

## Adicionando suporte a D-Link DWL-G650+ (todas elas) no FreeBSD.

Autoria de P. Tracanelli (FreeBSD Brasil)  
14/12/2008  
Última Atualização 14/12/2008

Ao longo desse artigo vamos adicionar suporte no FreeBSD a placa wireless 802.11g D-Link DWL-G650+, conhecida como DLink Airplus Extreme G+, que usam algum chipset da Texas Instrument. Vamos observar que é uma loteria descobrir que placa você tem em mãos.

O artigo serve também como dica para se pensar bem antes de adquirir uma placa DLink. Não que seja ruim, nem boa. É que você não tem a menor garantia do que está comprando.

Ao final do artigo teremos uma acx0 funcionando e sendo controlada.

### Adicionando suporte a D-Link DWL-G650+ (todas elas) no FreeBSD

Durante o procedimento de colocar o PC-BSD como sistema Unix padrão no laboratório de mobilidade de uma Universidade em Belo Horizonte, observei que o FreeBSD não suporta adequadamente algumas placas wireless DLink Airplus Extreme G(+). O curioso é que antes da aquisição desse hardware, foi feita uma análise de compatibilidade de hardware com um LiveCD da FreeBSD Brasil, e funcionava adequadamente.

O porém é que é DLink. Ahh, DLink! Adoro a DLink e essa mania de mudar completamente um hardware e continuar vendendo-o com mesmo nome, mesmo modelo, mesma aparência e até mesmo mesma caixa e mesma adesivagem. Enfim, tudo em prol do baixo custo (mesmo assim DLINK.BO não é um exemplo de ativo a se manter em carteira).

O fato é que você nunca sabe quando compra um equipamento DLink, o que está comprando. Já sugeri a clientes comprar algumas placas Wireless PCI, e ao comprar, o equipamento era completamente diferente do pretendido. Desde então digo: teste antes, compare número serial e lote de fabricação. Parece exagero? Não é.

Algumas vezes se damos sorte, existem algumas diferenças no modelo, o + pode fazer diferença. Por exemplo, DWL-530 é chipset Intersil Prism 2, enquanto DWL-530+ as vezes é Texas Instruments e outras vezes é Crystal Semiconductor. Bom, mas o fato é, quando foi feita análise de hardware para estabelecer como padrão a placa PCMCIA padrão IEEE 802.11g, foram compradas 32 placas das que foram homologadas: DLink Airplus Extreme G.

Excelente, porque DLink Airplus Extreme G ou apenas a DWL-G650 é uma excelente placa, de chipset Atheros 5212. Sem dúvida uma escolha mais que aprovada. Acontece que foi pro departamento de compras e a aquisição foi um mixto de DWL-G650 e DWL-G650+, também conhecida como DLink Airplus Extreme G+. Conclusão lógica do departamento de compras: mesmo preço e com um +, excelente, é melhor. Mais pelo mesmo! Aprovada a compra.

Ai vem a surpresa. A DWL-G650 não tem, rigorosamente, nada a ver com a DWL-G650+. Enquanto a primeira é Atheros 5212, a segunda é algum chipset do fabricante Texas Instruments. Confuso? Não se preocupe, nada que não possa piorar. A DWL-G650+ por sua vez pode ser reconhecida como uma das seguintes:

- Texas Instruments (TI) 802.11b+ 22Mbps Wireless Adapter
- TNETW1130(ACX111) 802.11b/g Wireless Cardbus/PCI Adapter
- Texas Instruments (TI) 802.11b+/g 54Mbps Wireless Adapter
- Texas Instruments (TI) 802.11b/g 54Mbps Wireless Adapter

Não é ótimo? Não, não é! As diferenças são a Revisão da placa. Revisão 4 ou 5 podem ter 2 modelos distintos de chipset cada, totalizando 4 combinações possíveis para exatamente o mesmo equipamento/modelo, sem nem contar o DWL-G650 (sem o +) que é Atheros.

Resultado, que driver controla? Para Windows, o mesmo driver! Por algum motivo os firmware binários no Windows, fornecidos pela DLink são capazes de reconhecer qual das placas é e controla-la. Na prática isso significa que a abordagem de usar o NDIS para usar o driver do Windows no FreeBSD se torna uma loteria: as vezes funciona, as vezes não. De fato o `if_ndis.ko` gerado controla 1 dos 4 chips, e não controla os outros. Ou seja, usar NDIS com os drivers fornecidos pelo fabricante, sem chance de sucesso certo, é outra loteria.

Mas tem algumas coisas curiosas que a DLink consegue. Por exemplo, temos aqui literalmente dezenas de clientes que adquirem a placa DWL-G520 para atuar como Ponto de Acesso Wireless. Excelente, chipset Atheros, sem dúvida para quem quer uma opção PCI de baixo custo e acessível no mercado brasileiro, sem problemas. Junto com as mini-PCI

CM9 e Engenius acredito que são os equipamentos mais populares para atuar como Access Point no Brasil. E a ANATEL homologa, então sem problema, certo? Não exatamente.

A DWL-G520 (e de fato algumas outras da DLink com chipset Atheros) usam um modelo do chip Atheros que atua em múltiplos domínios de regulação (Regulatory Domains). A ANATEL homologa o modelo DWL-G520 nacional, também chamado de DWL-G520 BR. Todavia, o Regulatory Domain dessa placa é FCC, sem problemas, mesmas características em essência do que a ANATEL quer pro Brasil. Mas com um simples upgrade de firmware podemos colocar em ETSI Europe (1-13), isso permite que a placa atue em 802.11b até o canal 13.

Curioso? Bom, com o firmware europeu o limite entre a órgão regulador europeu e o japonês fica restrito ao nível do sistema operacional, que obedece aquele Regulatory Domain por padrão. Na prática para alternar do padrão Europeu para o Japonês e poder usar do canal 1 ao 16 a mesma placa, é complexo: uma variável sysctl(8) no FreeBSD (Atheros HAL).

Resultado? Ou uma fiscalização da ANATEL usa um analisador de espectro para avaliar a propagação do sinal de RF da placa, ou temos em mãos uma placa DLink com selinho de homologação da ANATEL, operando até o canal 16. Se baixar o driver de Windows da DLink União Européia a diferença entre o padrão Europeu e Japonês é 1 valor em uma variável do Registro do Windows. Nada que um duplo-click em um arquivo .reg não resolva.

Obviamente ninguém vai fazer isso devido a ilegalidade, mas o fato é que outros fabricantes limitam a atualização de firmware para os seriais/modelos de hardware, evitando um firmware poder ser utilizado em um equipamento montado para outro Regulatory Domain. A maioria dos fabricantes faz isso, inclusive a própria Atheros e outros fabricantes populares de placas wireless. A DLink não!

Mas voltando ao objetivo do artigo, fiquei feliz ao descobrir que na Coleção de Ports do FreeBSD temos um módulo de kernel, o `/usr/ports/net/acx100:# cat /usr/ports/net/acx100/pkg-descr`  
Texas Instruments (TI) ACX100 IEEE 802.11 driver.

The acx driver supports ACX100 based Cardbus or PCI network adapters.

The following cards are known to work with the acx driver:

Card	Bus
Binatone WL-1000	CARDBUS
D-Link DWL-650+	CARDBUS
US Robotics USR2210	CARDBUS
US Robotics USR2216	PCI

Excelente, era o que eu precisava, certo? Apenas parcialmente. Primeiro, porque o módulo em questão não está na base do FreeBSD? Simples, ele usa um firmware proprietário da Dlink com termos de licença ainda mais assustadores que a Intel e seus firmwares de placa wireless. Resultado, temos que sempre fazer download do firmware no site da DLink.

De acordo com a listagem, DWL-G650+ parece ser controlado. Excelente, o driver controla as placas ACX100 da Texas Instruments, e esse é o chipset da DWL-G650+! Errado! Nem sempre! As vezes o chipset é o ACX111. E as vezes é um mixto do ACX100 com o ACX111, algo único e exclusivo da DLink. Nenhum outro fabricante cliente OEM da Texas Instruments tem a capacidade de usar uma variação que precisa do suporte mixto de ambos os chipsets. Só a DLink.

Texas Instruments acx100+acx111

Felizmente o mesmo autor (Darron Brod) do módulo utilizado no `net/acx100` fez também uma versão para o uso mixto do acx100 e acx111. Para isso basta baixa-lo: `# fetch -o /tmp/ http://dev.kewl.org/acx100+111/acx-6.1.tgz`

```
Depois descompacte e mande compilar:# cd /tmp/
# tar xzf acx-6.1.tgz
# cd acx-6.1/
# make
```

Você vai observar que o próprio Makefile tem instruções para baixar o firmware direto do site d D-Link. Vai desmontar o driver e usar partes deles para gerar três produtos finais, dois módulos com firmware (acx100fw.ko e acx111fw.ko) e o módulo da placa de rádio, `if_acx.ko`.

O make install irá instala-los em /boot/kernel/, que é completamente adequado para fontes de kernel parte da base, mas completamente inadequado quando obtido de terceiros, pois a recompilação do seu kernel não vai incluir esse módulo no diretório em questão. Então depois de instalar mova para o /boot/modules/:# make install

```
# mv -v /boot/kernel/*acx*.ko* /boot/modules/
/boot/kernel/acx100fw.ko -> /boot/modules/acx100fw.ko
/boot/kernel/acx111fw.ko -> /boot/modules/acx111fw.ko
/boot/kernel/if_acx.ko -> /boot/modules/if_acx.ko
/boot/kernel/if_acx.ko.symbols -> /boot/modules/if_acx.ko.symbols
```

No /boot/loader.conf adicione:acx100fw\_load="YES"  
acx111fw\_load="YES"  
if\_acx\_load="YES"

Para não precisar reiniciar, carregue na mão:

```
# kldload acx100fw
# kldload acx111fw
# kldload if_acx
```

Pronto, agora você terá a interface acx0 disponível. Use o pciconf -lv se quiser saber qual foi sua sorte, na loteria que é adquirir essas placas (um dmesg também vai apresentar a informação).

Não é possível configurar completamente com o ifconfig(8). Para configurar os recursos wireless da placa use o acxtool que foi instalado:# rehash  
# acxtool

Você verá o help:acx100/111:

```
-M str      Set operating mode to str (managed, ad-hoc, master)
-S str      Set the SSID to str
-C num      Set the Channel to num

-P on/off   Set WEP privacy mode on or off
-W num:str  Set WEP tx key num (0..3) to str

-T num      Set Tx BROADCAST/MULTICAST RATE to num (1, 2, 5.5, 6 etc)
-t num      Set Tx MANAGEMENT RATE to num

-r num      Set RATE FALLBACK to num
-s num      Set SHORT RETRY LIMIT to num
-l num      Set LONG RETRY LIMIT to num (unsupported)
-u on/off   Set RATE UP to on (FAST) or off (SLOW)
-m num      Set MAX TX LIFETIME to num
-p num      Set POWER LEVEL to num
-a num      Set CURRENT ANTENNA to num (ant1:0x080a ant2:0x044a ?)
```

```
acx100:
-c num      Set CLEAR CHANNEL ASSESSMENT MODE to num
-e num      Set ENERGY DETECT THRESHOLD to num
```

```
acx111:
-f num      Set FRAG THRESHOLD to num
```

debug args (one per option):

```
-d on/off   Turn on all debugging options
-d rateon/rateoff Turn Tx RATE REPORTING on or off
-d erron/erroff Turn Tx ERROR REPORTING on or off
-d nodeon/nodeoff Turn NODE REPORTING on or off
-d num      Set MGMT REPORTING mask to num (0=off, eg 0xfeef)
```

Eg: acxtool acx0 -M managed -S kewl5 -C 5 -P on -W 0:0xc0dedeadc0dedeadc0dedead69

```

set the ssid to kewl5
set the channel to 5
set the WEP privacy mode on
set the WEP tx key index to 0
and set the WEP tx key to 0xc0dedeadc0dedeadc0dedead69

```

Em negrito as opções que provavelmente você vai usar com mais frequência. Configure então sua placa, adicionando o endereço IP com `ifconfig(8)` e configurando a placa com `acxtool`:

```
# ifconfig acx0 inet 10.69.70.74 netmask 255.255.255.0
```

E então o `acxtool` com pelo menos os seguintes parâmetros:

```

acxtool acx0 -S <ssid> -C <canal> -M managed -u on -d rateon# acxtool acx0 -S freebsdap -C 1 -M managed -u on -d
rateonCom ifconfig(8) você observará:
acx0: flags=843<UP,BROADCAST,RUNNING,SIMPLEX> mtu 1500
inet 10.69.70.74 netmask 0xfffff00 broadcast 10.69.70.255
ether 00:c0:49:ca:3f:b3
media: IEEE 802.11 Wireless Ethernet autoselect
status: associated
ssid freebsdap channel 1 (0) bssid 00:c0:49:c3:6c:f9
authmode OPEN privacy ON deftxkey 1 txpowmax 63 txpower 63

```

Como não é possível configurar tudo usando o `ifconfig(8)`, não será possível tornar a configuração permanente com `rc.conf(5)`, então use um recurso (pouco conhecido, mas documentado) do startup do FreeBSD e crie o `/etc/start_if.acx0`:

```

# cat /etc/start_if.acx0
/usr/local/sbin/acxtool acx0 -S freebsdap
/usr/local/sbin/acxtool acx0 -M managed
/usr/local/sbin/acxtool acx0 -C 1
/usr/local/sbin/acxtool acx0 -u on -d rateon
/sbin/ifconfig acx0 10.69.70.74/24
# /sbin/dhclient acx0
/sbin/route flush
/sbin/route add default 10.69.70.1

```

Pronto. A placa funciona com WEP, e com WPA só com PSK de até 64bits na senha (8 bytes), sem EAP e sem PSK maior que 8 bytes. Ainda assim apenas em hexadecimal (nesse caso use o `wpa_passhourse(8)` para converter para hexa).

Espero que esse artigo seja útil pra poucas pessoas ;-). E aos que forem, tentem trocar de placa o quanto antes. A performance dessa placa é baixa e tem facilidade para erros de transmissão.