

Re: Trouble loading if_ndis with the Win2k driver (XP one works)

Source: <http://unix.derkeiler.com/Mailing-Lists/FreeBSD/current/2004-06/0689.html>

From: Bill Paul (wpaul_at_FreeBSD.ORG)

Date: 06/10/04

To: doconnor@gsoft.com.au (Daniel O'Connor)

Date: Thu, 10 Jun 2004 16:08:21 +0000 (GMT)

> -----BEGIN PGP SIGNED MESSAGE-----
> Hash: SHA1
>
> *I have a -current from 2004.05.27.14.30.00 (because of this ACPI stuff*
> *breaking by bfe card :) but I can't load if_ndis anymore..*

Ok, listen to me: the way you've phrased this implies that you think you've discovered a new problem that wasn't there in the past. But you're wrong: you were *_never_* able to use the W70N5.INF file to generate ndis_driver_data.h. All along, you were using W70N51.INF, until one day you accidentally left off the '1' at the end, and were surprised when it didn't work. You then became confused and jumped to the incorrect conclusion that you had encountered a new problem that hadn't been there before.

You're *_SUPPOSED_* to use the Windowx XP version. The NDISulator pretends to be Windows XP, and the ndiscvt utility pretends to parse .INF files the same way Windows XP would. If you'd taken the time to actually read the W70N5.INF file, you would have seen the following lines:

```
[Manufacturer]  
%COMPANY_NAME% = Device,NT.5.1
```

```
[Device.NT.5.1]  
; Used to Prevent XP Installations using this INF
```

Intel has tried to be clever and rigged the .INF files such that if you try to install the NDIS 5.0 driver on XP, the installer will be redirected to a dummy device list section which has no entries in it. This will prevent the installer from matching the driver to any devices when run on XP, which appears to be what Intel wants. Conversely, the W70N51.INF file for Windows XP has the opposite logic in it to prevent that driver from being installed on Windows 2000 or NT.

freebsd-current: Re: Trouble loading if_ndis with the Win2k driver (XP one works)

The ndiscvt(8) utility is following the instructions in the W70N5.INF and not generating a device list section, which means the resulting ndis_driver_data.h file won't have an NDIS_PCI_DEV_TABLE in it. The if_ndis_pci.o module only has any brains compiled into it if NDIS_PCI_DEV_TABLE is #defined. Part of those brains includes the MODULE_DEPEND() macros that tell the compiler that the resulting if_ndis.ko module has a linker dependency on ndis.ko (which is how the runtime linker knows to try and resolve external symbol references in if_ndis.ko against ndis.ko.) No linker dependency info in the module means the linker can't resolve things like ndis_get_info(), which lives in ndis.ko, is why you get the error that you do.

There is no bug here. The Windows XP version of the driver clearly works. That's what you're supposed to be using. Running the Win2K version of the driver isn't going to gain you anything anyway. If you really wanted to use the Win2K driver object, you could do it by editing the ndis_driver_data.h file (or the W70N5.INF file) but there's really no point. These aren't the droids you're looking for. I can go about my business. Ok?

-Bill

--

```
=====
-Bill Paul          (510) 749-2329 | Senior Engineer, Master of Unix-Fu
                    wpaul@windriver.com | Wind River Systems
=====
<adamw> you're just BEGGING to face the moose
=====
```

freebsd-current@freebsd.org mailing list
<http://lists.freebsd.org/mailman/listinfo/freebsd-current>
To unsubscribe, send any mail to "freebsd-current-unsubscribe@freebsd.org"