前言：

在iPhone手机与嵌入式平台通讯的方案选型中，可通过libusbmuxd映射USB网络设备的方式，绕过IOS系统USB的MFI认证流程。

Libusbmuxd库移植及整套通讯流程开发步骤如下：

1. 资源库下载

GitHub账户：<https://kgithub.com/libimobiledevice>

libimobiledevice：<https://kgithub.com/libimobiledevice/libimobiledevice>

libplist：<https://kgithub.com/libimobiledevice/libusbmuxd>

libimobiledevice-glue：<https://kgithub.com/libimobiledevice>/libimobiledevice-glue

libusbmuxd：<https://kgithub.com/libimobiledevice>/libusbmuxd

usbmuxd：<https://kgithub.com/libimobiledevice>/usbmuxd

libusb：https://kgithub.com/libusb/libusb

1. 资源库移植

平台：raspberrypi-4B

交叉编译工具：arm-linux-gnueabihf-

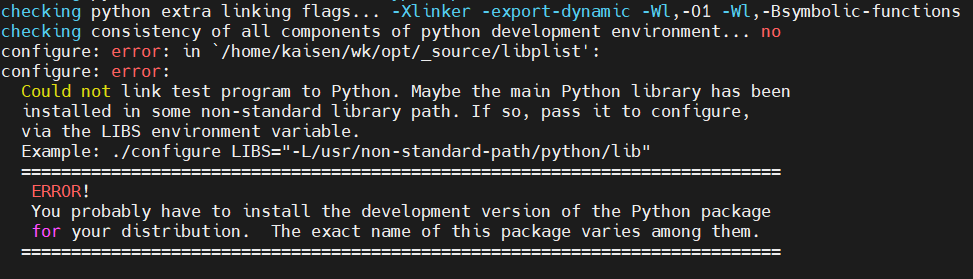
1. libplist

./configure --prefix=/home/kaisen/wk/opt/host-gnueabihf --host=arm-linux-gnueabihf

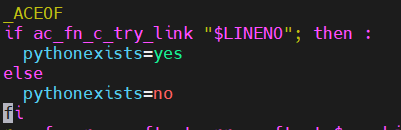
make

make install

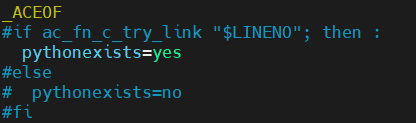
如果提示以下python错误需修改库configure文件中内容如下



将以下内容替换



替换后



再执行配置编译安装命令即可

1. libimobiledevice-glue

PKG\_CONFIG\_PATH=/home/kaisen/wk/opt/host-gnueabihf/lib/pkgconfig ./configure --prefix=/home/kaisen/wk/opt/host-gnueabihf --host=arm-linux-gnueabihf

make

make install

3、libusbmuxd

PKG\_CONFIG\_PATH=/home/kaisen/wk/opt/host-gnueabihf/lib/pkgconfig ./configure --prefix=/home/kaisen/wk/opt/host-gnueabihf --host=arm-linux-gnueabihf

make

make install

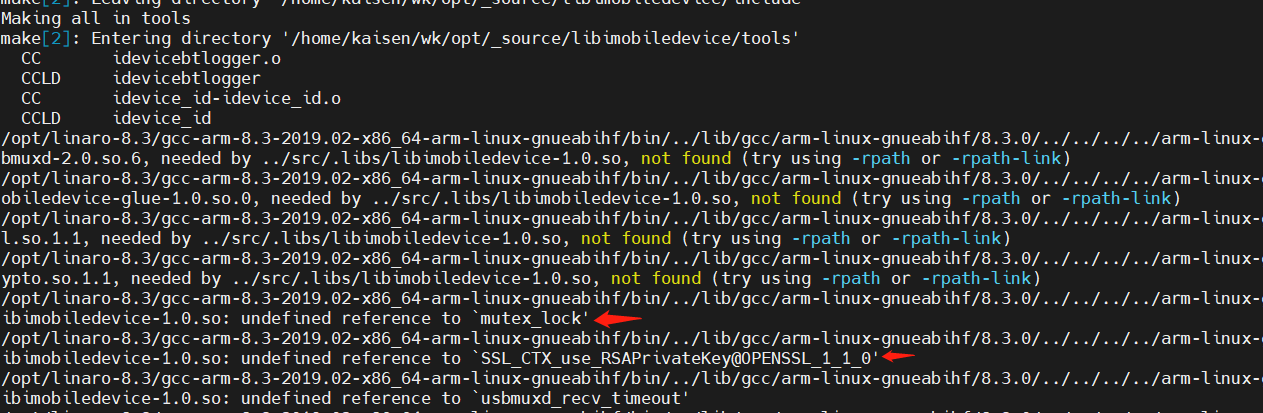
4、libimobiledevice

PKG\_CONFIG\_PATH=/home/kaisen/wk/opt/host-gnueabihf/lib/pkgconfig ./configure --prefix=/home/kaisen/wk/opt/host-guneabihf--host=arm-linux-gnueabihf

make

make install

如果在make步骤时，编译的终端界面上提示 undefined reference to `mutex\_lock' 等如下图信息，仅表示编译libimobledevice库tool里面工具时遭遇失败，但是libimobiledevice可使用库已经编译生成在 src/.libs 目录下，所以只需手动将所用的到库拷贝到开发的环境目录即可。



动态库拷贝

cd src/.libs/

tar cvf libimobiledevice.tar libimobiledevice.so\*

tar xvf libimobiledevice.tar -C /home/kaisen/wk/opt/host-gnueabihf/lib

cd ../../

头文件拷贝

cp include/libimobiledevice /home/kaisen/wk/opt/host-gnueabihf/include -rf

PC文件拷贝

cp src/libimobiledevice-1.0.pc /home/kaisen/wk/opt/host-gnueabihf/lib/pkgconfig

至此libimobiledevice库算移植完成

1. libusb

./configure --prefix=/home/kaisen/wk/opt/host-gnueabihf --host=arm-linux-gnueabihf --disable-udev

make

make install

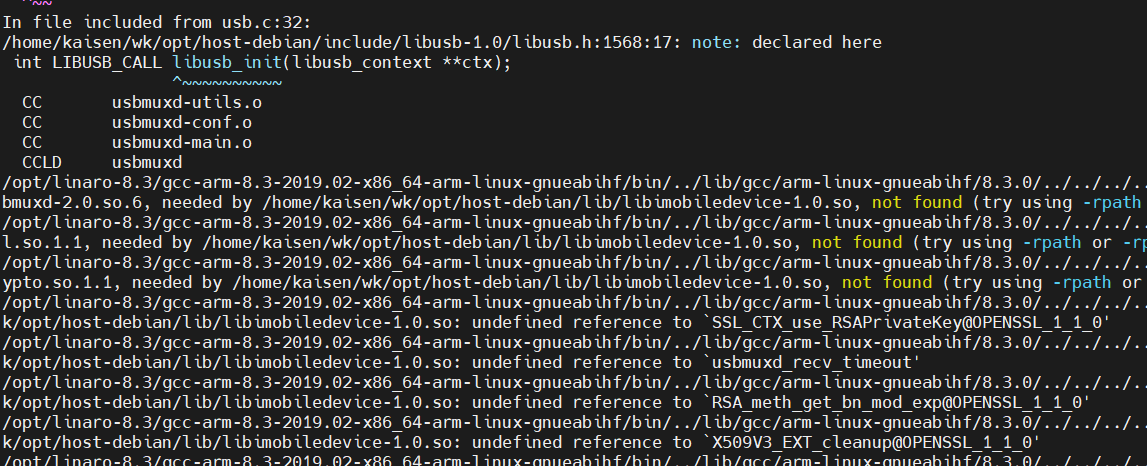
1. usbmuxd服务程序编译

原始的交叉编译命令如下：

PKG\_CONFIG\_PATH=/home/kaisen/wk/opt/host-debian/lib/pkgconfig ./configure --prefix=/home/kaisen/wk/opt/host-debian --host=arm-linux-gnueabihf

make

到make步骤时发现依旧提示跟编译libimobiledevice时一样报错，如下图



解决方案是自己再新建一个构建框架，如下：

在usbmuxd工作目录下新建以下文件及目录

-------------------------------------------------------------------------------------

| - build

| - cmake

| ---- Findimobiledevice.cmake

| ---- Findlibusb.cmake

|- CMakeLists.txt

-------------------------------------------------------------------------------------



工程编译命令如下：

cd build

cmake .. -DCMAKE\_PREFIX\_PATH=/home/kaisen/wk/opt/host-gnueabihf -DCMAKE\_C\_COMPILER=arm-linux-gnueabihf-gcc

make

1. 嵌入式板环境配置
   1. 将libimobiledevice、libimobiledevice-glue、libplist、libusb、libusbmuxd的动态库文件上传到嵌入式板的/usr/lib目录中
   2. usbmuxd服务
2. 将编译usbmuxd工程生成的usbmuxd可执行文件上传到嵌入式板/usr/bin目录中
3. 将usbmuxd工程下 udev/39-usbmuxd.rules.in 文件复制命名为 39-usbmuxd.rules 并修改39-usbmuxd.rules文件中内容如下：



然后将修改后的39-usbmuxd.rules文件上传到嵌入式板/etc/udev/rules.d目录中

1. 通讯测试

嵌入式板启动usbmuxd服务：

usbmux -s &

测试例程：



1. systemd托管usbmuxd服务

将usbmuxd工程中的 usbmuxd.service 文件修改内容并上传到嵌入式板的/lib/systemd/system/ 目录下。



创建service文本内指定的usbmuxd.target文件，存放在相同路径下



根据service文本内的指定运行命令，修改39-usbmuxd.rules文本内容



在启动脚本中添加启动systemd服务命令：

systemctl enable usbmuxd.service

systemctl start usbmuxd.service