

Raspberry Pi 3(2) headless über WLAN in Betrieb nehmen

Quelle: <http://blog.helmutkarger.de/raspbian-headless-wifi-konfiguration-am-raspberry-pi-3/>
<https://pi-buch.info/wlan-schon-vor-der-installation-konfigurieren/>

1 Einleitung

Wer einen Raspberry Pi als Server betreibt, also ohne Bildschirm und Tastatur, der kann auch die Installation ‚headless‘ vornehmen. Das passiert per SSH über das Netzwerk.

Der Raspberry Pi 3 bringt - im Unterschied zu seinen Vorgängern - WiFi bereits onboard mit, so dass eine Inbetriebnahme/Konfiguration über WLAN sinnvoll ist. Aber wie soll man die WLAN Einstellungen headless konfigurieren? Wie es ohne LAN-Kabel, ohne Tastatur und ohne Monitor am RasPi funktioniert, zeigt dieser Artikel.

Ein LAN-Anschluss konfiguriert sich beim Anstecken des Kabels automatisch sofern ein DHCP-Server eine IP-Adresse liefert und in der boot-Partition ein leeres File mit dem Namen `ssh` angelegt wurde. So lässt sich sofort per SSH auf einen jungfräulich frisch aufgesetzten Raspberry Pi zugreifen und die Grundkonfiguration vornehmen. Detaillierte Informationen hierzu finden Sie in den Artikeln [Raspbian headless Installation per SSH](#).

Dieser Artikel beschränkt sich auf die Besonderheiten bei WiFi anstelle LAN.

2 Das Problem

Aus naheliegenden Gründen funktioniert eine headless Installation per WiFi erst mal nicht, da wir die SSID und den Netzwerkschlüssel nicht per SSH konfigurieren können, so lange das Netzwerk nicht funktioniert. Den Trick, wie man es trotzdem hinbekommt findet man in diesem [Artikel](#).

3 Die Lösung

Ein neueres Raspbian hat natürlich alle Treiber für den WLAN-Chip des Raspberry Pi 3 bereits mit dabei.

Aber auch ein Raspberry Pi 2 mit gestecktem USB-WLAN-Dongle sollte sich so in betrieb nehmen bzw. konfigurieren lassen.

Der Raspberry wird auch von sich aus versuchen WLAN zu aktivieren, wenn kein LAN-Kabel vorhanden ist. Trotzdem müssen wir dem RasPi zumindest die SSID und das WLAN-Passwort unseres Netzes mitgeben, sonst geht gar nichts. Wenn wir weder Tastatur/Monitor noch LAN-Kabel anstecken wollen, dann bleibt nur übrig, diese Informationen bereits auf die Micro-SD-Karte zu schreiben, bevor sie in den Raspberry gesteckt wird.

4 Schritt für Schritt

Wir gehen erst mal genauso vor, wie in meinem Artikel [Raspbian headless Installation](#) beschrieben:

- **Download** der neuesten Raspbian Software
- **Entpacken** der Datei
- **Schreiben** auf die Micro-SD-Karte mit dem Tool Win32DiskImager. Die SDcard sollte vorher mit dem Tool SDFormatter formatiert werden.

Diese drei Schritte können auf einem Windows- oder einem Linux-Rechner vorgenommen werden. Für den nächsten Schritt müssen wir unterscheiden, welche Raspbian Version wir verwendet:

- Raspbian vor Mai 2016 siehe Kapitel 4.1
- Raspbian ab Mai 2016 siehe Kapitel 4.2

Leider lässt sich nicht genau herausfinden, ab welcher Mai-Version es die entscheidende Änderung gab. Es gab im Mai 2016 die Versionen: 2016-05-10 und 2016-05-27 (es gibt keinen Vermerk in den Release Notes).

4.1 Raspbian Images vor Mai 2016

Um diese Images für WLAN fit zu machen, brauchen unbedingt Linux, denn wir müssen ein EXT4-Dateisystem mounten.

Wer keinen Linux-Rechner hat - nicht mal einen laufenden Raspberry Pi, der könnte über eine virtuelle Maschine zum Beispiel unter VirtualBox nachdenken, oder einen Windowsrechner [temporär von einer Linux-CD booten](#). Es geht nur darum, dass wir ein EXT4-Linux-Dateisystem mounten und dort eine Datei verändern können.

Das Mounten eines SD-Speichers funktioniert üblicherweise automatisch. Wenn wir das Raspbian-Image gerade eben mit Win32DiskImager auf die SD-Karte geschrieben haben, dann ziehen wir die SD-Karte, bzw den USB-Adapter kurz aus dem Rechner und stecken ihn dann zurück. Linux mountet dann zwei Dateisysteme: boot und eine 1,3GB große Partition mit einem langen kryptischen Namen.

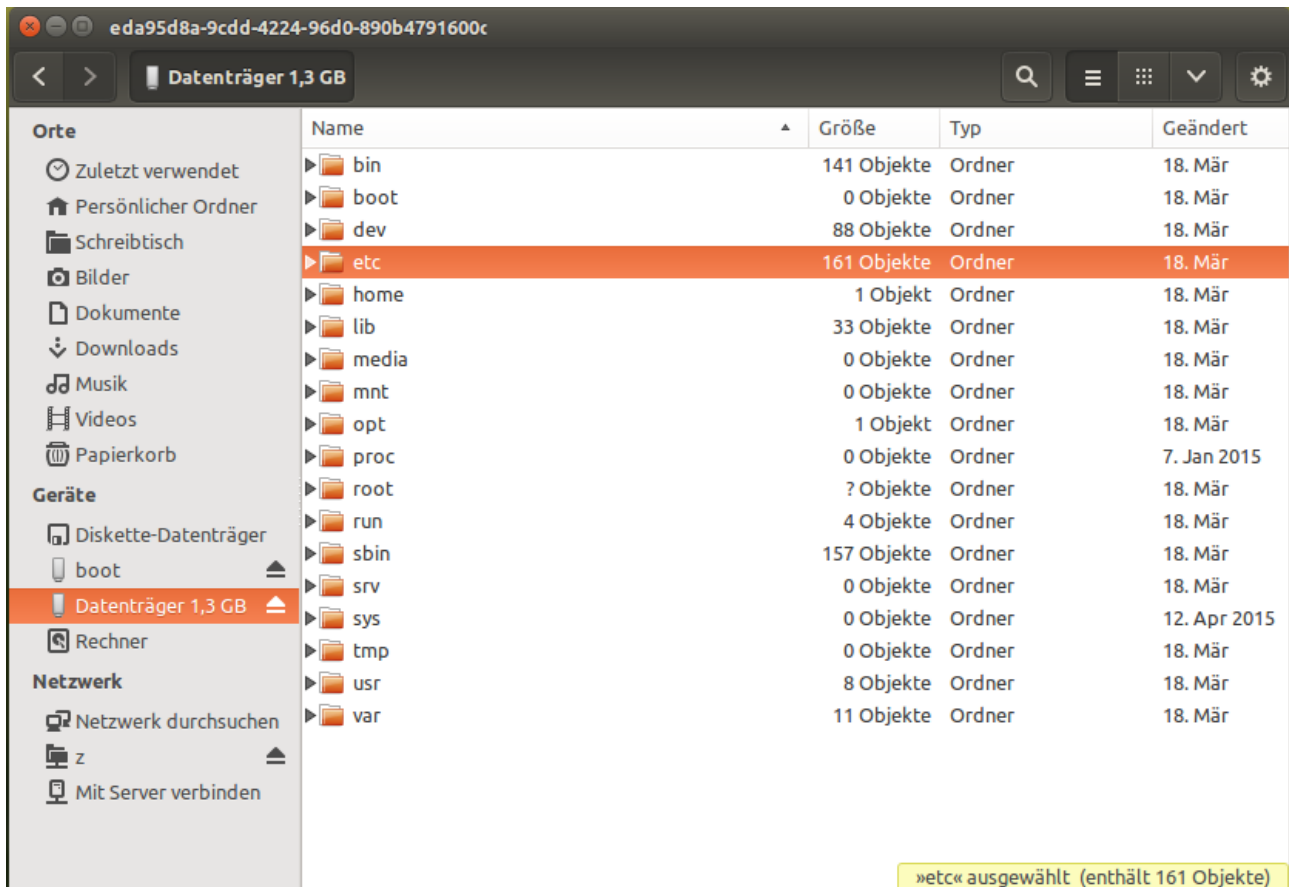


Abbildung 1: Screenshot Dateimanager

Der Ubuntu-Screenshot zeigt das sehr schön. Wir brauchen genau diese große Partition, die alle Raspbian Dateien enthält. Zum Bearbeiten von Dateien auf dieser Partition benötigen wir allerdings **Rootrechte**. Am einfachsten starten wir den Linux Dateimanager - in Ubuntu ist das Nautilus - per `sudo`, dann können wir uns weiter in der grafischen Benutzeroberfläche bewegen und brauchen nicht so viel tippen:

```
gksudo nautilus &
```

oder alternativ z.B. bei Lubuntu

```
gksudo pcmanfm &
```

GkSudo ist die grafische Variante des Sudo-Befehls und das kaufmännische Und hinter Nautilus bewirkt, dass der Dateimanager als unabhängiger Prozess gestartet wird und das Terminalfenster gleich wieder freigegeben wird (kann man auch weglassen). Im Dateimanager bewegen wir uns nun in besagter Partition ins Verzeichnis `etc/wpa_supplicant`. Dort befindet sich die Datei `wpa_supplicant.conf`, die wir editieren wollen. Dazu klicken wir die Datei mit der rechten Maustaste an und wählen ‚mit gedit öffnen‘. Der Inhalt der `wpa_supplicant.conf` sieht in etwa so aus:

```
country=GB
ctrl_interface=DIR=/var/run/wpa_supplicant GROUP=netdev
```

```
update_config=1
```

Den Countrycode können wir hier gleich von **GB** auf **DE** abändern, oder wir erledigen das später mit `raspi-config` bei den Internationalisation Options. Die wichtigen Ergänzungen sind aber folgende:

```
network={
    ssid="WLAN SSID"
    psk="WLAN Netzwerkschlüssel"
}
```

Diesen Block fügen wir ans Ende der Datei ein und passen `ssid` und `psk` entsprechend dem eigenen Netzwerk an. An sich war es das schon. Weiter mit Kapitel 4.3.

4.2 Raspbian ab Versionen Mai 2016

Ab der Mai 2016 Version wurde in Raspbian ein Mechanismus eingebaut, der es erlaubt eine notwendige Konfigurationsdatei auf der `root`-Partition anzulegen, die dann, falls vorhanden, beim ersten Start von Raspbian an die richtige Stelle kopiert wird.

In der `boot`-Partition der SD-Karte erstellen wir eine Datei `wpa_supplicant.conf` mit einem Texteditor, z.B. `notepad++`.

In die Datei schreiben wir folgende Zeilen:

```
country=DE
ctrl_interface=DIR=/var/run/wpa_supplicant GROUP=netdev
update_config=1

network={
    ssid="WLAN SSID" ← hier die eigene SSID eintragen
    psk="WLAN Netzwerkschlüssel" ← hier den eigenen WPA Schlüssel eint.
    key_mgmt=WPA-PSK
}
```

Nach dem Speichern der Datei sind wir mit der Vorbereitung der SD-Karte fertig. Weiter mit Kapitel 4.3.

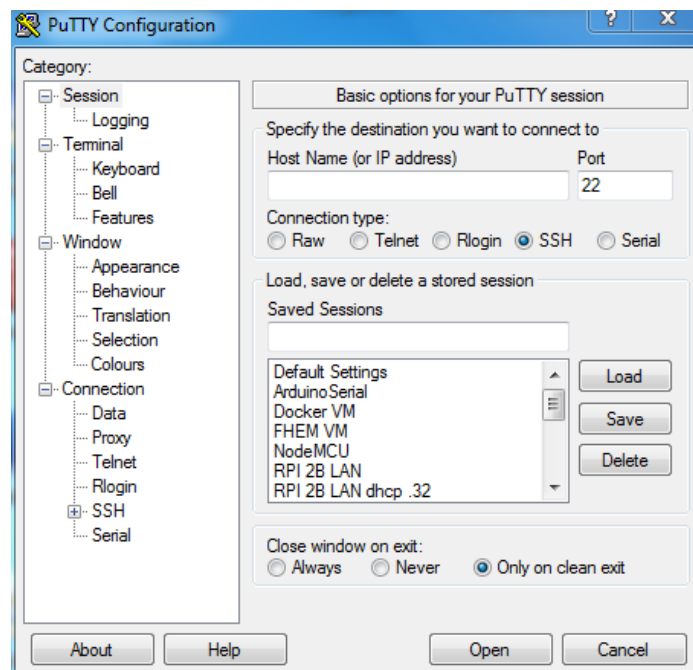
4.3 Weiter für alle Versionen

An dieser Stelle, wenn alles gespeichert ist, kann die Micro-SD-Karte in den Raspberry Pi 3(2) gesteckt und der mit Strom versorgt werden. Vorher empfiehlt es sich, einen Blick auf die Weboberfläche des WLAN-Routers (Fritzbox, oder was auch immer) zu werfen um beim Boot des RasPi sehen zu können, ob und welche IP-Adresse er zugewiesen bekommt.

Bei neueren Raspbian Versionen ist `ssh` per default nicht freigegeben. Damit aber ein Zugriff auf den Raspberry Pi per `ssh` möglich ist, muss auf der `root`-Partition der SD-Karte ein leerer File mit dem Filename `ssh` angelegt werden. Beim Booten prüft Raspbian ob es diesen File gibt und lässt dann einen Zugriff über `ssh` zu.

5 Grundkonfiguration des Raspberry

Voraussetzung für die weitere Konfiguration ist eine ssh-Verbindung zum Raspberry, z.B. über das Windows-Tool putty. Nach dem Start von putty erscheint ein Fenster, in das wir die IP-Adresse unseres Raspberry in das Feld Host Name eintragen. Wenn wir die gleiche Verbindung mehrfach



benutzen wollen, kann diese abgespeichert werden. Dazu in dem Feld 'Saved Sessions' ein Name eingeben und mit Klick auf 'Save' abspeichern. Über 'Load' kann die Konfiguration wieder geladen werden.

Bei der ersten Verbindung zu dem Raspberry kommt noch eine Abfrage, ob putty sich auch wirklich mit diesem Server verbinden soll. Das beantworten wir einfach mit ok.

Danach erscheint das Linux Login, bei dem wir uns jetzt als:

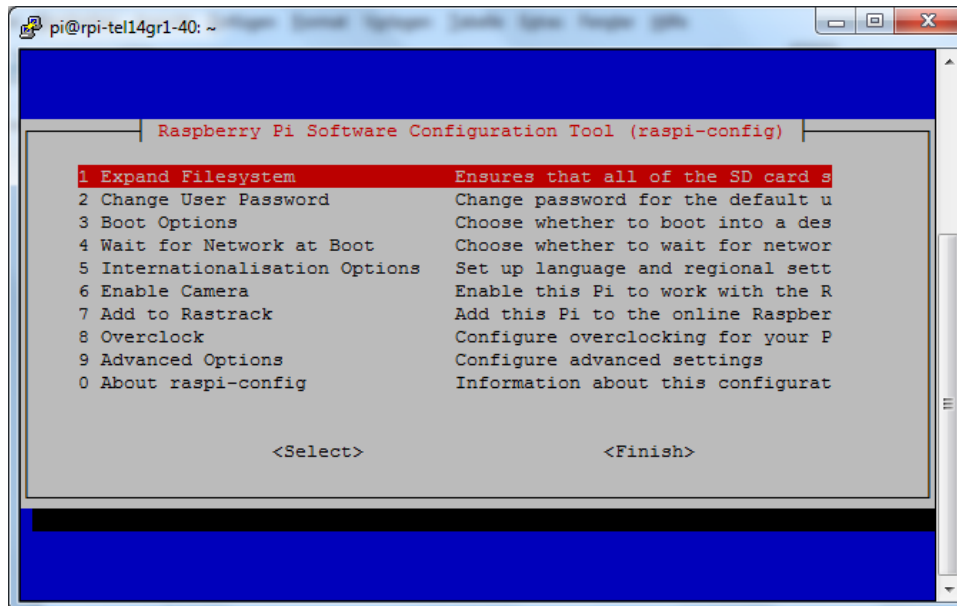
```
User: pi
PW:  raspberry
```

anmelden.

Für die 'Erstkonfiguration' starten wir das Raspbian Konfigurationstool `raspi-config`

```
$ sudo raspi-config
```

In dem nun erscheinenden Fenster



wählen wir von oben nach unten jetzt verschiedene Menü-Punkte aus. Die Navigation in dem Fenster erfolgt mit der Tab-Taste. Einzelne Felder können mit der 'Leerzeichen'-Taste ausgewählt werden.

- **Extand Filesystem:** um die gesamte Speicherkarte für Raspbian bereitzustellen (Wird bei neueren Raspbian Versionen auch schon automatisch nach dem ersten Start ausgeführt).
- **Boot Options:** hier wählen wir B3, damit Raspbian in die graphische Oberfläche bootet (evtl. heißt der Menüpunkt bei neueren Raspbian Versionen etwas anders).
- **Internationalisation Options:** dieser Punkt muss mehrfach ausgeführt werden um Land, Zeitzone, Tastaturlayout und Wi-Fi-Land zu wählen.

Hinweis: je nach Raspian Version kann das raspi-config Menü auch etwas anders aussehen.

Anschließend wird noch ein Update des Raspbian Betriebssystems durchgeführt, damit die aktuellen Sicherheits- und sonstigen Updates eingespielt werden.

```
$ sudo apt-get update  
$ sudo apt-get upgrade
```

Hinweis: Sollte die WLAN Verbindung zum Raspberry bei Inaktivität unterbrochen werden, könnte dies gegebenenfalls an der Stromsparfunktion des WLAN(Edimax)-Treibers liegen. Dann sollte diese deaktivieren werden. Dazu legen Sie eine Konfigurationsdatei für den Treiber an. In die Datei

`/etc/modprobe.d/8192cu.conf` wird die folgende Zeite geschrieben:

```
options 8192cu rtw_power_mgnt=0 rtw_enusbss=0
```

für andere Treiber kann die Zeile anders aussehen.

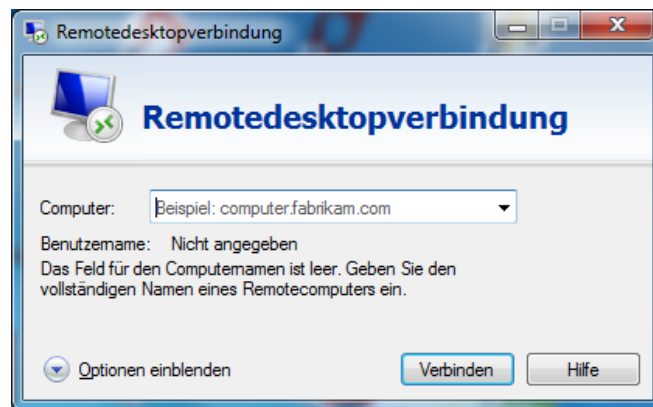
6 Remote Zugriff auf den Desktop einrichten

Um auf den graphischen Desktop des Linux-Raspbian zuzugreifen, müssen wir noch ein geeignetes SoftwarePaket nachinstallieren.

Für Raspbian Versionen **VOR** 'raspbian-jessie 2016-09-23' (z.B. raspbian-jessie 2016-03-18) wird dazu das Paket xrdp installiert.

```
$ sudo apt-get install xrdp
```

Danach kann auf dem Windows Rechner das Programm Remote Desktop (unter Zubehör) gestartet werden



In das Feld 'Computer' geben wir die IP-Adresse des Raspberry ein und klicken dann auf 'Verbinden'. Die folgende Sicherheitsabfrage kann dann bestätigt werden.

Wenn alles richtig konfiguriert ist, meldet sich xrdp mit einem Anmeldefenster, in das dann der Benutzername `pi` und das Passwort `raspberrypi` eingegeben wird. Anschließend sollte der Raspbian Desktop zu sehen sein.

Wer xrdp mit einer neueren Raspbian version verwenden möchte, muss zunächst den schon per default installierten realVNC-Server deinstallieren. Dazu verwenden wir das Kommando

```
sudo apt-get remove realvnc-server
```

Danach kann wie oben beschrieben, `xrdp` installiert werden.

Ab der Raspbian Versionen 'raspbian-jessie 2016-09-23' kann über raspi-config der VNC-Server für den Remote Desktop freigeschaltet werden. Dazu wird unter dem Menüpunkt 'Advanced Options' VNC auf enable gesetzt. Siehe hierzu auch <https://www.realvnc.com/docs/raspberrypi.html>

Weil kein Bildschirm an den Raspberry angeschlossen ist, gibt Raspbian den Desktop mit einer sehr kleinen Größe aus. Damit eine vernünftige Desktop Größe eingestellt wird, siehe hierzu auch <https://support.realvnc.com/knowledgebase/article/View/523>, muss die Datei `/boot/config.txt` mit einem Editor

```
$ sudo nano /boot/config.txt
```

um die folgenden Zeilen erweitert werden.

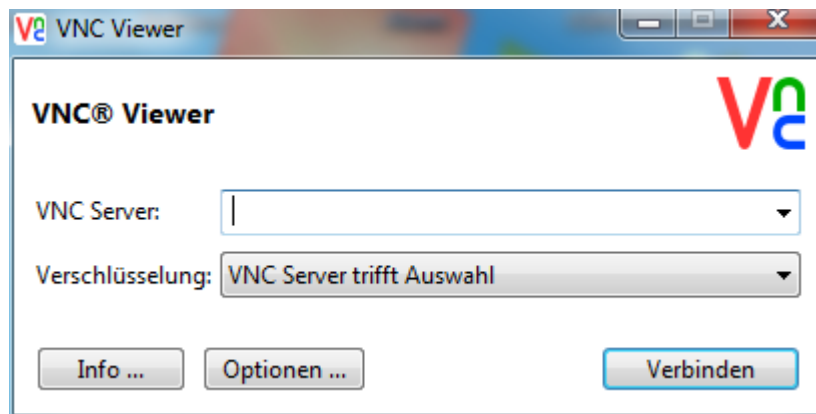
```
# Set display resolution for VNC headless operation
hdmi_force_hotplug=1
hdmi_ignore_edid=0xa5000080
hdmi_group=2
# hdmi_modes within hdmi_group=2 see also
# https://www.raspberrypi.org/documentation/configuration/config-txt.md
# 9 = 800x600 at 60Hz
# 16 = 1024x768 at 60Hz
hdmi_mode=16
```

Danach ist ein Reboot des Raspberry notwendig:

```
$ sudo reboot
```

Der Remote-Zugriff auf den Raspbian Desktop erfolgt von einem Windows-Rechner mit einem VNC-Client(Viewer), z.B. von realvnc <https://www.realvnc.com/download/viewer/>

Nach dem Start des VNC Client erhält man folgendes Fenster:



In dem Feld 'VNC Server' wird die IP-Adresse des Raspberry eingegeben. Nach einem Klick auf 'Verbinden' erscheint ein Anmeldefenster in das noch der Benutzername und das Passwort für den raspberry eingegeben werden muss.

Danach sollte der Raspbian Desktop sichtbar sein.

7 Links

Win32DiskImager <https://sourceforge.net/projects/win32diskimager/>

SDFormatter https://www.sdcard.org/downloads/formatter_4/