**Lab 1: Basic Wireless Network Setup and Surveillance**

汪大洋 201311211918

其他组员：杜蓉、任梅、汪大洋

**一、实验说明及实验目的**

*Most wireless routers run Linux which is hidden behind the cloud of a fancy web interface. OpenWRT is essentially an open source Linux distribution for embedded devices, designed to fit into a small memory footprint. It was originally developed specifically for the Linksys WRT54G series routers, but very quickly other devices were discovered to be compatible. Most of these compatible devices use the same Broadcom processor, but some other processors are also compatible. The idea behind OpenWRT is to open up functionality that wasn't originally available, but also to provide a Linux frame work, for customizing your device to do far more than it was originally designed for. We will accomplish the following tasks in this lab：*

*Section 1: Equipment List*

*Section 2: Install OpenWRT*

*Section 3: Basic Wireless Network Configurations*

*Section 4: In-Depth Wireless Configurations*

*Section 5: Rudimentary Penetration Testing*

*Section 6: General Questions*

大多数无线路由器运行隐藏在云网络接口的Linux。OpenWRT本质上是一个为嵌入式设备而设计的内存占用小的开源的Linux发行版。它最初是专为开发WRT54G系列路由器而设计的,但很快发现，它与其他设备也是兼容的。这些兼容的设备中的大多数使用相同的博通处理器,但一些其他处理器也可以兼容。基于OpenWRT的想法是开放一些最初不可用的功能，同时提供一个Linux框架，来定制你的设备，使它能做的远远超过最初的设计。我们将在实验室完成以下任务：

第一部分：设备清单

第二部分：安装OpenWRT

第三部分：无线网络配置基础

第四部分：无线网络配置提高

第五部分：基本无线渗透测试

第六部分：问题

**二、实验过程：**

Section 1: Equipment List

*In this lab, you can use a NETGEAR WNDR3700v4 series access point (AP, also known as router, http://wiki.openwrt.org/toh/netgear/wndr3700), and a laptop with both Windows and Fedora Core 22 (via a Live USB). Please note that you may need to install various software tools for each Operating System (OS) to perform the lab tasks.*

笔记本（win7）、电源、鼠标、无线网卡、无线网卡驱动、fedora系统安装盘（U盘）、网线各一个。

Section 2: Install OpenWRT

*Power on your laptop into the Windows OS*

进入windows7 系统，进行接下来的操作。

1、网线连接笔记本和路由器的一个LAN口，安装Tftpd32-4.00-setup.exe，按照实验要求设置笔记本的IP。

2、按下路由器电源和恢复出厂设置按钮（restore factory settings button），直到绿灯闪亮，

3、打开tftp32，在Tftp Client选项卡中按照实验要求设置，单击put。

4、打开浏览器，跳转到192.168.1.1，设置路由器的名称和密码（root，123456）。

Section 3: Basic Wireless Network Configurations

打开putty，登录名为root，密码123456。可以看到实验要求中的界面。

3.1 Configure wireless parameters

1、 vi（vim）命令的使用：查看存储在/etc/config/wireless 中的Wireless specific (Layers 1 and 2) configurations，和存储在/etc/config/network中的Network (Layer 3) parameters。（在文件界面下，:i是修改，:w是保存，:q是退出，按两下d是删除一行）

2、cp command’s use: cp wireless wireless\_backup you can see the required interface

cp命令的使用：cp wireless wireless\_backup （备份）可以看到实验要求中的界面。

3、uci command’s use: set up wireless network parameters

uci命令的使用：设置无线网络参数

*uci set wireless.radio0.disabled=0 //启动*

*uci set wireless.radio0.channel=6 //选择频道*

*uci set wireless.@wifi-iface[0].mode=ap //accesspoint接入点（wds）*

*uci set wireless.@wifi-iface[0].ssid=BNUUUUUU //名称*

*uci set wireless.@wifi-iface[0].network=lan //连接lan口*

*uci commit wireless && wifi //确认*

可以看到实验要求中的界面。

4、输入echo -n 'xdhdkkewioddd' | hexdump -e '13/1 "%02x" "\n"' | cut -d ':' -f 1-13，设置wifi密码，得到的密码是786468646b6b6577696f646464。

//echo:显示到管道，输入到十六进制命令，去掉分隔符‘：’，生成密钥（脚本语言）

5、输入：

*uci set wireless.@wifi-iface[0].encryption=wep //设置wep加密方式*

*uci set wireless.@wifi-iface[0].key1=786468646b6b6577696f646464。 //密钥*

*uci set wireless.@wifi-iface[0].key=1 //密钥个数*

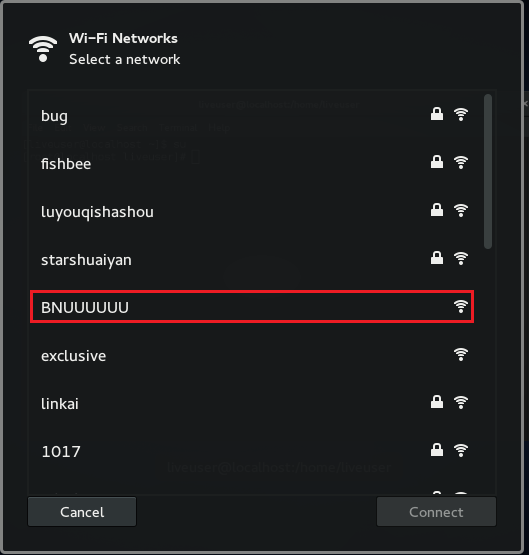
*uci commit wireless && wifi*

6、look over /etc/config/wireless，you can see the required interface.

查看/etc/config/wireless，可以看到实验要求中的界面。

7、使用ping连接，在Wireshark软件中抓包。在这一步，因为驱动等原因没能成功。

**3.1.1 结果截图**：(因为第一次截图的图片损坏，所以这里的图是后来fedora系统下的截图。)



**3.1.2 抓包截图**：（这一步最初因为驱动原因不能做而跳过，后来由于路由器不能连接的原因没有做，老师说可以以后做了再截图。）

Section 4: In-Depth Wireless Configurations

进入fedora系统进行接下来的操作。

**4.1 Transmission power adjustments 修改信号传输功率**

1、打开终端，输入ssh root@192.168.1.1，以root的身份登录，输入密码（123456），会看到和实验要求中一样的界面。

2、再打开一个终端（这样方便输入和查看功率的改变），输入iwconfig，可以看到当前无线网卡的状态。再输入iwlist wlp9s0 scanning（wlp9s0是无线网卡名称），可以看到当前无线网络的信息，找到ESSID是自己的无线名称（"BNUUUUUU"）的内容，查看Signal leval（一开始是-24）。

3、在第一个终端中输入 uci set wireless.radio0.txpwr=10

uci commit wireless && wifi

4、在另一个终端中重复输入先前的两行命令，再次查看Signal leval。

5、重复3、4步，每次的txpwr都不同。记录每次的Quality和Signal Leval数据。

**4.1.1数据记录表格**

|  |  |  |  |
| --- | --- | --- | --- |
| txpwr | mW | Quality | Signal Leval(dBm) |
| 最初 | 未知 | 70/70 | -18 |
| 0 | 0.5 | 70/70 | -23 |
| 3 | 3 | 70/70 | -18 |
| 6 | 5 | 70/70 | -24 |
| 10 | 8 | 70/70 | -17 |
| 17 | 12 | 70/70 | -22 |
| 30 | 25 | 70/70 | -25 |
| 70 | 50 | 70/70 | -21 |
| 100 | 125 | 70/70 | -20 |
| 200 | 148 | 70/70 | -23 |
| 250 | 160 | 70/70 | -24 |

**4.1.2可以注意到，并不总是信号传输功率越高，信号质量、等级（感觉这两个词不是很合适）就越好。这是为什么？**

答：接收到的信号强度除了受到功率的影响以外，还和接收端（笔记本）的配置有关系，因此，当功率过高而接收端无法接收时信号反而不如低功率时的好。但表格中的数据变化比较复杂，应该还有其他原因。

（参考：<http://net.it168.com/a2010/0204/848/000000848744_all.shtml>）

**4.1.3在实际中什么功率？**

答：首先考虑3mW、8mW、125mW这三个功率，因为它们的强度较高。平常可能不需要大功率的信号，因此选择用3mW或8mW。根据表格数据猜测，最初路由器刚设置好时的信号功率可能是3mW，所以选择3mW。

**4.2 The effects of radio path obstructions 传输通路上阻碍的影响**

用锡纸作为阻碍，用iwlist wlan0 scanning 测试Signal Quality 和 Signal Leval。记录如下：

|  |  |  |
| --- | --- | --- |
| 锡纸位置 | Quality | Signal Leval(dBm) |
| 没有放锡纸 | 70/70 | -16 |
| 锡纸覆盖在路由器上 | 70/70 | -20 |
| 路由器位置移动后完全被锡纸包住 | 70/70 | -32 |
| 揭开锡纸，不改变路由器位置 | 70/70 | -18 |
| 锡纸在路由器上方约10cm的地方 | 70/70 | -28 |
| 锡纸在路由器上方约20~30cm的地方 | 70/70 | -16 |
| 锡纸彻底拿开 | 56/70 | -54 |
| 保持上次的状态再测一次 | 70/70 | -21 |
| 锡纸在电脑和路由器的中间 | 70/70 | -39 |
| 锡纸紧靠路由器靠近电脑一侧 | 55/70 | -55 |
| 锡纸包裹电脑屏幕边缘（天线） | 70/70 | -22 |

**4.3 选作（现在还没做）**

Section 5: Rudimentary Penetration Testing

1、在fedora的终端中输入ssh 192.168.1.1，登录，输入如下语句，将无线网络修改成新名字并隐藏。

uci set wireless.@wifi-iface[0].hidden=1

uci set wireless.@wifi-iface[0].ssid=BNU++（改成不同名字是因为可能先前的网络名称被一些wireless scanning 程序而存储）

uci commit wireless && wifi

可以发现，BNU++无法被搜索到。

2、输入yum –y install aircrack–ng，安装aircrack。

3、程序安装完成后，断开当前网络。输入chkconfig NetworkManager off，看到相关信息后，输入service NetworkManager stop。

4、输入 airmon-ng start wlp9s0，airodump-ng –w test wlp9s0，用手机或是其他同学的设备搜索无线网络，可以看到接下来出现的信息中有自己的无线SSID。

**5.1.1 为什么可以成功看到自己的无线网的SSID？**

答：aircrack软件使我们的设备进入监听模式（？），因此在不断的像外界发信号，当有设备在检索无线网络时，aircrack通过抓包能够捕捉到数据的流动，从而找到无线SSID的名称。

**5.1.2 为什么隐藏SSID无法提高网络安全？**

答：主要有以下几点：

1）无线SSID本身只是一个和其他网络区分的名称，并非设计成隐藏的，因此隐藏SSID并不能提高安全性；

2）由实验可知，找到隐藏的SSID十分容易；

3）微软的Technet解释为何隐藏SSID不是一项安全特性如下：

非广播网络并非不可探测。其设置为，由无线客户端发送探测请求，无线接入点发送回应。不同于广播网，当设置为连接到非广播网络，运行winxp sp2或win2003 sp1的无线客户端会不停广播这些网络的SSID，即使它们不在通信范围内。

由此，使用非广播网络削弱了无线客户端网络配置的隐私性，因为它会定期广播其优先的非广播网络配置。

（以上内容参考：

<http://www.aircrack-ng.org/>，

<http://article.yeeyan.org/view/127744/168572/>）

**三、实验中遇到主要问题**

1、抓包时，一直无法抓到无线网络的包，发现是因为没有ping，ping之后依旧不行，猜测是因为无线网络设置了密码，取消密码后还是不成功，最后发现可能是驱动的问题。解决这个问题要安装新的软件，现在还没有做。

2、很多终端下的命令都不熟悉，比如查看无线网络状态的命令：iwconfig和iwlist wlp9s0 scanning是在询问老师后知道的。

3、在第五部分抓包找隐藏SSID时，如果是先用设备搜索，然后再输入命令，就会始终能看到那个SSID，无法知道这个SSID到底隐藏了没。

**四、总结与体会**

通过这次实验，学到了许多无线网络设置的方法和操作，这些方法和操作都很实用。比如关于路由器的设置、关于wifi信号强弱的测试等等。

实验的内容有些多，而且以前没有接触过相关内容，因此一开始做起来有些困难，而成功之后的感悟也更丰富。在两天的实验中，对linux系统也有一些体会，初步使用了一些简单的shell命令。

很多时候实验现象和理论预测不同甚至截然相反，这时候应该重做几次，来判断是因为实验有干扰、操作中有不对的地方或是理论并不完全正确。

**五、其他参考资料：**

由于内容较多，因此只列出资料来源。

1、关于ssh命令：

<http://os.51cto.com/art/201205/335402.htm>

2、关于vi命令：

参考百度文库中的vi命令手册

3、关于uci命令：

<http://blog.csdn.net/jk110333/article/details/19479747>

4、关于iwconfig和iwlist：

iwconfig的百度百科

<http://blog.chinaunix.net/uid-9185047-id-445208.html>

5、chkconfig off 和 service stop命令：

chkconfig百度百科

<http://www.jbxue.com/LINUXjishu/20642.html>

6、使用openWRT而不是其他同类软件的原因：

http://www.leiphone.com/news/201406/openwrt-ten-years.html