

Erledigt

Intel X550-AT2 10Gbit LAN

Beitrag von „DSM2“ vom 18. November 2018, 09:29

Hallo allerseits!

Eigentlich wollte ich das schon vor Monaten machen aber leider hatte ich einfach nicht die Zeit dazu.

Mein Asus WS X299 Sage/10G hat ja bekanntlich 2x 10 Gbit Lan Ports Onboard, dabei handelt es sich um die Intel X550-AT2.

Der Ursprüngliche Plan war anfangs, dass ich das Lan per Geräte ID oder aber FakeID an den SmallTree anpasse und damit dann befeuer,

was an sich zwar auch klappt aber alles andere als zuverlässig funktionieren tut per Kext only, den manchmal bindet er die Intel X550 korrekt ein und manchmal nicht.

Das ist und war selbstverständlich keine Option aber ich lies es zu beginn stehen, da ich ja noch meine Original Aquantia Karte hatte.

Als nun der MacPro 5.1 ins Haus flatterte, war klar das die Aquantia ihren Einzug in diesen finden wird, den der X299 soll natürlich mit dem MacPro kommunizieren. 😊

Das ganze habe ich nun heute Nacht umgesetzt und für Leute die vielleicht ebenfalls ein Asus WS X299 Sage/10G ihr eigen nennen festgehalten.

Da der Wiki/Anleitungsbereich aktuell gesperrt ist, zwecks Überarbeitung, packe ich das ganze nun hier rein.

Was wird benötigt:

- 1) USB Stick
- 2) Windows für das erstellen des Ubuntu USB Sticks via Rufus
- 3) Rufus - https://rufus.ie/en_IE.html

4) Ubuntu Image - <https://www.ubuntu.com/download/desktop>

5) SmallTree Treiber sobald umgesetzt : https://www.small-tree.com/sup...ownload_category?cat_id=6

How to:

- 1) Per Rufus einen USB Stick erstellen
- 2) Per F8 vom Stick starten und Ubuntu mit ""Try Ubuntu without Installing" booten.
- 3) Mit dem Internet verbinden
- 4) Terminal öffnen

und dann kann der Spaß beginnen :

Code

1. 1) sudo apt install net-tools
- 2.
3. 2) sudo apt-get install ethtool
- 4.
5. 3) ifconfig

Nachdem letzten Command habt ihr dann das vor Augen:

```
unpacking ethtool (1:4.15-0ubuntu1) ...
Processing triggers for man-db (2.8.3-2) ...
Setting up ethtool (1:4.15-0ubuntu1) ...
ubuntu@ubuntu:~$ ifconfig
enp225s0f0: flags=4095<UP,BROADCAST,MULTICAST> mtu 1500
    ether 18:31:bfcf:7c:1d:26 txqueuelen 1000 (Ethernet)
    RX packets 0 bytes 0 (0.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 0 bytes 0 (0.0 B)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

enp225s0f1: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    ether 18:31:bfcf:7c:1d:26 txqueuelen 1000 (Ethernet)
    RX packets 0 bytes 0 (0.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 123 bytes 19725 (19.7 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid host<host>
    loop txqueuelen 1000 (Local Loopback)
    RX packets 496 bytes 33569 (33.5 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 496 bytes 33569 (33.5 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

wlp9958: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.182.27 netmask 255.255.0.0 broadcast 192.168.182.255
    inet6 fe80::46d9:65b8:a88a:1508 prefixlen 64 scopeid host<link>
    ether f4:1c:8b:43:e9:19 txqueuelen 1000 (Ethernet)
    RX packets 308 bytes 18341 (18.3 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 308 bytes 33716 (33.7 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

ubuntu@ubuntu:~$
```

Meine Intel X550-AT2 ist enp225s0f0 und enp225s0f1 zugewiesen, da die Intel X550-AT2 eine Dual 10 Gbit Lösung ist. Die enp Adressen in meinem Fall können von euren abweichen!!!

Das ganze hängt von der verwendeten Linux Version ab, also bitte drauf achten, das ihr in den befehlen eure enp korrekt hinterlegt, falls diese von meinen abweichen sollten.

Der Rest der Befehle bleibt gleich!

Machen wir weiter.

Diese Werte sind für uns wichtig...

Code

1. 4) lspci -nn -vvv | grep Ethernet

```

RX errors 0 dropped 0 overruns 0 frame 0
TX packets 498 bytes 33569 (33.5 KB)
TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

wlp195s0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
inet 192.168.192.27 netmask 255.255.255.0 broadcast 192.168.192.255
inet6 fe80::45d9:45d8:a0a8:115d prefixlen 64 scopeid 0x20::1::ab
ether f419c188:a5:c9:1d typegenetlan 1000 (Ethernet)
RX packets 338 bytes 283812 (283.8 KB)
RX errors 0 dropped 0 overruns 0 frame 0
TX packets 339 bytes 33716 (33.7 KB)
TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

ubuntu@ubuntu:~$ ipnetool -m -vvv | grep ethernet
*1.00.0 Ethernet Controller [8086]: Intel Corporation Ethernet Controller 100.020F (9000:1000) (rev 02)
Subsystem: ASUSTeK Computer Inc. Ethernet Controller 100.020F (1000:1000)
*1.00.1 Ethernet Controller [8086]: Intel Corporation Ethernet Controller 100.020F (9000:1000) (rev 02)
Subsystem: ASUSTeK Computer Inc. Ethernet Controller 100.020F (1000:1000)
ubuntu@ubuntu:~$

```

Der Relevante Wert lautet 8712 da dieser zu 000a verändert werden muss auf beiden Ports.

Code

1. 5) sudo ethtool -e enp225s0f0 | less

Es erscheinen eine Menge an Offset Werten aber uns interessiert nur einer: und das ist die Subsystem ID ...

43 10 = 1043 = Subsystem-Hersteller-ID - 12 87 = 8712 = Subsystem-ID !!! Da bist du ja 😊

```

0x0170: ff ff
0x0180: ff ff
0x0190: ff ff
0x01a0: ff ff
0x01b0: ff ff
0x01c0: ff ff
0x01d0: ff ff
0x01e0: ff ff
0x01f0: ff ff
0x0200: 07 00 18 31 bf cf 7c 2d 40 41 84 45 00 00 01 00
0x0210: 07 00 18 31 bf cf 7c 2e 40 41 84 45 00 00 01 00
0x0220: 7c 40 00 40 49 19 00 00 00 00 c9 00 00 00 00 00
0x0230: 00 00 00 00 fd 00 1f c8 01 00 00 00 00 00 00 00
0x0240: 43 10 12 87 00 00 00 00 00 00 00 00 00 02 00 00 00
0x0250: 00 18 00 00 83 0e 00 00 97 73 00 00 04 10 00 00
0x0260: 86 80 00 00 cb 3f 00 00 63 15 63 15 00 00 00 00
0x0270: 65 15 65 15 00 00 00 00 63 15 63 15 20 00 00 00
0x0280: 65 15 65 15 00 00 00 18 31 bf cf 7c 2d 18 31
0x0290: bf cf 7c 2e 08 00 04 00 ff ff ff ff ff ff ff ff
0x02a0: ff ff ff ff 03 00 ff ff ff ff ff ff ff ff ff ff
0x02b0: ff ff ff ff ff ff 69 53 e0 05 01 00 00 00 00 00
0x02c0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x02d0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x02e0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x02f0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

```

Folgendes müsst ihr nun ausführen:

Code

1. sudo ethtool -E enp225s0f0 magic 0x15638086 offset 0x242 value 0x0a
2. sudo ethtool -E enp225s0f0 magic 0x15638086 offset 0x243 value 0x00

3. `sudo ethtool -E enp225s0f1 magic 0x15638086 offset 0x242 value 0x0a`
4. `sudo ethtool -E enp225s0f1 magic 0x15638086 offset 0x243 value 0x00`

Danach Neustarten und MacOS booten, offiziellen SmallTree 10 Gbit Treiber installieren und sicher in die Zukunft gehen.

Vor dem Patchen:

```
ethernet:  
  
Typ: Ethernet-Controller  
Treiber installiert: Nein  
MSI: Nein  
Bus: PCI  
Steckplatz: AirPort@225,0,0  
Hersteller-ID: 0x8086  
Geräte-ID: 0x1563  
Subsystem-Hersteller-ID: 0x1043  
Subsystem-ID: 0x8712  
Versions-ID: 0x0001  
Link-Breite: x4  
Link-Geschwindigkeit: 8.0 GT/s
```

Danach:

```
ethernet:  
  
Typ: Ethernet-Controller  
Treiber installiert: Ja  
MSI: Ja  
Bus: PCI  
Steckplatz: AirPort@225,0,0  
Hersteller-ID: 0x8086  
Geräte-ID: 0x1563  
Subsystem-Hersteller-ID: 0x1043  
Subsystem-ID: 0x000a  
Versions-ID: 0x0001  
Link-Breite: x4  
Link-Geschwindigkeit: 8.0 GT/s
```

PS: Ich bin nicht der Erfinder dieser Methode oder habe es rausgefunden...

Ich habe es lediglich für meine Intel Karte angepasst bzw umgesetzt und für andere festgehalten,

die das vielleicht wiederholen wollen oder müssen.

All credits goes to : [Squiid](https://forums.macrumors.com/t...ee-macos-drivers.1968456/) von MacRumors - <https://forums.macrumors.com/t...ee-macos-drivers.1968456/>