NAME

    mwl - Marvell 88W8363 IEEE 802.11n wireless network driver

SYNOPSIS

    To compile this driver into the kernel, place the following lines in your kernel configuration file:

        device mwl
        device mwlfw
        device wlan
        device firmware

    Alternatively, to load the driver as a module at boot time, place the following line in loader.conf(5):

        if_mwl_load="YES"

DESCRIPTION

    The **mwl** driver provides support for IEEE 802.11n wireless network adapters based on Marvell 88W8363 parts. PCI and/or CardBus interfaces are supported.

    This driver requires the firmware built with the **mwlfw** module to work. Normally this module is loaded on demand by the driver but it may also be compiled into the kernel.

    Supported features include 802.11n, power management, BSS, MBSS, and host-based access point operation modes. All host/device interaction is via DMA.

    The **mwl** driver encapsulates IP and ARP traffic as 802.11 frames, however it can receive either 802.11 or 802.3 frames. Devices support 802.11n, 802.11a, 802.11g, and 802.11b operation with transmit speeds appropriate to each. The actual transmit speed used is dependent on signal quality and the "rate control" algorithm implemented in the firmware. All chips have hardware support for WEP, AES-CCM, TKIP, and Michael cryptographic operations.

    The driver supports **station**, **hostap**, **mesh**, and **wds** mode operation. Multiple **hostap** virtual interfaces may be configured for simultaneous use. When multiple interfaces are configured each may have a separate mac address that is formed by setting the U/L bits in the mac address assigned to the underlying device. Any number of **wds** virtual interfaces may be configured together with **hostap** interfaces. Multiple **station** interfaces may be operated together with **hostap** interfaces to construct a wireless repeater device. For more information on configuring this device, see ifconfig(8).

    Devices supported by the **mwl** driver come in either Cardbus or mini-PCI packages. Wireless cards in Cardbus slots may be inserted and ejected on the fly.

EXAMPLES

Join an existing BSS network (ie: connect to an access point):

```
ifconfig wlan create wlandev mwl0 inet 192.168.0.20 \
        netmask 0xffffff00"
```

Join a specific BSS network with network name "my_net":

```
ifconfig wlan create wlandev mwl0 inet 192.168.0.20 \
        netmask 0xffffff00 ssid my_net"
```

Join a specific BSS network with WEP encryption:

```
ifconfig wlan0 create wlandev mwl0
ifconfig wlan0 inet 192.168.0.20 netmask 0xffffff00 ssid my_net \
        wepmode on wepkey 0x8736639624
```

Create an 802.11g host-based access point:

```
ifconfig wlan0 create wlandev mwl0 wlanmode hostap
ifconfig wlan0 inet 192.168.0.10 netmask 0xffffff00 ssid my_ap \
        mode 11g
```

Create an 802.11a mesh station:

```
ifconfig wlan0 create wlandev mwl0 wlanmode mesh
ifconfig wlan0 meshid my_mesh mode 11a inet 192.168.0.10/24
```

Create two virtual 802.11a host-based access points, one with WEP enabled and one with no security, and bridge them to the fxp0 (wired) device:

```
ifconfig wlan0 create wlandev mwl0 wlanmode hostap \
        ssid paying-customers wepmode on wepkey 0x1234567890 \
        mode 11a up
ifconfig wlan1 create wlandev mwl0 wlanmode hostap bssid \
        ssid freeloaders up
ifconfig bridge0 create addm wlan0 addm wlan1 addm fxp0 up
```

DIAGNOSTICS

   mwl%d: unable to setup builtin firmware There was a problem downloading and/or setting up the

firmware. The device is not usable.

mwl%d: failed to setup descriptors: %d There was a problem setting up the DMA data structures. This typically is caused by not being able to allocate contiguous memory.

mwl%d: transmit timeout A frame dispatched to the hardware for transmission did not complete in time. This should not happen.

mwl%d: device not present A cardbus device was ejected while active; the request to the firmware was not completed.

SEE ALSO

cardbus(4), intro(4), mwlfw(4), pci(4), wlan(4), wlan_ccmp(4), wlan_tkip(4), wlan_wep(4), wlan_xauth(4), hostapd(8), ifconfig(8), wpa_supplicant(8)

HISTORY

The **mwl** device driver first appeared in FreeBSD 8.0.

BUGS

The driver does not support power-save operation in station mode; consequently power use is suboptimal (e.g. on a laptop).