

Wireless Router with a 3G/UMTS/HSDPA dongle (OPENWRT 无线网络连接)

原创

[adeam6050656](#) 于 2009-11-12 16:14:00 发布 3845 收藏

文章标签: [网络 encryption interface network authentication command](#)

版权声明: 本文为博主原创文章, 遵循 [CC 4.0 BY-SA](#) 版权协议, 转载请附上原文出处链接和本声明。

本文链接: <https://blog.csdn.net/adeam6050656/article/details/4803079>

版权

Wireless Router with a 3G/UMTS/HSDPA dongle

This article describes how I set up a wireless access point that is connected to the Internet using a USB-based 3G/UMTS/HSDPA dongle.

The software and version I use is [OpenWRT](#) version 8.09.1. For hardware, I'm using a Asus WL-500G Premium, but any platform supported by OpenWRT should work. The 3G/UMTS/HSDPA provider is the Swedish 3 and the USB dongle is the Huawei E220.

If you don't care about the 3G setup, and care more about WDS or PSK2 encryption, you may find my [home wireless network](#) writeup more relevant. If you have an older PCMCIA based 3G/UMTS card you may want to compare this article with my [summer house wireless+3g network](#) article (however that uses older OpenWRT versions).

If you want to comment on this, please go to the [blog entry](#).

Created by Simon Josefsson on 2009-03-05. Updated for OpenWRT 8.09.1 on 2009-06-04.

Chosing OpenWRT Image

You may build your own OpenWRT image, or you may download one from [openwrt.org](#).

For the ASUS WL-500gP device I chose the brcm-2.4 variant of OpenWRT for this device. The plain brcm variant will work, and uses the newer 2.6 kernel, but wireless will not work.

Downloading Image

To avoid building the image yourself, you may download it from [openwrt.org](#).

```
$ wget --quiet http://downloads.openwrt.org/kamikaze/8.09.1/brcm-2.4/openwrt-brcm-2.4-squashfs.trx
$ md5sum openwrt-brcm-2.4-squashfs.trx
6ccb44d015a8477e2eb0fa81b9b83f1a openwrt-brcm-2.4-squashfs.trx
```

Building OpenWRT

You may skip this section if you do not want to build your own image.

Building OpenWRT is quite easy these days. I'm using Debian x86 lenny, and you will need to install gcc and other development tools. Here are the commands I used:

```
svn co https://svn.openwrt.org/openwrt/trunk/  
mv trunk openwrt  
cd openwrt  
make menuconfig  
# toggle 'select all packages by default'  
# quit and save config  
make
```

That's it! The complete build takes up about 2GB on my machine, and it will take quite some time to build (up to several hours).

The output files will be stored into the bin/ directory.

Installing OpenWRT

There are generic [Installation instructions](#) on the the [OpenWRT Wiki](#).

You'll have to use a ethernet cable at this point. Connect it to LAN1-LAN4, not WAN. Configure your local machine on the 192.168.1.x network, for example as 192.168.1.42. The router will use 192.168.1.1.

On many routers, including the [Asus WL-500g Premium](#) that I use, you flash an image by disconnecting power, press and hold down the reset button, and connect the power again. Wait a few seconds and the PWR led will start to blink. Release the reset button. The device will now have a tftp server running on 192.168.1.1.

```
$ tftp 192.168.1.1  
tftp> trace  
Packet tracing on.  
tftp> binary  
tftp> put openwrt-brcm-2.4-squashfs.trx  
...
```

Wait a minute and restart the box by disconnecting and reconnecting power. Some documentations claim that the device should restart by itself but I have never seen this happen.

Configuring OpenWRT

When the router has restarted, it will have a telnet server running listening on 192.168.1.1.

```

$ telnet 192.168.1.1
Trying 192.168.1.1...
Connected to 192.168.1.1.
Escape character is '^]'.
=== IMPORTANT =====
Use 'passwd' to set your login password
this will disable telnet and enable SSH
-----

BusyBox v1.11.2 (2009-05-28 19:38:17 UTC) built-in shell (ash)
Enter 'help' for a list of built-in commands.

```

```

      _____
     |         | .----- .----- .----- | | | | .----- | | _
     |  -   ||  _ |  -__|   || | | | | |  _||  _|
     |_____| ||  _|_____|_|_|_|_____|_|_|_|_____|_|_|_|_____|
           |__| W I R E L E S S   F R E E D O M
KAMIKAZE (8.09.1, r16278) -----
* 10 oz Vodka      Shake well with ice and strain
* 10 oz Triple sec mixture into 10 shot glasses.
* 10 oz lime juice Salute!
-----

root@OpenWrt:/#

```

Now you can configure OpenWRT. I'll go through each step I made, but obviously you may deviate from this description.

Set hostname

My first step was to set a hostname on the machine, use vi to edit `/etc/config/system` to make it read:

```

config system
    option hostname caramel

```

If you prefer to cut'n'paste a command line invocation to do this, try:

```

root@OpenWrt:~# cat /etc/config/system | sed -e 's/OpenWrt/caramel/' > /etc/config/system

```

Setting a hostname will help when you have multiple OpenWRT boxes running, as the command prompt will show your hostname instead of `OpenWRT` (after you have restarted the router).

Set IP address

If you want the device to use another IP address than 192.168.1.1 you can modify `/etc/config/network`, in the LAN configuration section. A quick command to do this:

```

root@OpenWrt:~# cat /etc/config/network | sed -e 's/192.168.1.1/192.168.1.15/' > /etc/config/network

```

Replace telnet with SSH

As you do not want to use insecure telnet, you will need to set up ssh. You can type `passwd` to set a root password, and this will disable telnet. I dislike passwords, even if used over ssh, so I always set up public-key authentication instead. For reference, there is a [dropbear public key authentication howto for OpenWRT wiki page](#).

The command to make your ssh key authorized to login as root on the router is:

```
root@OpenWrt:/# echo 'PUBLICKEY' > /etc/dropbear/authorized_keys
```

Replace PUBLICKEY with the content of the ~/.ssh/id*.pub file on your own machine.

Disable password based login using:

```
root@OpenWrt:/# cat /etc/config/dropbear | sed -e "s/'on'/'off'/" > /etc/config/dropbear
```

Alternatively, edit /etc/config/dropbear to look like:

```
config dropbear
    option PasswordAuth 'off'
    option Port          '22'
```

At this point, you can print the ssh host key fingerprint using:

```
root@OpenWrt:~# dropbearkey -y -f /etc/dropbear/dropbear_rsa_host_key
Public key portion is:
ssh-rsa AAAA... root@OpenWrt
Fingerprint: md5 b4:...
```

To use the web interface, you may want to set a root password.

```
root@OpenWrt:/# passwd
Changing password for root
New password:
Retype password:
Password for root changed by root
root@OpenWrt:/#
```

If you do not set a root password, you will need to manually disable the telnet server as follows.

```
root@OpenWrt:~# rm /etc/rc.d/S50telnet
```

After you have made your changes, try the new configuration by rebooting both devices.

```
root@OpenWrt:/etc/config# reboot
```

You should now be able to login as root using SSH. Try it from your machine. The SSH fingerprint should match what was printed by *dropbearkey* earlier.

```
jas@mocca:~$ ssh -l root 192.168.1.15
The authenticity of host '192.168.1.15 (192.168.1.15)' can't be established.
RSA key fingerprint is b4:...
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '192.168.1.15' (RSA) to the list of known hosts.
```

```
BusyBox v1.11.2 (2009-05-28 19:38:17 UTC) built-in shell (ash)
Enter 'help' for a list of built-in commands.
```

```

  _____
 |         | .----- .----- .----- .| | | | .----- .| | | | | | | |
 |  -   ||  _  |  _  |  || | | |  ||  _  |  |
 |_____| |  _  |_____|_____|_____|_____|_____|_____|_____|
           |__| W I R E L E S S   F R E E D O M
KAMIKAZE (8.09.1, r16278) -----
* 10 oz Vodka          Shake well with ice and strain
* 10 oz Triple sec    mixture into 10 shot glasses.
* 10 oz lime juice    Salute!
-----
root@caramel:~#
```

Enabling 3G/UMTS

The USB drivers are not installed by default. You'll have to copy some packages onto the device somehow. The simplest is if you have access to the Internet via ethernet. Then just update the package list:

```
root@caramel:~# opkg update
Downloading http://downloads.openwrt.org/kamikaze/8.09.1/brcm-2.4/packages/Packages.gz
Connecting to downloads.openwrt.org (195.56.146.238:80)
Packages.gz          100% |*****| 130k 00:00:00 ETA
Inflating http://downloads.openwrt.org/kamikaze/8.09.1/brcm-2.4/packages/Packages.gz
Updated list of available packages in /var/opkg-lists/snapshots
root@caramel:~#
```

You will need to re-run *opkg update* after every restart of the router to be able to install packages over the Internet.

For USB I needed the *kmod-usb-serial* and *kmod-usb-uhci* packages. If the *kmod-usb-uhci* package doesn't work on your router, try the *kmod-usb-ohci* package.

```
root@caramel:/# opkg install kmod-usb-uhci kmod-usb-serial
Installing kmod-usb-uhci (2.4.35.4-brcm-2.4-1) to root...
Downloading http://downloads.openwrt.org/kamikaze/8.09.1/brcm-2.4/packages/kmod-usb-uhci_2.4.35.4-brcm-2.4-1_mipsel.ipk
Connecting to downloads.openwrt.org (195.56.146.238:80)
kmod-usb-uhci_2.4.35 100% |*****| 21245 00:00:00 ETA
Installing kmod-usb-core (2.4.35.4-brcm-2.4-1) to root...
Downloading http://downloads.openwrt.org/kamikaze/8.09.1/brcm-2.4/packages/kmod-usb-core_2.4.35.4-brcm-2.4-1_mipsel.ipk
Connecting to downloads.openwrt.org (195.56.146.238:80)
kmod-usb-core_2.4.35 100% |*****| 46546 --:--:-- ETA
Installing kmod-usb-serial (2.4.35.4-brcm-2.4-1) to root...
Downloading http://downloads.openwrt.org/kamikaze/8.09.1/brcm-2.4/packages/kmod-usb-serial_2.4.35.4-brcm-2.4-1_mipsel.ipk
Connecting to downloads.openwrt.org (195.56.146.238:80)
kmod-usb-serial_2.4. 100% |*****| 14767 --:--:-- ETA
Configuring kmod-usb-core
Configuring kmod-usb-serial
Configuring kmod-usb-uhci
root@caramel:/#
```

If it works, you should see something like this:

```
root@caramel:/# dmesg|tail
usb.c: new USB bus registered, assigned bus number 1
hub.c: USB hub found
hub.c: 2 ports detected
PCI: Enabling device 01:03.1 (0000 -> 0001)
uhci.c: USB UHCI at I/O 0x120, IRQ 2
usb.c: new USB bus registered, assigned bus number 2
hub.c: USB hub found
hub.c: 2 ports detected
hub.c: new USB device 01:03.0-1, assigned address 2
usb.c: USB device 2 (vend/prod 0x12d1/0x1003) is not claimed by any active driver.
root@caramel:/#
```

To claim the device, you need to configure the usbserial module and reboot the router. Here is how to do it:

```
root@caramel:/# cat>/etc/modules.d/60-usb-serial
usbserial vendor=0x12d1 product=0x1003 maxSize=4096
root@caramel:/# reboot
```

If you are using some other device than the E220 you will need to modify the USB vendor/product IDs as appropriate.

Reboot the box, and if you are lucky you should see something like this in the kernel log:

```
usbserial.c: Generic converter detected
usbserial.c: Generic converter now attached to ttyUSB0 (or usb/tts/0 for devfs)
usbserial.c: Generic converter detected
usbserial.c: Generic converter now attached to ttyUSB1 (or usb/tts/1 for devfs)
usbserial.c: Generic converter detected
usbserial.c: Generic converter now attached to ttyUSB2 (or usb/tts/2 for devfs)
```

If that doesn't happen, you may need a tool that nudges the modem into revealing its serial ports (by default it is a storage device). I believe the E220 will reveal its serial ports automatically after a few minutes, so if you have the patience to wait you should not need this step. If you decide to speed things up, the package [huaweiaktbbo](#) helps here. Don't forget to run `opkg update` first.

```
root@caramel:/# opkg install huaweiaktbbo
Installing huaweiaktbbo (1) to root...
Downloading http://downloads.openwrt.org/kamikaze/8.09.1/brcm-2.4/packages/huaweiaktbbo_1_mipsel.ipk
Connecting to downloads.openwrt.org (195.56.146.238:80)
huaweiaktbbo_1_mipse 100% |*****| 3073  --:--:-- ETA
Installing libusb (0.1.12-2) to root...
Downloading http://downloads.openwrt.org/kamikaze/8.09.1/brcm-2.4/packages/libusb_0.1.12-2_mipsel.ipk
Connecting to downloads.openwrt.org (195.56.146.238:80)
libusb_0.1.12-2_mips 100% |*****| 10523  --:--:-- ETA
Configuring huaweiaktbbo
Configuring libusb
root@caramel:/#
```

The tool is used as follows.

```
root@caramel:/# huaweiAktBbo
Hladam HUAWEI E220 a prepnem na modem - bbo 06
4 set feature request returned 0
Prepnute-OK, Mas ttyUSB0 ttyUSB1 (cez usbserial vendor=0x12d1 product=0x1003)
pozri /proc/bus/usb/devices
root@caramel:/#
```

To run this command automatically after each reboot, you can add a small init.d script as follows (instead of ^D press Control-D):

```
root@caramel:~# cat>/etc/init.d/huaweiAktBbo
#!/bin/sh /etc/rc.common
# Written by Simon Josefsson 2009-03-05. Released into the public domain.

START=70
start() {
    huaweiAktBbo
}
^D
root@caramel:~# chmod +x /etc/init.d/huaweiAktBbo
root@caramel:~# /etc/init.d/huaweiAktBbo enable
```

The next step is to get dial-up PPP working. The [comgt](#) package contains several useful 3G/UMTS scripts.

```
root@caramel:/# opkg install comgt
Installing comgt (0.32-1.1) to root...
Downloading http://downloads.openwrt.org/kamikaze/8.09.1/brcm-2.4/packages/comgt_0.32-1.1_mipsel.ipk
Connecting to downloads.openwrt.org (195.56.146.238:80)
comgt_0.32-1.1_mipse 100% |*****| 22840  --:--:-- ETA
Installing chat (2.4.3-11.1) to root...
Downloading http://downloads.openwrt.org/kamikaze/8.09.1/brcm-2.4/packages/chat_2.4.3-11.1_mipsel.ipk
Connecting to downloads.openwrt.org (195.56.146.238:80)
chat_2.4.3-11.1_mips 100% |*****| 10705  --:--:-- ETA
Configuring chat
Configuring comgt
root@caramel:/#
```

The next step is to configure a network interface for the 3G/UMTS/HSDPA interface. I ended up modifying the *wan* section of */etc/config/network* to read as follows:

```
#### WAN configuration
config interface wan
    option ifname    ppp0
    option proto     3g
    option device    /dev/usb/tts/0
    option apn       bredband.tre.se
    option pincode   1234
```

You will need to modify the *apn* and *pincode* settings as appropriate.

If you for some reason wish to keep the ethernet WAN port on the box as the "real" Internet connection (maybe to use the 3G part only as a backup Internet connection in case the other one goes down?), instead of replacing the existing WAN section in the file, add a new section:

```
#### 3G/UMTS configuration
config interface ppp0
    option ifname    ppp0
    option proto     3g
    option device    /dev/usb/tts/0
    option apn       bredband.tre.se
    option pincode   1234
```

Now the commands *ifup wan* (or *ifup ppp0* if you use the second configuration) should work. The speed I got when downloading a Debian CD image is around 235kb/s (1880Mbps) which is sufficiently fast that I haven't looked into ways to increase speed.

To make the interface come up automatically on startup, you can add a small init.d script like this (replace *^D* with *Control-D* as before):


```
root@caramel:/etc/rc.d# cat>/etc/init.d/wan
#!/bin/sh /etc/rc.common
# Written by Simon Josefsson 2009-03-05. Released into the public domain.

START=90
start() {
    ifup wan
}
^D
root@caramel:/etc/rc.d# chmod +x /etc/init.d/wan
root@caramel:/etc/rc.d# /etc/init.d/wan enable
root@caramel:/etc/rc.d#
```

Replace *ifup wan* with *ifup ppp0* as appropriate.

Setup wireless

Wireless is disabled by default, and you will have to modify */etc/config/wireless* to configure it. To use PSK2 encryption on your wireless link, here is a sample configuration:

```
config wifi-device w10
    option type      broadcom
    option channel   5

config wifi-iface
    option device    w10
    option network   lan
    option mode      ap
    option ssid      YourNetworkName
    option encryption psk2
    option key       DEADBEEFDEADBEEFDEADBEEFDEADBEEF
```

Remember, your security is never better than your password.

If you want more fancy wireless configurations, there are many documents out there to help you. I have written about my [home wireless network](#) setup which may be of interest.

Remote logins

I want to be able to access the router remotely and it doesn't have a static IP address. The first step is to make sure I can find the IP address of the router remotely. Installing and using *ez-ipupdate* is easy.

```
root@caramel:/# opkg install ez-ipupdate
Installing ez-ipupdate (3.0.11b8-3) to root...
Downloading http://downloads.openwrt.org/kamikaze/8.09.1/brcm-2.4/packages/ez-ipupdate_3.0.11b8-3_mipsel.ipk
Connecting to downloads.openwrt.org (195.56.146.238:80)
ez-ipupdate_3.0.11b8 100% |*****| 27613  --:--:-- ETA
Configuring ez-ipupdate
root@caramel:/#
```

I'm using DynDNS which doesn't cost anything and have been working without problems for me so far. My */etc/config/ddns* file looks like:

```
config ddns
    option enabled 1
    option interface wan
    option program ez-ipupdate
    option service dyndns
    option username USERNAME
    option password PASSWORD
    option hostname HOSTNAME

# config ddns
#     option interface wan
#     option program ez-ipupdate
#     option config /etc/ez-ipupdate.conf
```

Replace USERNAME and PASSWORD with your DynDNS account information. Replace HOSTNAME with the hostname you've registered. Note that you must change the value for "enabled" from 0 to 1.

You also need to open up the firewall to allow incoming connections on port 22 (the ssh port), which can be done using a small init.d script like this:

```
root@caramel:~# cat>/etc/init.d/open-wan-ssh-port
#!/bin/sh /etc/rc.common
# Written by Simon Josefsson 2009-03-05. Released into the public domain.

START=90
start() {
    iptables --append input_wan --protocol tcp --dport 22 --jump ACCEPT
}
^D
root@caramel:~# chmod +x /etc/init.d/open-wan-ssh-port
root@caramel:~# /etc/init.d/open-wan-ssh-port enable
```

Additional software

You can use *opkg* to install additional tools easily. First run *opkg update* to update the package list:

```
root@caramel:~# opkg update
Downloading http://downloads.openwrt.org/kamikaze/8.09.1/brcm-2.4/packages/Packages.gz
Connecting to downloads.openwrt.org (195.56.146.238:80)
Packages.gz          100% |*****| 130k 00:00:00 ETA
Inflating http://downloads.openwrt.org/kamikaze/8.09.1/brcm-2.4/packages/Packages.gz
Updated list of available packages in /var/opkg-lists/snapshots
root@caramel:~#
```

Use *opkg list* to list all available packages.

Useful tools to install on a typical router includes avahi-daemon, miniupnpd, and openntpd.

Avahi allows DNS based service discovery. Install and enable as follows:

```

root@caramel:~# opkg install avahi-daemon
Installing avahi-daemon (0.6.23-2) to root...
Downloading http://downloads.openwrt.org/kamikaze/8.09.1/brcm-2.4/packages/avahi-daemon_0.6.23-2_mipsel.ipk
Connecting to downloads.openwrt.org (195.56.146.238:80)
avahi-daemon_0.6.23- 100% |*****| 19698 00:00:00 ETA
Installing libavahi (0.6.23-2) to root...
Downloading http://downloads.openwrt.org/kamikaze/8.09.1/brcm-2.4/packages/libavahi_0.6.23-2_mipsel.ipk
Connecting to downloads.openwrt.org (195.56.146.238:80)
libavahi_0.6.23-2_mi 100% |*****| 69773 00:00:00 ETA
Installing libdaemon (0.12-1) to root...
Downloading http://downloads.openwrt.org/kamikaze/8.09.1/brcm-2.4/packages/libdaemon_0.12-1_mipsel.ipk
Connecting to downloads.openwrt.org (195.56.146.238:80)
libdaemon_0.12-1_mip 100% |*****| 8352 --:--:-- ETA
Installing libpthread (0.9.29-14.1) to root...
Downloading http://downloads.openwrt.org/kamikaze/8.09.1/brcm-2.4/packages/libpthread_0.9.29-14.1_mipsel.ipk
Connecting to downloads.openwrt.org (195.56.146.238:80)
libpthread_0.9.29-14 100% |*****| 19278 00:00:00 ETA
Installing libexpat (1.95.8-1) to root...
Downloading http://downloads.openwrt.org/kamikaze/8.09.1/brcm-2.4/packages/libexpat_1.95.8-1_mipsel.ipk
Connecting to downloads.openwrt.org (195.56.146.238:80)
libexpat_1.95.8-1_mi 100% |*****| 47267 00:00:00 ETA
Configuring avahi-daemon
Configuring libavahi
Configuring libdaemon
Configuring libexpat
Configuring libpthread
root@caramel:~# /etc/init.d/avahi-daemon enable
root@caramel:~#

```

Bittorrent clients, for example, can use UPnP to open up ports in the firewall. For this to work, you need a UPnP server on your router:

```

root@caramel:~# opkg install miniupnpd luci-app-upnp
Installing miniupnpd (1.1-5) to root...
Downloading http://downloads.openwrt.org/kamikaze/8.09.1/brcm-2.4/packages/miniupnpd_1.1-5_mipsel.ipk
Connecting to downloads.openwrt.org (195.56.146.238:80)
miniupnpd_1.1-5_mips 100% |*****| 36923 00:00:00 ETA
Installing luci-app-upnp (0.8.7-1) to root...
Downloading http://downloads.openwrt.org/kamikaze/8.09.1/brcm-2.4/packages/luci-app-upnp_0.8.7-1_mipsel.ipk
Connecting to downloads.openwrt.org (195.56.146.238:80)
luci-app-upnp_0.8.7- 100% |*****| 2748 00:00:00 ETA
Configuring luci-app-upnp
Configuring miniupnpd
root@caramel:~# /etc/init.d/miniupnpd enable
root@caramel:~#

```

The clock will typically be way off unless you set it. Install a NTP client to automate this. (The package *openntpd* also works but takes more space.)

```
root@caramel:~# opkg install luci-app-ntpc
Installing luci-app-ntpc (0.8.7-1) to root...
Downloading http://downloads.openwrt.org/kamikaze/8.09.1/brcm-2.4/packages/luci-app-ntpc_0.8.7-1_mipsel.ipk
Connecting to downloads.openwrt.org (195.56.146.238:80)
luci-app-ntpc_0.8.7- 100% |*****| 2894 00:00:00 ETA
Installing ntpclient (2007_365-1) to root...
Downloading http://downloads.openwrt.org/kamikaze/8.09.1/brcm-2.4/packages/ntpclient_2007_365-1_mipsel.ipk
Connecting to downloads.openwrt.org (195.56.146.238:80)
ntpclient_2007_365-1 100% |*****| 12512 00:00:00 ETA
Configuring luci-app-ntpc
Configuring ntpclient
root@caramel:~#
```

The End

[Comments or feedback?](#)

This work is licensed under a [Creative Commons Attribution-Share Alike 3.0 Unported License](#).



[创作打卡挑战赛](#) >
[赢取流量/现金/CSDN周边激励大奖](#)