfig一下，不然会出现一些意想不到的结果。不会花太多的时间，但是会省很多的事。