

1. Das "Smart-Steering"-Problem (Dualband-Router)

Moderne Router (z. B. Fritz!Box, Telekom Speedport) nutzen denselben Netzwerknamen (SSID) für das 2,4-GHz- und das 5-GHz-Frequenzband. Sie versuchen, Geräte dynamisch zwischen den Bändern hin- und herzuschieben. Der Raspberry Pi 5 verliert bei diesem Frequenzwechsel (Band-Steering) im Headless-Modus oft die Verbindung und meldet sich nicht wieder an. Die Lösung: Zwingen Sie den Pi 5 dazu, sich fest auf die MAC-Adresse (BSSID) Ihres 5-GHz- oder 2,4-GHz-Kanals zu binden. Verfügbare Bänder und deren BSSID (MAC-Adressen) scannen:

```
nmcli dev wifi
```

Suchen Sie Ihr WLAN aus der Liste und kopieren Sie die dazugehörige BSSID (z. B. 00:11:22:33:44:55). Binden Sie Ihre Verbindung fest an diese eine BSSID (ersetzen Sie IHR_WLAN_NAME und die Beispiel-BSSID):

```
sudo nmcli connection modify "IHR_WLAN_NAME" 802-11-wireless.bssid  
"00:11:22:33:44:55"  
sudo nmcli connection up "IHR_WLAN_NAME"
```

2. Der Router wirft den inaktiven Pi raus (Fehlender Heartbeat)

Viele Router trennen WLAN-Geräte nach einigen Stunden Inaktivität, um Funkkanäle freizumachen, wenn kein Datenverkehr registriert wird. Da Ihr Django-Webservice vermutlich primär auf Anfragen wartet, sendet der Pi von sich aus keine Pakete. Die Lösung: Richten Sie einen minimalen "Heartbeat" (Dauer-Ping) per Cronjob ein, der alle 5 Minuten ein winziges Paket an Ihren Router sendet und die Verbindung künstlich "wach" hält. Cronjob-Editor öffnen:

```
crontab -e
```

Fügen Sie ganz unten folgende Zeile hinzu (ersetzen Sie 192.168.178.1 durch die IP-Adresse Ihres Routers):

```
text*/5 * * * * ping -c 1 192.168.178.1 > /dev/null 2>&1
```

Speichern und schließen.

3. DHCP-Lease-Time läuft ab (IP-Zuweisung schlägt fehl)

Nach exakt 24 Stunden (oder der im Router definierten Lease-Time) muss der Pi seine IP-Adresse beim Router erneuern. Schlägt diese Anfrage im NetworkManager einmalig fehl (z. B. wegen minimalem Funkrauschen), bricht der Pi den Vorgang ab und versucht oft keine automatische Reaktivierung mehr. Die Lösung: Stellen Sie die WLAN-Verbindung im Pi von DHCP auf eine statische (feste) IP-Adresse um. Dadurch entfällt das fehleranfällige Aushandeln nach 24 Stunden. Statische IP zuweisen (Beispielhaft für eine freie IP 192.168.178.50, Gateway 192.168.178.1 und DNS 1.1.1.1):

```
sudo nmcli connection modify "IHR_WLAN_NAME" ipv4.addresses  
"192.168.178.50/24" ipv4.gateway "192.168.178.1" ipv4.dns "1.1.1.1"
```

```
ipv4.method "manual"  
sudo nmcli connection up "IHR_WLAN_NAME"
```

4. Ein "echtes" Auto-Reconnect-Skript (Das Sicherheitsnetz)

Um im Ernstfall (z. B. wenn der Router nachts zwecks Zwangstrennung oder Updates neu startet) zu verhindern, dass der Pi offline bleibt, installieren Sie ein kleines Watchdog-Skript, das die WLAN-Schnittstelle bei Verbindungsverlust automatisch neu startet. Erstellen Sie ein Skript:

```
sudo nano /usr/local/bin/wifi-check.sh
```

Fügen Sie folgenden Code ein (prüft, ob der Router erreichbar ist – falls nicht, wird das WLAN neu gestartet):

```
#!/bin/bash  
ROUTER_IP="192.168.178.1" # Bitte anpassen  
  
if ! ping -c 2 "$ROUTER_IP" > /dev/null 2>&1; then  
    echo "WLAN-Verbindung verloren. Starte neu..."  
    nmcli connection down "IHR_WLAN_NAME"  
    sleep 5  
    nmcli connection up "IHR_WLAN_NAME"  
fi
```

Ausführbar machen:

```
sudo chmod +x /usr/local/bin/wifi-check.sh
```

Als Cronjob alle 15 Minuten ausführen lassen (sudo crontab -e für Root-Rechte):

- `/15 * * * * /usr/local/bin/wifi-check.sh > /dev/null 2>&1`

From:
<https://wiki.serious-seeds.zwohundertvier.de/> - **Serious Seeds**

Permanent link:
https://wiki.serious-seeds.zwohundertvier.de/doku.php?id=netzwerk_issues_beheben&rev=1781342055

Last update: **2026/06/13 09:14**

