



Tsihang's CNB

- [博客园](#)
- [首页](#)
- [新随笔](#)
- [联系](#)
- [订阅](#)
- [管理](#)

随笔- 21 文章- 1 评论- 16

【原】net-snmp学习笔记<二> — 移植

net-snmp到Powerpc平台的移植

准备工作

- 1、net-snmp5.7.1源码
- 2、Linux编译环境，IP地址192.168.2.115
- 3、Powerpc平台，本次移植到飞思卡尔，IP地址192.168.2.20

环境搭建步骤

- 1、主机-Powerpc:

主机和Powerpc互相ping，检测网口连通性；

如果网口无法工作，使用串口的zmodem进行文件下载，但是此方法较慢；

- 2、下载文件方法:

windows平台

开启tftpd.exe，配置服务器工作路径为文件所在目录；

通过串口或者网口与Powerpc通信，在Powerpcshell下运行：

```
tftp -r [download_file] 192.168.2.115 -g
```

SecureCRT使用zmodem传送文件，此方法比tftp方法速度慢。

Linux系统需要tftp服务器或者nfs服务器支持

移植net-snmp

1、配置

```
CC=/usr/freescale-2010.09/bin/powerpc-linux-gnu-gcc./configure--build=i386-linux --host=ppc-linux --disable-ipv6--with-endianness=big--disable-manuals--disable-ucd-snmp-compatibility--enable-as-needed--disable-embedded-perl --without-perl-modules--disable-snmptrap-subagent--disable-scripts--with-mib-modules="ucd_snmp" --enable-shared
```

修改项包括：平台相关交叉编译链，host取值，字节序(Powerpc属大端字节序)，其他配置不变

2、编译

本工程因为需要加载动态库，因此须使用动态编译

动态编译：

```
#make
```

编译时不需要重新指定安装目录，一律选择默认

//若有动态库，会提示错误：缺少库，拷贝过去即可。编译完成后大小41K左右

静态编译：

```
#makeLDFLAGS="-static"
```

//使用-static选项指明是静态编译，安装完成后大小1.6M左右

3、安装

```
#make install
```

编译完snmp后，在安装阶段会链接目标文件生成库文件，并将库文件默认安装在本地/usr/local/lib和 /usr/lib下

4、压缩文件

安装完成后snmpd路径：/usr/local/sbin/snmpd

```
@Powerpc-linux-strip snmpd
```

静态：会将1.6M的文件压缩为649K

动态：只有46K

拷贝文件

1、SNMPD的拷贝

将压缩后的snmpd拷贝至/usr/sbin目录下，修改权限为755；

2、配置文件的拷贝

将编译环境的/etc/snmp目录下snmpd.conf snmp.conf 这两个文件拷贝至：Powerpc板的/usr/local/share/snmp目录下；

3、MIB库文件的拷贝

将MIB库文件拷贝至Powerpc板/usr/local/share/snmp/mibs中：

拷贝SNMP系统MIB库和我们自己开发的MIB库到目标板 的/usr/local/share/snmp/mibs；

```
cp ./mib <目标板>/usr/local/share/snmp/mibs；
```

```
cp SGCC-MIB.my <目标板>/usr/local/share/snmp/mibs；
```

这样，我们自己开发的MIB库SGCC-MIB和snmp自带的MIB库就整合在一起了。

4、命令的拷贝

通过tftp或者nfs挂载方式，拷贝本地snmp相关命令到Powerpc板：

#这里是snmp相关操作方法

```
cd/usr/local/bin/
```

```
cp agentxtrapnet-snmp-c* snmp* encode_keychange /usr/bin/
```

snmp的守护线程和trap监听命令权限更改为755。

```
cp/usr/local/sbin/snmp*/usr/sbin/
```

[注]路径不定，只要在需要时将命令所在路径导入PATH即可

使用命令测试：

```
@snmpget -v2c -cprivate 127.0.0.1 sysUpTime.0 #返回正确
```

5、运行库的拷贝

通过tftp或者nfs挂载方式，拷贝本地snmp相关库文件到目标板，用以支持项目运行：

```
Cp usr/lib/libnetsnmp* /usr/lib
```

```
Cp usr/local/lib/libnetsnmp* /usr/lib
```

将需要snmp支持的第三方库——我们的arm网管程序编译而成的库文件，拷贝/usr/local/lib；

这样，我们的网管程序就可得到SNMP的支持

[注] 路径不定，只要在需要时将库文件所在路径导入LD_LIBRARY_PATH即可

6、建立自己的网管程序和SNMP之间的关系

编辑/usr/local/share/snmp/snmp.conf，增加如下一行：

```
mibs +XXX-MIB;#这是自己建立的网管程序生成的MIB库文件
```

编辑/etc/snmp/snmpd.conf文件141行，将源地址修改为trap的发送目标IP地址，端口不变；193行，使用dlmod命令，加载动态链接库文件，该库文件指向我们的网管程序入口libXXX.so，如下，dlmodsgcc/usr/local/lib/libXXX.so。启动snmp时，会自动加载该库，从而运行网管程序。

7、启动snmpd服务

```
#snmpd -f -Le -d
```

```
#snmpd -f -Lo -d&
```

```
#snmpd-Lsd
```

如果因路径问题造成第三方库无法找到或者打开，可通过导入库文件路径的方式，写入脚本文件，在启动系统的时候自动导入路径。

```
exportLD_LIBRARY_PATH=$LD_LIBRARY_PATH:/usr/local/lib
```

其他启动方式

```
#snmpd -f -Le -d log to 标准出错
```

```
#snmpd -f -Lo -d log to 标准输出
```

```
#snmpd -Lsd直接后台启动
```

```
#snmpd-f -Le -Dread_config可以使用此命令来查看日志文件信息
```

8、测试

```
@snmptranslate -Tp -IR sgcc
```

```
snmpget -v2c -c private localhost armip.0
```

SNMPv3补充：

由于每次断电再上电后，SNMP的/var/net-snmp/snmpd.conf文件会重新初始化，丢失上次创建的v3用户信息，为了每次上电后不需要再重新配置，使用以下方法：

移植完snmp后，在/usr/local/share/snmp/目录下建立snmpv3.txt文件，在文件中输入创建v3的用户名和密码，并保存。

```
createUsermifi SHA "authpassword" AESprivpassword（跟MIBBrowser设置保持一致）
```

在etc/init.d/rcS中snmp启动脚本前加入：

```
snmpd
```

```
killall-9 snmpd
```

```
cat/usr/local/share/snmp/snmpv3.txt >>/var/net-snmp/snmpd.conf
```

【完】

关于SNMP的介绍, 如果感兴趣, 请看本博客: <http://www.cnblogs.com/iTsihang/archive/2013/05/30/3109347.html>

【本博客 <http://www.cnblogs.com/iTsihang> 中原创之博文, 版权属作者所有, 欢迎转载。转载之时请保留本段内容, 否则作者将保留追究版权之权利】

分类: [开源项目](#), [网络](#)

标签: [snmp](#), [平台](#), [powepc](#), [移植](#)

绿色通道: [好文要顶](#) [关注我](#) [收藏该文](#) [与我联系](#) 



[Tsihang](#)

[关注 - 6](#)

[粉丝 - 9](#)

[+加关注](#)

0

0

(请您对文章做出评价)

» 下一篇: [【原】在51上简单模拟I2C通信](#)

posted @ 2012-10-04 12:32 [Tsihang](#) 阅读(1284) 评论(0) [编辑](#) [收藏](#)

[刷新评论](#)[刷新页面](#)[返回顶部](#)

注册用户登录后才能发表评论, 请 [登录](#) 或 [注册](#), [访问](#) 网站首页。

[【免费课程】案例: 慕课网2048私人订制](#)

[【推荐】50万行VC++源码: 大型组态工控、电力仿真CAD与GIS源码库](#)

[融云, 免费为你的App加入IM功能——让你的App“聊”起来!!](#)

[写“听云”原创博文, 赢取iPhone 6超级大奖](#)

最新IT新闻:

- [Go语言改用Git迁移到GitHub](#)
- [Google的捐赠让图灵奖奖金提高到百万美元](#)
- [中国商用飞机准备进军国际市场](#)
- [揭秘阿里生态的最后闭环](#)
- [.NET 开源核心类库: .NET Core](#)
- » [更多新闻...](#)

最新知识库文章:

- [编译器的工作过程](#)
- [如何领导一个自组织团队？](#)
- [“米粉节”背后的故事——小米网抢购系统开发实践](#)
- [什么是用户体验，什么不是？](#)
- [项目初始会议 – 如何在一次会议中达成共识](#)
- » [更多知识库文章...](#)

公告

昵称: [Tsihang](#)
园龄: [2年1个月](#)
粉丝: [9](#)
关注: [6](#)
[+加关注](#)

[≤](#)2012年10月[≥](#)

日	一	二	三	四	五	六
30	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30	31	1	2	3
4	5	6	7	8	9	10

搜索

常用链接

- [我的随笔](#)
- [我的评论](#)
- [我的参与](#)
- [最新评论](#)
- [我的标签](#)
- [更多链接](#)

我的标签

- [autoconf](#)(2)
- [automake](#)(2)
- [snmp](#)(2)
- [zebra](#)(2)
- [交换芯片](#)(2)
- [句柄](#)(1)
- [开发板](#)(1)
- [可重入性](#)(1)
- [控制帧](#)(1)
- [老化](#)(1)
- [更多](#)

随笔分类(53)

- [C语言编程](#)(13)
- [工作点滴](#)(1)

- [交换芯片驱动\(3\)](#)
- [开源项目\(10\)](#)
- [嵌入式\(14\)](#)
- [网卡驱动\(2\)](#)
- [网络\(10\)](#)

随笔档案(20)

- [2013年5月 \(2\)](#)
- [2013年2月 \(1\)](#)
- [2012年12月 \(2\)](#)
- [2012年11月 \(14\)](#)
- [2012年10月 \(1\)](#)

文章档案(1)

- [2012年11月 \(1\)](#)

最新评论

- [1. Re: 【原】zlog，软件项目中日志管理模块的不二之选](#)
- 你好，我有个问题请教一下。今天我按照你的方法，修改了你所说的那两个文件，用gcc编译不通过。我要移植到mini2440的开发板上，是不是要把makefile里的编译器改成交叉编译器，可是我不知道改哪里，请指教。 还有就是你提供的你修改过的代码下载下来，解压不出来。
- --IT-SUMMER

阅读排行榜

- [1. 【原】epoll + 多线程实现并发网络连接处理\(5734\)](#)
- [2. 【原】Fatfs 移植的那些事\(3262\)](#)
- [3. 【原】开源路由软件zebra系列之一 — 介绍和安装\(2587\)](#)
- [4. 【原】开源路由软件zebra系列之二 — 命令存储原理和使用方法\(1961\)](#)
- [5. 【原】uIP-1.0调试系列之一 ARP\(1793\)](#)

评论排行榜

- [1. 【原】epoll + 多线程实现并发网络连接处理\(4\)](#)
- [2. 【原】SQLite — 一款嵌入式开源数据库\(3\)](#)
- [3. 【原】三层网络设备对于IP报文的分片和重组处理原理\(2\)](#)
- [4. 【原】数组位映射的方法\(2\)](#)
- [5. 【原】Fatfs 移植的那些事\(2\)](#)

推荐排行榜

- [1. 【原】Fatfs 移植的那些事\(5\)](#)
- [2. 【原】BCM5633x Layer-2/3 千兆交换芯片系列之二 — 基于电信标准的二层功能的实现\(2\)](#)
- [3. 【原】开源路由软件zebra系列之一 — 介绍和安装\(1\)](#)
- [4. 【原】开源路由软件zebra系列之二 — 命令存储原理和使用方法\(1\)](#)
- [5. 【原】automake工具，帮你做软件架构，管理你的Makefile\(1\)](#)

Copyright ©2014 Tsihang