

# openwrt开发笔记

---

## 固件编译

虚拟机: Ubuntu14.04 64位, 联网, 足够硬盘空间(大于60G) 选用无涯stable版本:

1. git clone https://github.com/hi-wooya/openwrt-hiwooya-stable.git
  2. 进入openwrt源码目录, 执行 ./scripts/feeds update -a
  3. ./scripts/feeds install -a
  4. 执行 make menuconfig , 其中  
target system选Ralink RT288x/RT3xxx; subtarget选MT7620 base boards; target profile选HiWooya7620 network->ssh->openssh-stfp-server
  5. 执行 make V=99 进行编译(首次编译时间较长, 请耐心等待)
  6. 若编译成功, 则 bin/ramips/ 目录下的 openwrt-ramips-mt7620-HiWooya7620-squashfs-sysupgrade.bin 即是编译好的固件
- 

## 编译中可能遇到的问题

1. 编译过程需要下载, 需要科学上网
  2. 如果下载遇到困难, 将 3th\_MT7620A 开发套件光盘资料\配套开发所有源码\openwrt-1505-dl.tar.bz2 解压, 将里边所有文件拷到 dl 目录, 重新编译
- 

## 固件烧录

1. 开发板 COM 连接电脑, 串口工具波特率 57600, 上电, 倒数结束前按空格, 进入 uboot(串口显示 breed> 表明成功);
  2. 开发板任一 LAN 口用网线连接到电脑主机, 电脑浏览器 192.168.1.1 进入 Breed Web 恢复控制台
  3. Breed Web 恢复控制台左侧选择固件更新, 在右侧固件位置浏览选择编译好的固件, 点击上传, 确认信息后点更新, 等待完成重启
- 

## WAN口配置

/etc/config/network 为网络配置, 修改 config interface 'lan' 中的 option ipaddr , 不要跟上级网络一个网段 base-files/files/etc/config/network

## WIFI配置

1. 在 make menuconfig 时取消原网卡驱动: kernel modules->wireless drivers->kmod-rt2800-soc 取消勾选
2. 选择新驱动: 勾选 ralink->ralink-wifi, 重新编译

## 修改WiFi名与密码

- 修改配置文件方式: /etc/config/wireless
- 修改源码: openwrt-hiwooya-stable/package/hiwooya-wifi/ralink-wifi/files/lib/wifi/ralink.sh 中从后往前看

## 配置luci

- 无涯定制: make menuconfig:utilities->luciapk
- 原生: luci->1.collections->luci; 2.luci->modules->translations->chinese

## 定制固件

实际开发过程中, 我们总是会按照产品需求做固件定制, 但我们又尽量不想修改源码的一些配置, 这时我们有一种方法可以满足我们的需求。linux 的所有文件都是存放于根目录 '/' 下面, 如果需要修改某个目录下面的某个文件我们只要将对应的文件覆盖即可, 这时可以在源码目录下创建 'files' 目录, 这个目录可以看成是根目录的映射, 只要将要打包到固件的文件按照根目录的目录结构存放文件即可, 例如:

1. 修改 network 配置文件 源码 openwrt-hiwooya-stable/files/etc/config/network-> 固件 /etc/config/network
  2. 添加可执行文件 openwrt-hiwooya-stable/files/usr/sbin/setwifi-> 固件 /usr/sbin/setwifi
- 

查看系统 MTD 分配 cat /proc/mtd

查看系统 MTD 分区 cat /proc/partitions

将非文件系统分区读出来 hexdump -C /dev/mtd2

---

## 系统分区

- u-boot
  - u-boot-env
  - factory
  - firmware
  - rootfs
  - rootfs\_data
- 

## 文件系统

1. 首先，引导程序启动内核完成以后，由内核加载rootfs\_rom只读分区部分来完成系统的初步启动；
  2. rootfs\_rom只读分区采用的是linux内核支持的squashFS文件系统(一种压缩只读文件系统),加载完以后将其挂载到/rom目录(同时也挂载为/根目录)；
  3. 系统将使用JFFS2文件系统格式化的rootfs\_data可写文件分区并且将这部分挂载到/overlay目录；
  4. 系统再将/overlay透明挂载为/根目录
  5. 最后将一部分内存挂载为/tmp目录
- 

## 无线中继

aps命令扫描周围可用WiFi

```
iwpriv ra0 set SiteSurvey=1
iwpriv ra0 getsitesurvey
setwifi 账号 密码
```

---

## 防火墙

- default类型匿名节点：只有一个配置节点，默认配置
  - zone类型匿名节点：系统将LAN和WAN分为两个不同的zone，两个zone之间是隔离的
  - forwarding 类型匿名节点：用于不同zone之间的转发
  - rule类型匿名节点：用来进行配置哪些端口允许访问
  - direction类型匿名节点：用来配置实现具体端口转发功能
- 

## UPnP

安装UPnP

```
opkg update
opkg install miniupnpd
```

设置开机自启动

```
/etc/init.d/miniupnpd enable
```

查看miniupnpd是否驻留内存

```
netstat -lnp | grep miniupnpd
```

---

## 时区

为准确对时，需要采用NTP协议向网络上的时间授权服务器请求获得时间服务，再根据本地时区的配置转换为本地时间  
配置文件 /etc/config/system

---

## 服务管理

查看服务

```
ls /etc/init.d
```

服务的语言格式:

```
/etc/init.d服务名称 [命令]
```

可用命令

- start 临时开启这个服务
- stop 临时关闭这个服务
- restart 重启当前已开启服务，如果没开启就开启它
- reload 重新读取该服务的配置信息
- enable 设置该服务随系统一同启动
- disable 禁止该服务随系统一同启动

---

## SSH

SSH服务在openwrt下是通过一个名字叫dropbear的软件包实现的

防火墙限制SSH服务，所以SSH服务只面向LAN口，向WAN口开放需要新建防火墙rule

```
config rule
option name 'allow ssh'
option src 'wan'
option target 'ACCEPT'
option dest_port '22'
```

系统默认的root用户没有密码，这个时候SSH是登录不上，先使用passwd命令给root设置一个密码

---

## SCP

向开发板传文件用winscp客户端，选择SCP文件协议

---

## 4G模块

拨号 founret &

配置make menuconfig

kmod-usb-serial

kmod-usb-serial-option

kmod-usb-serial-wwan

usb-modeswitch

kmod-usb-net

---

## 配置网口灯

1. drv\_regopt放在package/kernel下面
2. reg放到package下面
3. make menuconfig Utilities ---><\*> reg
4. 编译重新烧录固件
5. reg w 0x10000060 0

---

## 华为4G模块

Vendor Id: 12d1

Product Id: 15c1 picocom /dev/ttyUSB0

拨号上网 在/etc/config/network中配置dns - echo "AT^NDISDUP=1,1" > /dev/ttyUSB2

- udhcpc -i usb0

---

## 内核补丁

openwrt源码中的Linux补丁文件放在target/linux/generic文件下面，有对于不同版本的Linux内核补丁文件

## openwrt启动流程：

```
/etc/preinit->/lib/preinit/->/etc/inittab->/etc/rc.d/S
```

---

以太网是一种广播传输技术，网络上的所有节点均能检测到网络介质上传输的帧，但一般只有自身Mac地址和帧目的Mac地址相同才会将内容复制到自己的缓冲区，交给ip层进一步检查ip信息。当网卡工作在混杂模式下时，会将所有侦听到的内容交给上层软件处理，例如tcpdump抓包软件。

---

ARP协议通过向局域网中的所有主机发送广播来查询目标ip的Mac地址

---

路由就是把报文从源主机传输到目标主机的过程，报文根据路由表来进行路由