

# Schnittstellen

ITT-Netzwerk

Sebastian Meisel

12. Dezember 2022

## 1 Physische Netzwerkschnittstellen

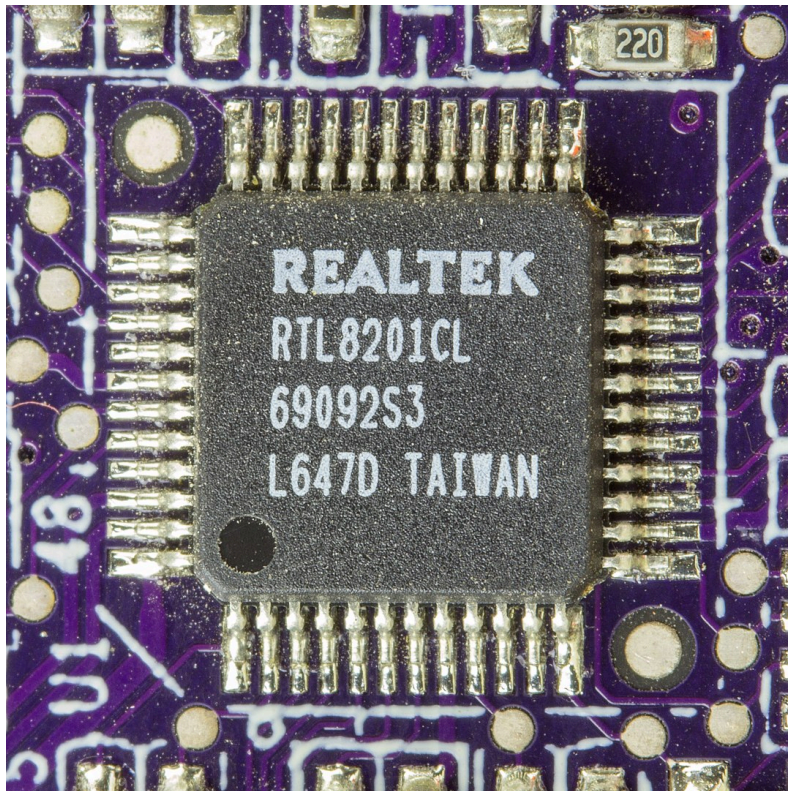


Abbildung 1: Von © Raimond Spekking / CC BY-SA 4.0

**Physikalische Netzwerkschnittstellen** sind Hardwarekomponenten, die

- Verbindung zu einem *Übertragungsmedium* herstellen, in dem sie
  - einen Anschluss (Buchse) für ein *Kabel* mit einem passenden Stecker bereitstellen.
  - \* **oder** einen Sender (Antenne / Emitter) und einen Empfänger (Antenne / Sensor) für ein kabelloses Signal.
  - einen Chip (*PHY*) bereitstellen, der

- \* eingehende Signale entschlüsselt.
- \* ausgehende Signale verschlüsselt.

In beiden Fällen ist schlicht die Übersetzung aus dem, bzw. in das *Netzwerksignal* gemeint, keine Verschlüsselung im Sinne, dass die Daten geschützt werden.

Die Netzwerkschnittstellen können jeweils fest auf dem *Motherboard* verlötet sein, wobei der *PHY* auch im Sinne eines *Systems on a Chip (SoC)* im selben Chip, wie die CPU integriert sein kann.

Häufiger sind *Netzwerkschnittstellen* aber als PCI(e)-Karten umgesetzt. Zudem gibt es Adapter über *USB*, *Thunderbolt* oder *Lightning* (früher auch über *PCMCIA*).

Wie man physische *Netzwerkschnittstellen* unter Windows findet haben wir schon bei den Grundbegriffen gesehen. Unter Linux gibt es verschiedene Möglichkeiten. Am komfortabelsten geht es mit dem Befehl:

```
lshw -c net
```

der allerdings zumeist nachinstalliert werden muss – z. B. mit `sudo apt install lshw` unter debian-basierten Distributionen wie Ubuntu oder Kali-Linux.

```
*-network
```

```
  Beschreibung: Kabellose Verbindung
```

```
  Produkt: Wireless-AC 9260
```

```
  Hersteller: Intel Corporation
```

```
  Physische ID: 0
```

```
  Bus-Informationen: pci@0000:01:00.0
```

```
  Logischer Name: wlp1s0
```

```
  Version: 29
```

```
  Seriennummer: 18:56:80:e5:88:5f
```

```
  Breite: 64 bits
```

```
  Takt: 33MHz
```

```
  Fähigkeiten: pm msi pciexpress msix bus_master cap_list ethernet physical wireless
```

```
  Konfiguration: broadcast=yes driver=iwlwifi driverversion=5.15.0-10053-tuxedo
firmware=46.fae53a8b.0 9260-th-b0-jf-b0- ip=192.168.0.10 latency=0 link=yes multicast=yes
wireless=IEEE 802.11
```

```
  Ressourcen: irq:19 memory:dfa00000-dfa03fff
```

```
*-network
```

```
  Beschreibung: Ethernet interface
```

```
  Produkt: RTL8111/8168/8411 PCI Express Gigabit Ethernet Controller
```

```
  Hersteller: Realtek Semiconductor Co., Ltd.
```

```
  Physische ID: 0.1
```

```
  Bus-Informationen: pci@0000:02:00.1
```

```
  Logischer Name: ens1f1
```

```
  Version: 12
```

```
Seriennummer: 80:fa:5b:5c:df:4d
Größe: 1Gbit/s
Kapazität: 1Gbit/s
Breite: 64 bits
Takt: 33MHz
Fähigkeiten: pm msi pciexpress msix vpd bus_master cap_list ethernet physical
tp mii 10bt 10bt-fd 100bt 100bt-fd 1000bt-fd autonegotiation
Konfiguration: autonegotiation=on broadcast=yes driver=r8169
driverversion=5.15.0-10053-tuxedo duplex=full firmware=rtl8411-2_0.0.1 07/08/13
ip=192.168.24.1 latency=0 link=yes multicast=yes port=twisted pa ir speed=1Gbit/s
Ressourcen: irq:16 ioport:e000(Größe=256) memory:df014000-df014fff m
emory:df010000-df013fff
```

## 1.1 Ethernet

### 1.1.1 PCI



Abbildung 2: By Dmitry Nosachev - Own work, CC BY-SA 4.0

Bereits als vor allem noch *Thin Ethernet* über Koaxialkabel verwendet wurde, gab es dafür PCI-Karten.

Es gibt PCIe Karten mit einem, aber auch mit bis zu 8 Ethernetports. Man muss zudem beachten, welche Bandbreite die Karte unterstützt.

Außerdem muss man bei CAT 7a-Kabeln und CAT 8.2-Kabeln beachten, dass die Karten den passenden Port (TERA, ARJ45, OG45) bietet.

### 1.1.2 USB-Adapter

Adapter ermöglichen es Ethernet an Geräten zu nutzen, die keinen Ethernetport haben und keine Erweiterung über PCIe bieten - vor allem Notebooks. Es ist damit auch auf einfache Weise möglich

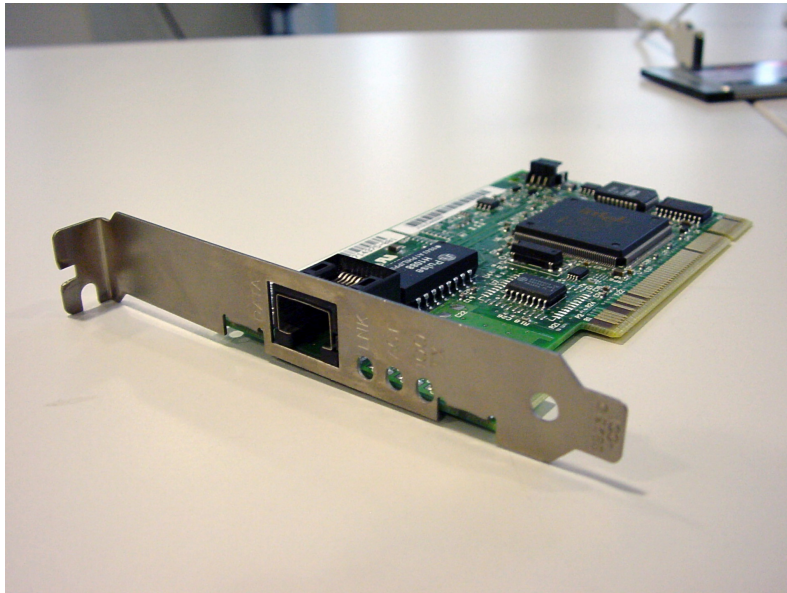


Abbildung 3: (C) Christian Hart, Public Domain



Abbildung 4: Von © Raimond Spekking / CC BY-SA 4.0



eine schnellere Schnittstelle zur Verfügung zu stellen, als das Gerät es ursprünglich bietet (10 statt 1 Gbs).

## 1.2 WLAN

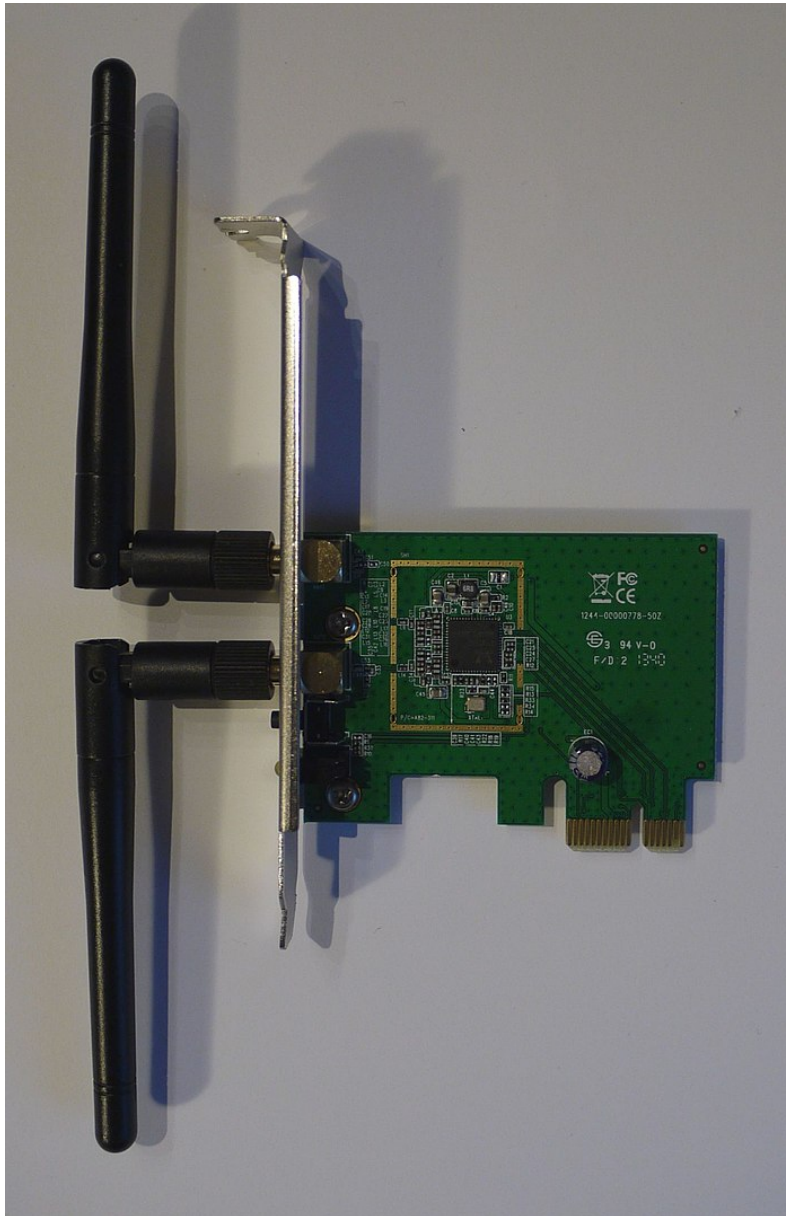


Abbildung 5: By Markus Säynevirta, CC BY-SA 4.0

Viele Motherboards bieten inzwischen WLAN *onBoard*, also mit einem festverlöteten *PHYs* oder in der CPU integriert. Allerdings nutzen diese oft sehr kleine Antennen. Eine dezidierte PCIe-Karte kann höhere Reichweiten ermöglichen.

Das WLAN-Modul kann auch über Mini-PCIe eingebunden werden.



Abbildung 6: By Sadenäyttely - Own work, CC BY-SA 4.0

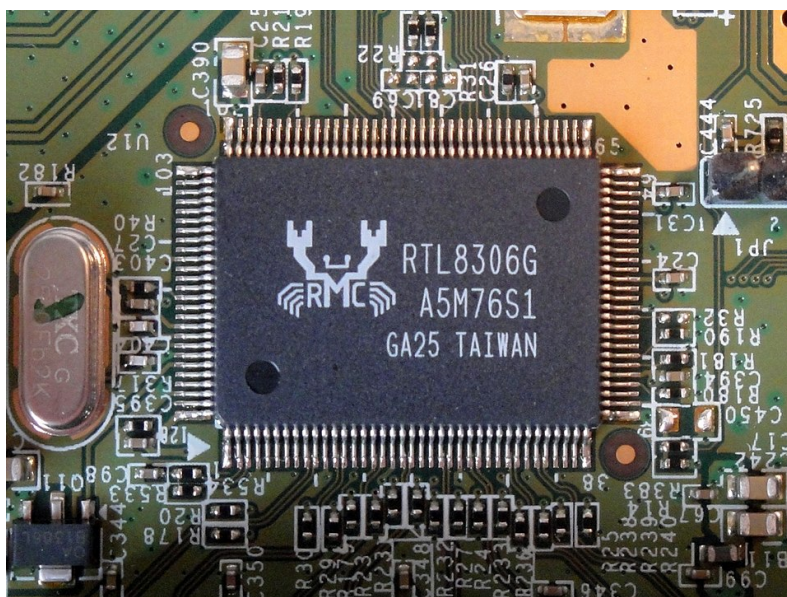


Abbildung 7: By Reise Reise - Own work, CC BY-SA 4.0

### 1.3 Ethernet over USB

Eine Alternative zum Ethernet über CAT-Kabel ist das Ethernet over USB Protokoll, das es erlaubt eine Ethernetverbindung über ein USB-Kabel aufzubauen, was bei den hohen Geschwindigkeiten von USB 4 (bis 40 Gbs) vor allem für Peer-to-Peer-Verbindungen über kurze Entfernungen eine Alternative ist.

Diese Funktionalität ist zum Beispiel in den Linuxkernel eingebaut.

## 2 Virtuelle Netzwerkschnittstellen

Neben den *Physischen* gibt es auch verschiedene Arten von **Virtuellen Netzwerkschnittstellen**. Sie werden auf Betriebssystemebene vom *Kernel* bereitgestellt.

```
ip link show
```

```
1: lo: <LOOPBACK,UP,LOWER_UP> ~~~
2: ens1f1: <BROADCAST,MULTICAST,UP,LOWER_UP> ~~~
3: wlp1s0: <BROADCAST,MULTICAST,UP,LOWER_UP> ~~~
4: virbr0: <NO-CARRIER,BROADCAST,MULTICAST,UP> ~~~
5: cni-podman0: <BROADCAST,MULTICAST,UP,LOWER_UP> ~~~
6: veth171e7929@if3: <BROADCAST,MULTICAST,UP,LOWER_UP> ~~~
7: docker0: <NO-CARRIER,BROADCAST,MULTICAST,UP> ~~~
8: br-f80eeef712c7: <BROADCAST,MULTICAST,UP,LOWER_UP> ~~~
10: veth3e38908@if9: <BROADCAST,MULTICAST,UP,LOWER_UP> ~~~
12: veth0f30863@if11: <BROADCAST,MULTICAST,UP,LOWER_UP> ~~~
20: tun0: <POINTOPOINT,MULTICAST,NOARP,UP,LOWER_UP> ~~~
```

### 2.1 Loopbackdevices

**Loopbackdevices** dienen dazu Netzwerkdienste, wie Web- oder E-Mailserver, die auf einer Netzwerkschnittstelle lauschen, lokal zu nutzen.

Anders als Windows stellt Linux immer ein Loopbackdevice mit der Bezeichnung *lo* zur Verfügung, über das der *Localhost*, also der eigene Rechner unter der IP *127.0.0.1* (IPv4) bzw. *::1* (IPv6) erreichbar ist.

### 2.2 Tunnelschnittstellen

Verschlüsselte Netzwerkverbindungen, sogenannte *Netzwerk-tunnel* können auf der *Anwendungsebene* oder über eine *virtuelle Netzwerkschnittstelle* bereitgestellt werden. Letzteres ermöglicht es z. B. ein VPN mit jeder beliebigen Clientsoftware zu nutzen.

Unter Linux werden *Netzwerkschnittstellen* zu *Netzwerk-tunneln* mit *tun* gekennzeichnet.

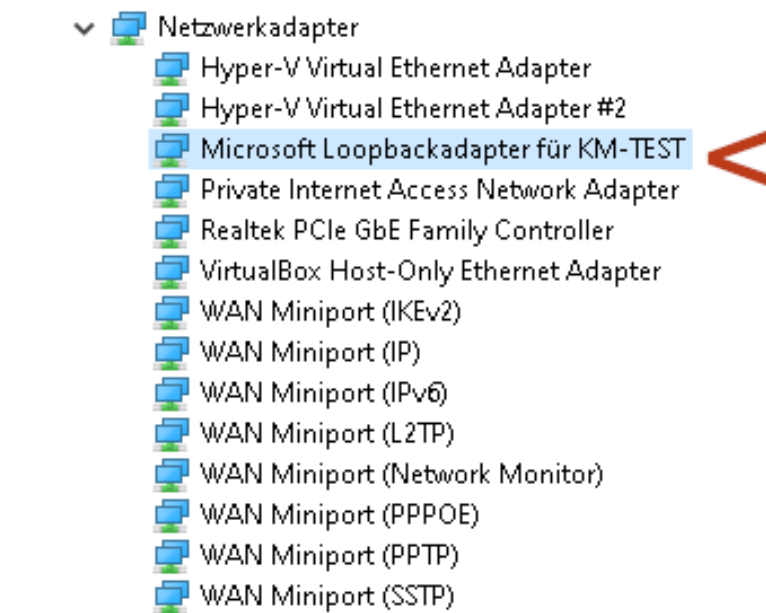


Abbildung 8: Loopbackdevice im Windowsgerätemanager

### 2.2.1 VPN

*Virtual Private Networks (VPNs)* werden noch ausführlich ein Thema sein. Zur Nutzung wird eine *virtuelle Netzwerkschnittstelle* eingerichtet über die man sich mit dem entfernten Rechner oder Netzwerk verbindet.

### 2.2.2 SSH-Tunnel

Es ist auch möglich über *SSH Netzwerktunnel* über eine *virtuelle Netzwerkschnittstelle* herzustellen. Das ist allerdings keine sehr zuverlässige Methode. Häufiger ein *Tunnel* auf der *Anwendungsebene* hergestellt.

So kann man mit dem Befehl:

```
ssh -N -L 2810:localhost:8000 user@entfer.nt
```

einen Dienst, der auf dem Rechner *entfer.nt* als lokaler Dienst nur über den *localhost* erreichbar ist, über einen Tunnel auf meinem Rechner an Port 2810 verfügbar machen.

Umgekehrt kann ich mit

```
ssh -N -R 2810:localhost:8000 user@entfer.nt
```

einen Dienst der auf meinem Rechner über *localhost:2810* erreichbar ist auf dem Rechner *entfer.nt* verfügbar machen.



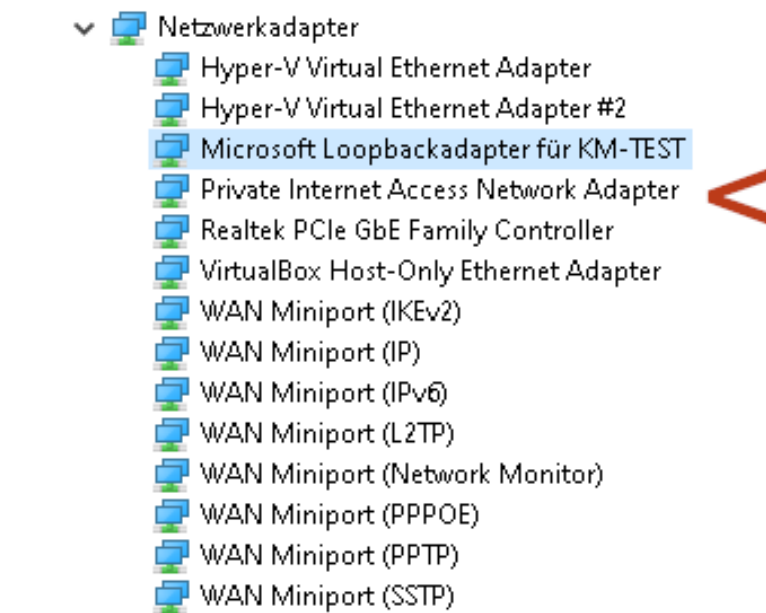


Abbildung 9: VPN-Tunnel-Adapter im Windowsgerätemanager

### 2.2.3 Fernwartung

Auch Fernwartungssoftware erstellt zum Teil *virtuelle Schnittstellen*. Teilweise werden aber auch *VPN-Verbindungen* oder ein *Tunnel* auf der *Anwendungsebene* genutzt.

## 2.3 Virtuelle Schnittstellen für Virtuelle Maschinen und Container

Damit *virtuelle Maschinen* mit einem Netzwerk kommunizieren können brauchen sie ebenfalls eine *virtuelle Netzwerkschnittstelle*.

Hierfür wird häufig eine *virtuelle Bridge* genutzt. Diese ermöglicht es zwei physisch getrennte Netzwerke zu einem zu verbinden. Zwei *Netzwerkschnittstellen* an einem Gerät sind normalerweise zwei getrennten Netzwerken zugeordnet. Sie können nicht *direkt* mit einander kommunizieren. Eine *Bridge* überbrückt diese Trennung der Netzwerke und macht zu einem.

## 2.4 WAN Miniports (Windows)

*WAN Miniports* sind eine spezielle Lösung von Windows mit *Wide Area Networks* also Weitverkehrsnetzwerken zu kommunizieren. Meist kommt hier das *Point to Point over Ethernet (PPOE)*- Protokoll zum Einsatz.

In der Praxis wird die Verbindung zum *WAN* heute von einem zentralen Router hergestellt, sodass diese *Netzwerkschnittstellen* meist nicht genutzt werden.

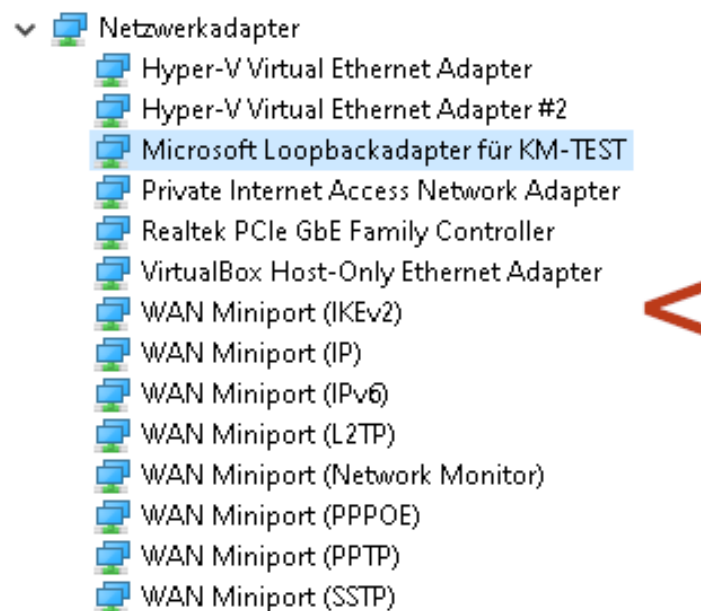


Abbildung 10: Microsoft Windows WAN Miniports