Apple iMac Wireless (BCM 4328) with Linux (Fedora Core 10)
(Last Revision 26112009)

Background Information

Basically I wrote this document because I am seriously annoyed in terms of software support for the Broadcom Wireless BCM4328 revision 5 under even the Linux 2.6.30.x kernel.

The bm43 module is basically broken at least for the BCM4328 revision 5 and the b43-cutter method to get is also not quite functional (I believe it only supports to revision 4 for this chip). The only way that I could get this functional is by using ndiswrapper.

You could use the rpmfusion of ndiswrapper method but that’s not really an option for me as have a custom kernel.

Requirements

Functional Installation Linux, in my case a Fedora Core 10 installation with some sort of custom kernel. If you don’t use a custom kernel I would add rpmfusion repository to yum and install it from there. This method should be application for any other Linux installations via source code.

Very important point to note, if you have a custom kernel make sure your source tree of your build is on the system. And I meant everything ie in my example below it shows you the kernel and where it looks for the source code for my build. It should have .config file (which contains the build information) and the “include” directory which contains the headers.
ashley@CSSE2301:/usr/local/src/linux-2.6.30.5:601> uname -a
Linux CSSE2301 2.6.30.5 #1 SMP PREEMPT Wed Aug 19 12:40:43 WST 2009 i686 i686 i386 GNU/Linux

ashley@CSSE2301:/lib/modules/2.6.30.5:597> pwd
/lib/modules/2.6.30.5

ashley@CSSE2301:/lib/modules/2.6.30.5:598> ls -al
 total 2552
 drwxr-xr-x 3 root root  4096 2009-08-20 12:29 .
drwxr-xr-x 9 root root  4096 2009-08-20 12:22 ..
drwxr-xr-x 9 root root  4096 2009-08-19 13:56 kernel/
-rw-r--r-- 1 root root  386563 2009-08-20 12:29 modules.alias
-rw-r--r-- 1 root root  333846 2009-08-20 12:29 modules.alias.bin
-rw-r--r-- 1 root root   69 2009-08-20 12:29 modules.ccwmap
-rw-r--r-- 1 root root  264586 2009-08-20 12:29 modules.dep
-rw-r--r-- 1 root root  348350 2009-08-20 12:29 modules.dep.bin
-rw-r--r-- 1 root root  813 2009-08-20 12:29 modules.ieee1394map
-rw-r--r-- 1 root root  218 2009-08-20 12:29 modules.inputmap
-rw-r--r-- 1 root root 18594 2009-08-20 12:29 modules.isapnpmap
-rw-r--r-- 1 root root   74 2009-08-20 12:29 modules.ofmap
-rw-r--r-- 1 root root  56682 2009-08-19 13:56 modules.order
-rw-r--r-- 1 root root 248893 2009-08-20 12:29 modules.pcmmap
-rw-r--r-- 1 root root   883 2009-08-20 12:29 modules.seriomap
-rw-r--r-- 1 root root 137832 2009-08-20 12:29 modules.symbols
-rw-r--r-- 1 root root 184591 2009-08-20 12:29 modules.symbols.bin
-rw-r--r-- 1 root root  541560 2009-08-20 12:29 modules.usbmap

ashley@CSSE2301:/lib/modules/2.6.30.5:599> cd /usr/local/src/linux-2.6.30.5
ashley@CSSE2301:/usr/local/src/linux-2.6.30.5:600> ls -al
 total 1164
drwxrwxr-x  4 root root  4096 2009-08-20 12:24 ..
drwxrwxr-x 24 root root  4096 2009-08-19 14:24 arch/
drwxrwxr-x  2 root root  4096 2009-08-19 14:24 block/
drwxr-xr-x  2 root root  4096 2009-08-19 12:22 config/
-rw-r--r--  1 root root  94457 2009-08-20 12:28 .config
-rw-r--r--  1 root root  94457 2009-08-19 12:20 .config.old
-rw-rw-r--  1 root root  18693 2009-08-17 05:19 .gitignore
drwxrwxr-x  3 root root  4096 2009-08-19 14:24 crypto/
drwxrwxr-x  83 root root 12288 2009-08-17 05:19 Documentation/
drwxrwxr-x  84 root root  4096 2009-08-19 14:24 drivers/
drwxrwxr-x  33 root root  4096 2009-08-19 14:24 firmware/
drwxrwxr-x  70 root root  4096 2009-08-19 14:24 fs/
-rw-r--r--  1 root root   885 2009-08-17 05:19 .gitignore
drwxrwxr-x  22 root root  4096 2009-08-20 12:28 include/
drwxrwxr-x  2 root root  4096 2009-08-19 14:24 init/
drwxrwxr-x  2 root root  4096 2009-08-19 14:24 ipc/
-rw-r--r--  1 root root  2430 2009-08-17 05:19 Kbuild
drwxrwxr-x  6 root root  4096 2009-08-20 12:28 kernel/
drwxrwxr-x  6 root root  4096 2009-08-19 14:24 lib/
-rw-r--r--  1 root root  4021 2009-08-17 05:19 .mailmap
-rw-r--r--  1 root root 148271 2009-08-17 05:19 MAINTAINERS
-rw-r--r--  1 root root  55181 2009-08-17 05:19 Makefile
-rw-r--r--  1 root root   258 2009-08-20 12:28 .missing-syscalls.d
-drwxrwxr-x  2 root root  4096 2009-08-19 14:24 mm/
Point to note, if you built your kernel and you type make clean. You assume it just purges the build information but it does delete some header files which you need for your build.

What we are doing is using the “ndiswrapper” to basically use the windows distributed network driver on Linux system to get wireless working.
Instructions

First thing you need to do is download the ndiswrapper from sourceforge ie [http://sourceforge.net/apps/mediawiki/ndiswrapper](http://sourceforge.net/apps/mediawiki/ndiswrapper).

```
[root@CSSE2501 tmp]# pwd
/tmp
[root@CSSE2501 tmp]# tar xpfz ndiswrapper-1.55.tar.gz
[root@CSSE2501 ndiswrapper-1.55]# pwd
/tmp/ndiswrapper-1.55
[root@CSSE2501 ndiswrapper-1.55]# make
make -C driver
make[1]: Entering directory `/tmp/ndiswrapper-1.55/driver'
make -C /usr/local/src/linux-2.6.28 M=/tmp/ndiswrapper-1.55/driver
make[2]: Entering directory `/usr/local/src/linux-2.6.28'
LD /tmp/ndiswrapper-1.55/driver/built-in.o
MKEXPORT /tmp/ndiswrapper-1.55/driver/crt_exports.h
MKEXPORT /tmp/ndiswrapper-1.55/driver/hal_exports.h
MKEXPORT /tmp/ndiswrapper-1.55/driver/ndis_exports.h
MKEXPORT /tmp/ndiswrapper-1.55/driver/ntoskernel_exports.h
MKEXPORT /tmp/ndiswrapper-1.55/driver/ntoskernel_io_exports.h
MKEXPORT /tmp/ndiswrapper-1.55/driver/rtl_exports.h
MKEXPORT /tmp/ndiswrapper-1.55/driver/usb_exports.h
CC [M] /tmp/ndiswrapper-1.55/driver/crt.o
In file included from include/linux/gfp.h:4,
   from include/linux/kmod.h:22,
   from include/linux/module.h:13,
   from /tmp/ndiswrapper-1.55/driver/ntoskernel.h:22,
   from /tmp/ndiswrapper-1.55/driver/crt.c:16:
infile/linux/mmzone.h:18:26: error: linux/bounds.h: No such file or directory
include/linux/mmzone.h:256:5: warning: "MAX_NR_ZONES" is not defined
In file included from include/linux/gfp.h:4,
   from include/linux/kmod.h:22,
   from include/linux/module.h:13,
   from /tmp/ndiswrapper-1.55/driver/ntoskernel.h:22,
   from /tmp/ndiswrapper-1.55/driver/crt.c:16:
in file included from include/linux/mmzone.h:277: error: "MAX_NR_ZONES" undeclared here (not in a function)
In file included from include/linux/scatterlist.h:6,
   from /usr/local/src/linux-2.6.28/arch/x86/include/asm/dma-mapping.h:9,
   from include/linux/dma-mapping.h:57,
   from include/linux/dmaengine.h:29,
   from include/linux/skbuff.h:29,
   from include/linux/if_ether.h:120,
   from include/linux/netdevice.h:29,
   from /tmp/ndiswrapper-1.55/driver/ntoskernel.h:25,
   from /tmp/ndiswrapper-1.55/driver/crt.c:16:
in file included from include/linux/mm.h:438:63: warning: "NR_PAGE_FLAGS" is not defined
include/linux/mm.h:486:62: warning: "NR_PAGE_FLAGS" is not defined
make[3]: *** /tmp/ndiswrapper-1.55/driver/crt.o Error 1
make[2]: *** [module/_tmp/ndiswrapper-1.55/driver] Error 2
make[2]: Leaving directory `/usr/local/src/linux-2.6.28'
make[1]: *** [modules] Error 2
make[1]: Leaving directory `/tmp/ndiswrapper-1.55/driver'
make: *** [all] Error 2
```

Email: ashley@csse.uwa.edu.au  Last Update: 26/11/2009
Note when I compiled ndiswrapper 1.55 I got a hard error “NR_PAGEFLAGS” and basically does not compile. This is a result of my kernel source tree being incomplete as “NR_PAGEFLAGS” is defined in bounds.h which should have been in my case /usr/local/src/linux-2.6.30.5/include/linux/bounds.h

But as I mentioned earlier on, I had a make clean when I compiled and installed my kernel to save space but that also delete “bounds.h”. But I still had a copy of my entire kernel tree before I typed “make clean”. The alternative is recompile everything again with the same .config which should regenerate all the files.

So I place back entire kernel compiled tree and compile “ndiswrapper” again and you should get something like this when you type make.

```
[root@CSSE2124 ndiswrapper-1.55]# make
make -C driver
make[1]: Entering directory `/tmp/ndiswrapper-1.55/driver'
make -C /usr/local/src/linux-2.6.30.5 M=/tmp/ndiswrapper-1.55/driver
make[2]: Entering directory `/usr/local/src/linux-2.6.30.5'
LD /tmp/ndiswrapper-1.55/driver/built-in.o
MKEXPORT /tmp/ndiswrapper-1.55/driver/crt_exports.h
MKEXPORT /tmp/ndiswrapper-1.55/driver/hal_exports.h
MKEXPORT /tmp/ndiswrapper-1.55/driver/ndis_exports.h
MKEXPORT /tmp/ndiswrapper-1.55/driver/ntoskernel_exports.h
MKEXPORT /tmp/ndiswrapper-1.55/driver/ntoskernel_io_exports.h
MKEXPORT /tmp/ndiswrapper-1.55/driver/rtl_exports.h
MKEXPORT /tmp/ndiswrapper-1.55/driver/usb_exports.h
CC [M] /tmp/ndiswrapper-1.55/driver/crt.o
CC [M] /tmp/ndiswrapper-1.55/driver/hal.o
CC [M] /tmp/ndiswrapper-1.55/driver/iw_ndis.o
CC [M] /tmp/ndiswrapper-1.55/driver/loader.o
CC [M] /tmp/ndiswrapper-1.55/driver/ndis.o
CC [M] /tmp/ndiswrapper-1.55/driver/ntoskernel.o
CC [M] /tmp/ndiswrapper-1.55/driver/ntoskernel_io.o
CC [M] /tmp/ndiswrapper-1.55/driver/pe_linker.o
CC [M] /tmp/ndiswrapper-1.55/driver/pnp.o
CC [M] /tmp/ndiswrapper-1.55/driver/proc.o
CC [M] /tmp/ndiswrapper-1.55/driver/rtl.o
CC [M] /tmp/ndiswrapper-1.55/driver/rapmem.o
CC [M] /tmp/ndiswrapper-1.55/driver/rapndis.o
CC [M] /tmp/ndiswrapper-1.55/driver/wrap.o
CC [M] /tmp/ndiswrapper-1.55/driver/usb.o
CC [M] /tmp/ndiswrapper-1.55/driver/divdi3.o
LD [M] /tmp/ndiswrapper-1.55/driver/ndiswrapper.o
Building modules, stage 2.
MODPOST 1 modules
CC /tmp/ndiswrapper-1.55/driver/ndiswrapper.mod.o
LD [M] /tmp/ndiswrapper-1.55/driver/ndiswrapper.ko
make[2]: Leaving directory `/usr/local/src/linux-2.6.30.5'
make[1]: Leaving directory `/tmp/ndiswrapper-1.55/driver'
make -C utils
make[1]: Entering directory `/tmp/ndiswrapper-1.55/utils'
gcc -g -Wall -l../driver -o loadndisdriver loadndisdriver.c
make[1]: Leaving directory `/tmp/ndiswrapper-1.55/utils'
```
If it passes the “make” command then you just simply type “make install” for it to install into the right places which is always a good sign.

Now we have “ndiswrapper” installed, now we get the windows drivers and install it. So download BCM4328 Windows Packaged Drivers ie


It is a basically a self extracting zip file but you can use the “unzip” command. Unzip and traverse down to a subdirectory called “Driver” with a “bcmwl5.inf” file in it. To physically install the driver you simply type “ndiswrapper -i bcmwl5.inf” followed by “ndiswrapper –m” which wires an alias for the wireless device into the module configuration files.

If it installed correctly you should be able to probe it with ndiswrapper itself ie “ndiswrapper –l”

[root@CSSE2124 tmp]# ndiswrapper -l
bcmwl5 : driver installed
    device {14E4:4328} present (alternate driver: ssb)
[root@CSSE2124 tmp]#

You might want to reboot it, if it doesn’t come automatically on the reboot you should be able to “modprobe ndiswrapper” to load it because its map the Windows Driver for it.

You should be able to configure it with the network manager or something similar afterwards. But using a normal iwconfig or ifconfig I can see the devices
And that’s basically it summing up my Document for BCM4328, I think the most annoying part was compiling the ndiswrapper looking for some missing declared value which happened to be tied to “making clean” to save space on the kernel source tree compile.