

M126 - Peripheriegeräte im Netzwerkbetrieb einsetzen

Eine Projektarbeit von:

Luis Ricardo Lüscher

IT Apprentice

SIX Group Services AG

Im Auftrag von:

Technische Berufsschule Zürich

Ausstellungsstrasse 70

8005 Zürich

Kontakt:

Luis Ricardo Lüscher

8102 Oberengstringen

luisricardoluescher@gmail.com

Inhaltsverzeichnis

Inhaltsverzeichnis	2
Vorwort	5
Vorwissen	5
Ausrüstung	5
Bewertungskriterien	6
Distance Learning	7
Einführung	7
IoT	7
Raspberry	7
Netzwerk	7
Moduljournal	7
Tag 1 (17.02.2020)	7
Tag 2 (24.02.2020)	8
Nebenarbeiten (26.02.2020)	8
Nebenarbeiten (29.01.2020 & 01.03.2020)	8
Tag 3 (02.03.2020)	9
Tag 4 (09.03.2020)	9
Tag 5 (16.03.2020)	9
Nebenarbeiten (17.03.2020 - 22.03.2020)	10
Tag 6 (23.03.2020)	10
Nebenarbeiten (26.03.2020 - 29.03.2020)	10
Tag 7 (30.03.2020)	10
Nebenarbeiten (05.04.2020)	10
Tag 8 (06.04.2020)	11
Dokumentation	12
Fixe IP und weitere Einstellungen	12
Fixe IP	12
Hostname	12
VNC aktivieren	13
Partitionen auslesen	14
SSD-Geschwindigkeit auslesen	15
Passwort ändern	15
SSHPort wechseln	15
Root Login disabled	15
Demoprojekt	15
W01 ngrok Tunnel	21

Installation und Konfiguration	21
Erklärung ngrok Tunnel - How it works	26
W02 Print Server & static IP	27
W04 RPI Monitor	37
W05 Netttime NTP	41
W06 NAS & SMB	44
W07 Webmin	51
W10a GrovePi Einführung	61
W10b GrovePi Sensor	63
W10c GrovePi Twitter Feed	65
Zusatz Digitec Preis Twitter	70
W10e GrovePi LCD	73
W16 HiFi-Berry & WebRadio	77
W24 RASPAP	81
Eigene Projekte	86
W95 PIVPN	86
W96 Plesk Media Server	97
W97 Pi-Hole	103
W98 Minecraft Server mit Cuberite	110
W100 Surveillance Station	116
W101 Discord Python Bot	120
W103 RDP Raspi	125
Glossar	127
Proxy	127
Was ist ein Proxy Server?	127
Wozu dient ein Proxy Server	127
Localhost	127
Putty	128
VNC	128
Stream-Multiplexing	128
Heartbeat-Mechanismus	128
Polling-Anfragen	129
Firewall	129
RAID	129
Software vs. Hardware RAID	129
Software RAID	129
Hardware RAID	129
Cache RAID Controller	130
Sicherungstechniken	130
Spiegelung	130
Streifen	130

Parität	130
RAID 0	130
RAID 1	130
RAID 5	131
RAID 6	131
RAID 10	131
GrovePi Erklärung	131
Linux Grundbefehle	132
Fazit	132
Quellenangaben	132
Zusammenarbeit	132
Inhaltliche Quellenangaben	132

Vorwort

Im Modul 126 - Peripheriegeräte im Netzwerk betreiben, geht es darum Peripheriegeräte im Netzwerk einzubinden. Durch kleinen Recheneinheiten, die über einen Netzwerkanschluss verfügen. In diesem, Modul werden wir mit Raspberry Pi's arbeiten, jedoch gibt es auf dem Markt viele andere alternativen wie zum Beispiel das Asus Tinkerboard S, Banana Pi M3, ODROID - N2 oder auch dem Atomic Pi. Gedacht sind diese Geräte um die Welt der Informatik besser kennenzulernen, man kann so seine Fantasie frei im Laufen lassen.

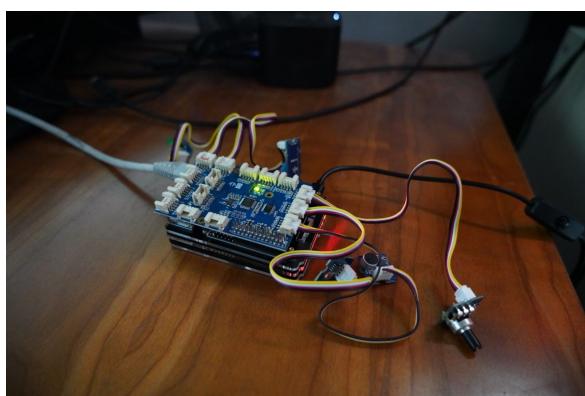
Ich habe mich bewusst für eine Einzelarbeit entschieden, aufgrund dessen, dass ich einfach nicht möchte das andere in meiner Klasse durch meinen extra Effort, Vorteile in Bezug auf die Benotung erhalten. Ich selber kenne mich relativ gut aus mit Linux und habe auch das Ziel in diesem Modul besonders gut abzuschliessen.

Vorwissen

Bereits vor meiner Lehre habe ich mit Raspberry Pi's gearbeitet. So habe ich zum ersten mal bei dem Selektionsverfahren an der ETH Zürich die Welt der RasPi und des OS Linux kennengelernt. Zuhause habe ich mich dann mehr theoretisch über diese Themen informiert. Im späteren Verlauf meiner Lehre hatte ich dann immer mehr mit Raspberry Pi's zu tun.

Ausrüstung

Von der Schule aus habe ich einen Raspberry Pi's 3B + mit einer 16 GB SD karte erhalten. Als kleines Depot musste ich Herr Calisto 20 CHF geben. Dazu habe ich noch ein Netzteil und einen GrovePi erhalten. Von meinem Betrieb habe ich auch noch einen Raspberry Pi 4 erhalten jedoch kann ich auf diesem nicht die GrovePi Aufgaben erledigen, darum werde ich wohl beide verwenden.



Dies ist der Raspberry Pi 3b+ von der TBZ.
Ich werde mit diesem einige GrovePi Aufträge erledigen.
Folgende Spezifikation:

CPU: 4 x 1.2GHz
RAM: 1 GB
Anschlüsse: 4 x USB2.0, 1 x HDMI
Bluetooth: 4.1



Dies ist der Pi den ich von meinem Betrieb zur Verfügung gestellt bekommen habe.

Folgende Spezifikation:

CPU: 4 x 1,5 GHz

RAM: 4 GB

Anschlüsse: 2 x Micro HDMI, 2 x USB 2.0, 2 x USB 3.0

Speicherkarte: 16 GB Micro SD Karte

Bluetooth: 5.0

Bewertungskriterien

Im ersten Teil des Moduls erlernen wir die theoretischen Grundlagen in bezug auf IoT und den Raspberry Pi. Dieses Wissen wird durch eine leistungsbeurteilung überprüft, die LB1. Danach ist es so, dass sich die Teams selbstständig neues Wissens im Rahmen des zweiten Teils des Moduls aneignen und dies dann Herrn Calisto zeigen, in Form einer Abnahme verschiedener Lernziele pro Werkstattauftrag die auf dem BSCW hinterlegt sind. Die einzelnen Werkstattaufträge geben dann unterschiedliche Punktzahlen, die dann die endgültige Benotung ausmachen.

Lernmap

Namen: (A) _____ (B) _____ Klasse: _____ Note: _____

Teil 1: Grundlagen		Teil 2: Werkstattarbeiten (mind. 6 x 4 Lektionen)			
Min. Abdeckung der HnKs		W00 Service	W10 GrovePi	W10 IO	W20 Challenge
00 Einstieg Modul, Vorbereitung, IoT HA: eNote = ePortfolio	Lektion 2	W01 Web Tunnel (__/4+P)	W01a GrovePi (__/6+P)	W11 Kamera (__/4-8P)	W21 TOR Proxy (__/4+P) *SD
01 Einstieg Raspberry Pi & OS Install via VNC & SSH, SFTP <i>00raspberry_VNC_Data_16GB.zip</i>	3.4	W02 Printserver & statische IP-Adr. (__/10P)	W10b GrovePi Sensor (__/4+P)	W12 NFC (__/4P)	W22 WebCAM (__/4+P)
02 Repetition Netzwerk, Integration 4 Varianten <i>01raspberry_TBZ_Data_16GB.zip</i>	5.6	W03 DynDNS (& IPv6) (__/4 (& 3-6P)) <i>Nur jeweils 1 Gruppe!</i>	W10c GrovePi Twitter (__/4P)	W13 PiGlow (__/4P)	W23 7"-TouchScreen mit Digital Wall Clock (__/2-7P)
03 Security mit Raspberry Pi Überwachungskameras & rechtliche Grundlagen	7.8	W04 RPi Monitor (__/4P)	W10d GrovePi GPS (__/5+P)	W14 unicornHAT (__/4P)	
Theoretetest Note: (A) ____ (B) ____	9	W05 Netttime NTP (__/4P)	W10e GrovePi LCD (__/5+P)	W15 senseHAT (Emulator) (__/5+P)	
05 Demoprojekt: Webserver einrichten GrovePi mit LED	10-12	W06 NAS (__/8P)		W16 HiFi / WebRadio (__/6+P)	
06 Lerndokumentation. Setup Werkstattaufgaben (__/10P)	13	W07 Webmin (__/5+P)	W10g GrovePi Eigene Idee (__/5+P)		W27... Eigene Idee (__/5+P)
→ Lehrer-Image: (.imz) <i>00raspberry_TBZ_Data_16GB.zip</i>		W08 OwnCloud (__/9+P) *SD			Ideen z.B. unter: http://raspberry.tips/

*SD: Es empfiehlt sich hier eine „neue“ SD-Karte zu verwenden. Fragen Sie beim Lehrer nach einem geeigneten Lehrer-Image!

Schwierigkeitsgrad: 1-6

Zusätzliches Material notig

ca. 55 P=Note 6

Der grüne Bereich der Lernmap bezeichnet die Ziele des Teil 1 und der weiße Bereich die Werkstätte des Teil 2.

Distance Learning

Distance Learning bezeichnet die Schulung von Schülern in einer nicht physischen Form. Einfach gesagt, die Schüler und Schülerinnen werden über das Internet unterrichtet. Aufgrund der Auswirkung der COVID-19 Pandemie, wurden wir von der TBZ aus, seit dem BR Entscheid des 13. März 2020 von zuhause aus unterrichtet. Hier geht auch ein grosses Lob an Marcello Calisto, er hat über das folgende Wochenende (14. & 15. März) eine komplette virtuelle Lernumgebung aufgebaut, sodass wir am nächsten Montag ganz normal weiter arbeiten konnten. Zu den davorstehenden Aufgaben, habe ich einen LinkedIn Post verfasst.

[LinkedIn Post](#)

Einführung

Sämtliche Dokumentationen sowie Tutorials zum M126 sind auf folgender [Website](#) einsehbar.

IoT

<https://m126.luis-luescher.com/?portfolio=00-einstieg-modul>

Raspberry

<https://m126.luis-luescher.com/?portfolio=00-einstieg-modul>

Netzwerk

<https://m126.luis-luescher.com/?portfolio=00-einstieg-modul>

Moduljournal

Tag 1 (17.02.2020)

Da Herr Calisto nicht da war hatten wir eine Vertretung. Ich kannte die Vertretung bereits durch ein anderes Fach. Zuerst erklärte die Lehrperson uns um was es im Modul geht. Er ging mit uns die Modulbeschreibung durch und gab noch einige Hinweise. Danach ging er auf das heutige Hauptthema ein, IoT (Internet of Things). Der Inhalt seinerseits war sehr interessant, jedoch fand ich dass der Input ein wenig einseitig war und wir eigentlich nur zuhören mussten. Zwei Stunden zuhören fand ich dann relativ monoton und mit der Zeit auch langweilig. Später konnten wir dann auch noch Ideen entwickeln, wo man IoT ebenfalls verwenden könnte.

Tag 2 (24.02.2020)

Heute gab uns Herr Calisto einen kurzen Input zu seinen Lernprojekt in den Philippinen. Danach haben wir nochmals eine Erklärung seinerseits erhalten zum Modul und zu den Werkstatt Aufgaben, da noch nicht alle in der Klasse verstanden haben wie das nun genau abläuft. Danach ging es endlich an die Raspberry Pi's. Zu Beginn habe ich alle Komponenten angeschlossen (SSD, LAN, Maus, Tastatur und Monitor). Danach habe ich dem Rasberry Pi eingeschalten, ihm eine fixe IP zugewiesen und dann nur noch von meinem Notebook via SSH gearbeitet. Dann musste ich noch weitere Konfiguration vollziehen. Nachdem ich fertig war, half ich Herr Zimmermann und Schuhmacher sowie Herr Koch und Abedini bei der Konfiguration. Insgesamt war es ein erfolgreicher Tag, ich konnte mein bereits vorhandenes Wissen anwenden und weitergeben. Zuhause werde ich evtl. schon an einigen Werkstattaufgaben arbeiten, zudem werde ich mit Ryan Simmonds und Fabio Lichtler der ST18A bei uns im Betrieb nochmals das Thema „Raspberry Pi's“ genauer anschauen, unsere Berufsbildner sind sehr offen gegenüber Arbeiten für die Schule im Betrieb was ich natürlich super finde, da steht einer guten Note nicht im Wege. Zudem werde ich mit meinem Lehrmeister reden, ob ich für Tests und für die Schule einen Raspberry Pi kaufen könnte und sich der Betrieb daran beteiligt, dann könnte ich eben einige Aufgaben bereits zuhause machen und mir mehr mühe geben mit der Dokumentation.

Nebenarbeiten (26.02.2020)

Heute habe ich im Betrieb an einigen Dingen bezüglich des Modules gearbeitet. Einerseits habe ich die Website erstellt und bereits angefangen kleinere Artikel zu schreiben. Zudem habe ich zusammen mit Fabio Lichtler (ST18B) unseren Vorgesetzten (Diego Suter) gefragt, ob es die Möglichkeit gäbe, einen Raspberry Pi zu kaufen und das unser Betrieb für die Kosten aufkommen würden. Wir erklärten Ihm wie das Modul aufgebaut ist und welche Vorteile es hätte wenn wir die Raspberry Pi's privat nutzen könnten. Der Hauptgrund für den eigenen Raspberry war, dass wir dann auch zu Hause daran arbeiten können. Zudem würde sich bei mir die Dokumentationsarbeit erleichtern, da ich normalerweise über VPN arbeite und somit immer über meine private Public IP arbeite (sprich diese welche ich zuhause habe), zudem hat mein Heimnetzwerk einen komplett anderen Netzanteil bei den IP's als den Range der TBZ.

Nebenarbeiten (29.01.2020 & 01.03.2020)

Am Freitag gab mir mein Lehrmeister einen Raspberry Pi 4 mit 4 GB RAM. Am Freitag habe ich dann den Raspberry Pi konfiguriert. Am Samstag begann ich dann einige Werkstattaufgaben zu machen. Ich hatte grossen Spass daran, am Sonntag machte ich weiter. Nun habe ich bereits 6 Werkstattaufgaben erledigt und muss sie nun nur noch dokumentieren. Ich hatte Probleme mit dem TOR Proxy, ich werde meine Fehleranalyse mit

Herr Calisto anschauen. Ich denke es liegt an der Weiterleitung der kommenden Paket durch den wlan0 Anschluss auf den eth0 Anschluss.

Tag 3 (02.03.2020)

Heute habe wir zum ersten mal ein richtiges Projekt gemacht mit dem RasPi. Zu Beginn der Lektion haben wir zuerst versucht über eine Konfigurationsdatei die Credentials für eine automatische Anmeldung an einen Access Point angegeben. Das password wurde in diesem Fall als Hash gespeichert. Da dies leider in der Klasse nicht zu gut funktionierte, haben wir weiter am Demoprojekt gearbeitet. Hier mussten wir mit GrovePi und einer kleinen Lampe mittels eines Python Scripts diese Lampe zu Beginn fünf mal leuchten lassen, danach ein Script schreiben welches die Lampe ausschaltet und eines welches die Lampe einschaltet. Später mussten wir dann noch diese Scripts auf einer Website implementieren, sodass man via einem GUI die LED Lampe ein- und ausschalten konnte. Das Projekt war sehr spannend und war relativ simple. Ich hoffe die nächsten Aufträge werden mich ein wenig mehr herausfordern.

Tag 4 (09.03.2020)

An diesem Montag begannen wir den Unterricht mit einem Kahoot. Das Kahoot schloss ich eher semi gut ab, insgesamt hatte ich 15 von 20 Fragen richtig und wurde damit leider nur zweiter. Danach hatten wir die LB1. Der Test verlief denke ich ganz gut. Als alle den Test abgeschlossen haben, begannen wir mit den Werkstattaufträgen, da ich einige schon zuhause gemacht habe, arbeitete ich an den GrovePi Aufträgen, da ich diese nur in der Schule machen kann. Die Dokumentationsarbeit mache ich dann zuhause oder im Büro. Ich gab bereits den GrovePi 10a Werkstatt Auftrag ab und habe ebenfalls 10b schon gemacht und arbeite momentan an 10c, ich hatte da leichte Probleme mit dem API Keys für meinen Twitter Account. Zuhause und im Büro werde ich dann die ganze Dokumentationsarbeit machen.

Tag 5 (16.03.2020)



Heute habe wir das erste mal Distance Learning gehabt, so haben sich einige Lernende vor dem Schulhaus getroffen und noch einen RasPi geholt. Ich habe nun einen RasPi mit einem GrovePi mitgenommen. Ich werde versuchen die Aufträge 10a, 10b, 10c und 10e zu machen. Nach dem Mittagessen haben wir uns dann in Teams Chat in einer Konferenz getroffen. Herr Calisto hat uns einiges zum Thema Distance Learning gezeigt und dann noch erklärt wie das mit der Leistungsabnahme funktioniert. Danach habe ich an einigen Dokumentationen für die Werkstattaufträge gearbeitet.

Nebenarbeiten (17.03.2020 - 22.03.2020)

Am Mittwoch habe ich die erste Leistungsabgabe gehabt. Insgesamt habe ich vier Aufträge abgegeben. Ich hätte ohne Problem mehr Aufträge abgeben können, jedoch wurden die anderen Aufträge nicht von mir dokumentiert. Dies werde ich am kommenden Samstag machen. Am Samstag habe ich begonnen weitere Werkstattaufträge zu dokumentieren. Am Morgen habe ich das Demoprojekt dokumentiert, sowie noch ein Video gemacht um die Funktionalität zu beweisen.

Tag 6 (23.03.2020)

Heute habe ich insgesamt neun Aufträge abgegeben, die Leistungsabgabe mit Marcello war noch sehr lange (ca. 1h). Ich denke trotzdem ich habe eine tolle Arbeit gemacht und der Dozent war auch dementsprechend zufrieden. Ich muss noch beim Auftrag noch einige Dinge nachliefern in Bezug auf den htop Prozessmanager. Ich habe ebenfalls gesagt, dass ich nun an eigenen Projekten arbeiten werde und die Aufgaben auf der Lernmap nicht mehr wahrnehmen werde, da diese mich weniger interessieren. Zudem möchte ich lieber Projekte angehen, die mir auch einen persönlichen Mehrwert bringen. So werde ich nun an einem TOR Proxy, einem Pi Hole und einer Docker Surveillance Station arbeiten.

Nebenarbeiten (26.03.2020 - 29.03.2020)

Am Donnerstag habe ich einige eigene Projekte umgesetzt. So habe ich insgesamt 3 Projekte erarbeitet. Am Freitag habe ich dann weitere gemacht. Ich werde die Dokumentation auf Montag verschieben, da ich am Wochenende noch relativ viel für das Modul 214 machen muss.

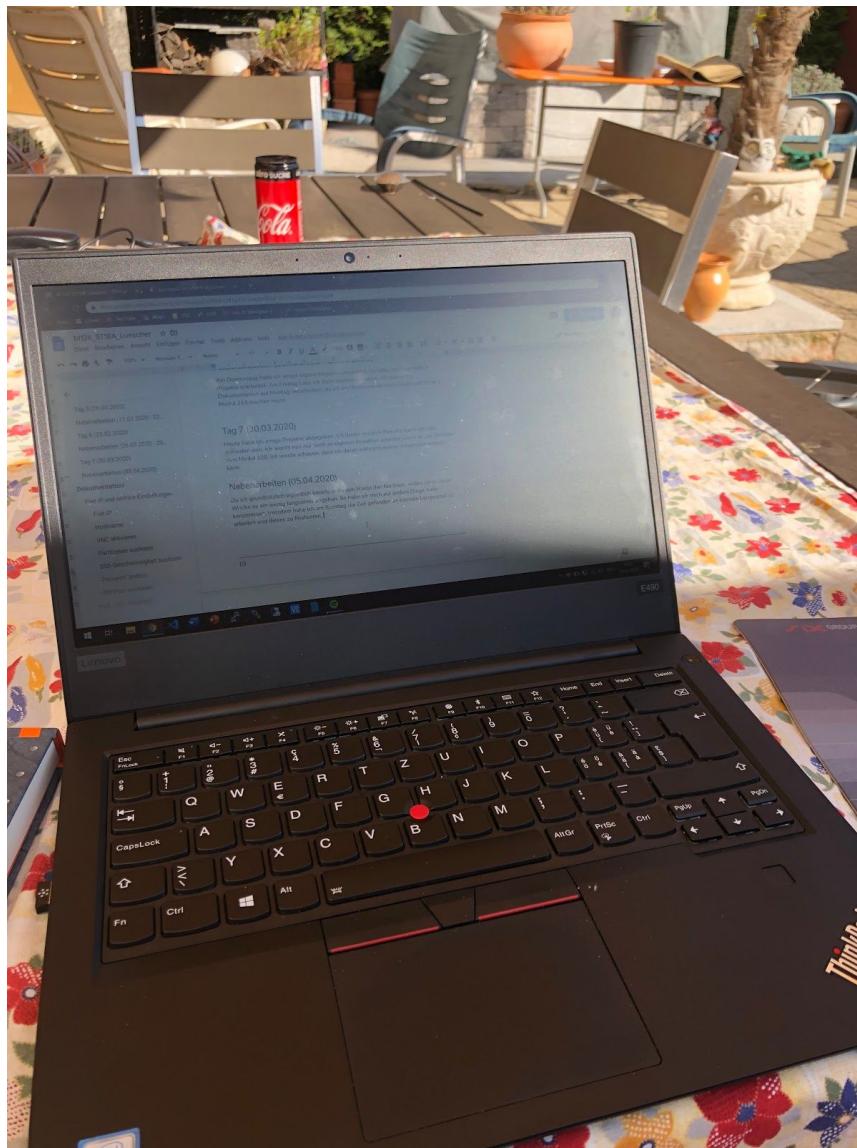
Tag 7 (30.03.2020)

Heute habe ich einige Projekte abgegeben. Ich denke mit dem Resultat kann ich sehr zufrieden sein. Ich werde nun nur noch an eigenen Projekten arbeiten sowie an der Website zum Modul 126. Ich werde schauen, dass ich daran während meiner Arbeitszeit arbeiten kann.

Nebenarbeiten (05.04.2020)

Da ich grundsätzlich eigentlich bereits in diesem Modul den 6er habe, wollte ich es diese Woche es ein wenig langsamer angehen. So habe ich mich auf andere Dinge mehr konzentriert, trotzdem habe ich am Sonntag die Zeit gefunden an meinem Lernjournal zu arbeiten und dieses zu finalisieren. Ich habe nun das Lernjournal so angepasst, dass ich es

genau so abgeben würde. Eventuell werde ich am Montag noch einige Dinge verbessern.



Tag 8 (06.04.2020)

Heute wäre eigentlich geplant gewesen, dass wir die Materialien zurückbringen an die TBZ. Dieses vorhaben wurde jedoch durch einen internen Entscheid gestoppt. Ich habe weitere Arbeiten an meinem Lernjournal getätig.

Dokumentation

Fixe IP und weitere Einstellungen

Fixe IP

Zuerst öffnen wir das dhcpd.conf file mit einem beliebigen Editor.

```
pi@z54:~ $ sudo nano /etc/dhcp/dhcpd.conf
```

Danach entfernen wir die Rautezeichen und setzen die entsprechenden Parameter des Netzwerks.

```
#slaac private

#Example static IP configuration:
interface eth0
static ip_address=172.16.17.54/24
#static ip6_address=fd51:42f8:caae:d92e:ff/64
static routers:172.16.17.1
static domain_name_servers=172.16.17.1 8.8.8.8 fd51:42f8:caae:d92e::1

# It is possible to fall back to a static IP if DHCP fails:
```

Hostname

Nun öffnen wir die Hostname Datei.

```
pi@z54:~ $ sudo nano /etc/hostname
```

Und danach kann man einfach den gewünschten Hostnamen eingeben.

```
z54
```

Nun haben wir den Hostname und die IP-Adresse gesetzt, jetzt nur noch die Hosts Datei bearbeiten.

```
pi@z54:~ $ sudo nano /etc/hosts
```

Nun im unteren 127.0.0.1 eintragen, den vorhin gesetzten Hostname hinschreiben.

```
127.0.0.1      localhost
::1            localhost ip6-localhost ip6-loopback
ff02::1  ip6-allnodes
```

```
ff02::1 ip6-allrouters
```

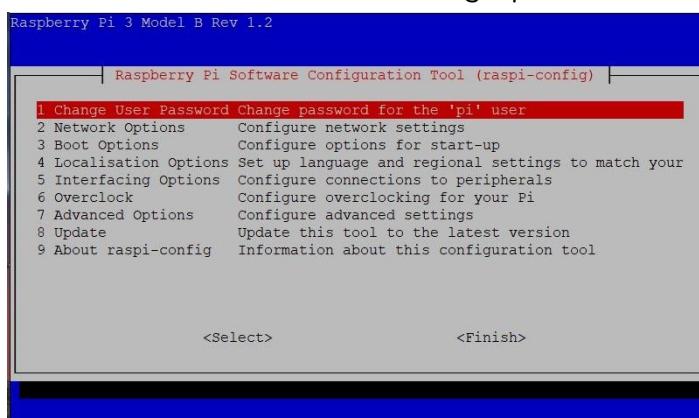
```
127.0.0.1      z54
```

VNC aktivieren

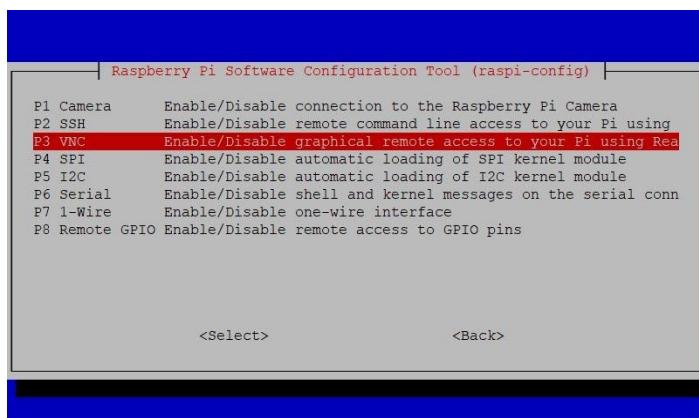
Nun müssen wir den Raspi-Konfigurator öffnen.

```
pi@z54:~ $ sudo raspi-config
```

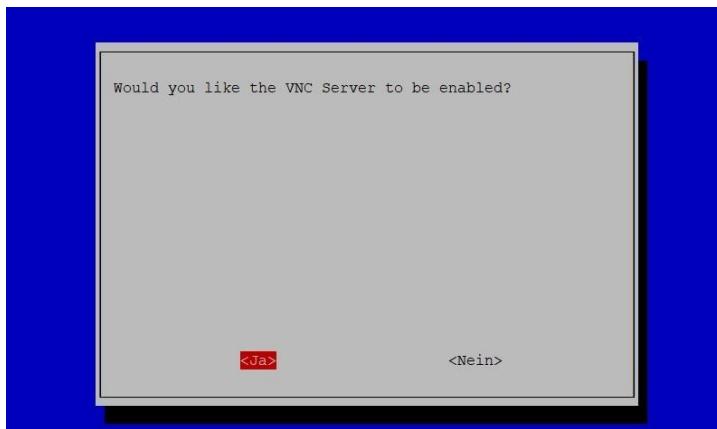
Nun einfach den 4. Punkt Interfacing Options öffnen.



Danach den 3. Punkt VNC auswählen.



Und dann nur noch die Aktivierung mittels "Ja" bestätigen.



Partitionen auslesen

Nun messen wir die Partitionen aus.

```
pi@z54:~ $ lsblk
```

Der Output sieht dann folgendermassen aus.

```
NAME      MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
mmcblk0   179:0    0 14,9G  0 disk 
└─mmcblk0p1 179:1    0  2,4G  0 part 
└─mmcblk0p2 179:2    0    1K  0 part 
└─mmcblk0p5 179:5    0   32M  0 part 
└─mmcblk0p6 179:6    0 256M  0 part /boot 
└─mmcblk0p7 179:7    0 11,3G  0 part /
└─mmcblk0p8 179:8    0   512M 0 part /media/pi/data
```

Es gibt auch noch diese Möglichkeit.

```
pi@z54:~ $ df -h
```

Der Output sieht dann folgendermassen aus.

```
Dateisystem  Größe Benutzt Verf. Verw% Eingehängt auf
/dev/root     11G   3,0G  7,5G  29% /
devtmpfs      459M     0  459M  0% /dev
tmpfs         464M     0  464M  0% /dev/shm
tmpfs         464M   6,3M  457M  2% /run
tmpfs         5,0M   4,0K  5,0M  1% /run/lock
tmpfs         464M     0  464M  0% /sys/fs/cgroup
/dev/mmcblk0p6 253M   54M  199M  22% /boot
tmpfs         93M     0   93M  0% /run/user/1000
/dev/mmcblk0p8 488M  408K  452M  1% /media/pi/data
```

SSD-Geschwindigkeit auslesen

```
pi@z54:~ $ dd if=/dev/zero of=tempfile bs=1M count=1024 conv=fdatasync,no trunc
```

Der Output sieht dann folgendermassen aus.

```
1024+0 records in  
1024+0 records out  
1073741824 bytes (1,1 GB, 1,0 GiB) copied, 7,45128 s, 144 MB/s
```

Passwort ändern

```
pi@z54:~ $ passwd
```

SSH Port wechseln

```
pi@z54:~ $ sudo nano /etc/ssh/sshd_config
```

Port 22 ersetzen durch Port 1357

Nun starten wir den SSH Service neu.

```
pi@z54:~ $ sudo /etc/init.d/ssh restart
```

Root Login disabled

```
pi@z54:~ $ sudo nano /etc/ssh/sshd_config
```

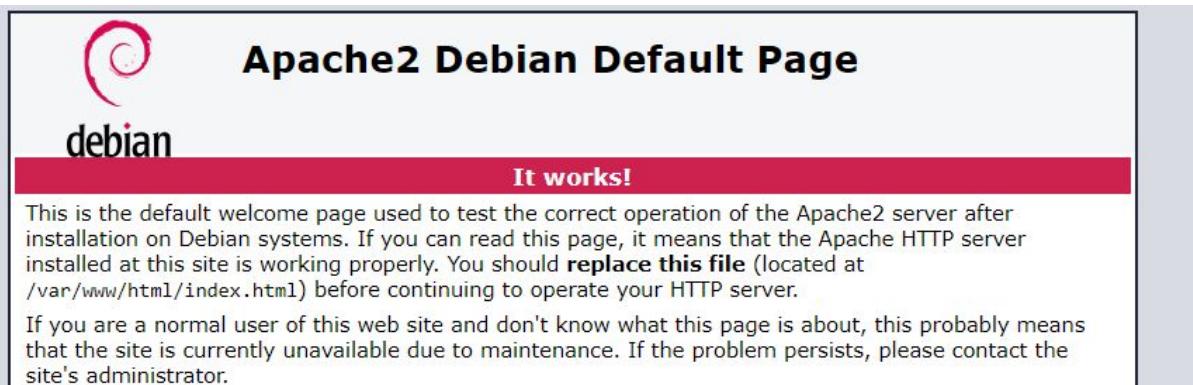
“PermitRootLogin yes” ersetzen mit “PermitRootLogin no”

Demoprojekt

Zuerst bringen wir die Repository List up to date. zudem laden wir den Apache Webserver herunter.

```
pi@z54:~ $ sudo apt-get update  
pi@z54:~ $ sudo apt-get install apache2 -y
```

Nun kann man Versuchen im beliebigen Browser den Webserver zu erreichen. Dafür einfach im Browser https://IP_RASPI eingeben.



Nun installieren wir PHP v7. Und wechseln dann ins Apache root Verzeichnis, entfernen das index.html file und erstellen nun ein .php File. Zudem geben wir dem pi User das Eigentum des neu erstellten File.

```
pi@z54:~ $ sudo apt-get install php libapache2-mod-php -y
pi@z54:~ $ cd /var/www/html
pi@z54:~ $ sudo rm index.html
pi@z54:~ $ sudo nano index.php
pi@z54:~ $ sudo chown pi: index.php
```

Nun kann man das neu erstellte PHP File öffnen und folgenden Inhalt eingeben.

```
pi@z54:~ $ nano index.php
<?php phpinfo(); ?>
```

Nun kann man den Webserver wieder im Browser öffnen und sieht dann alle Informationen zu der installierten PHP Version.

PHP Version 7.3.14-1~deb10u1	
System	Linux z54 4.19.97-v7+ #1294 SMP Thu Jan 30 13:15:58 GMT 2020 armv7l
Build Date	Feb 16 2020 15:07:23
Server API	Apache 2.0 Handler
Virtual Directory Support	disabled
Configuration File (php.ini) Path	/etc/php/7.3/apache2
Loaded Configuration File	/etc/php/7.3/apache2/php.ini
Scan this dir for additional .ini files	/etc/php/7.3/apache2/conf.d
Additional .ini files parsed	/etc/php/7.3/apache2/conf.d/10-opcache.ini, /etc/php/7.3/apache2/conf.d/10-pdo.ini, /etc/php/7.3/apache2/conf.d/20-calendar.ini, /etc/php/7.3/apache2/conf.d/20-ctype.ini, /etc/php/7.3/apache2/conf.d/20-exif.ini, /etc/php/7.3/apache2/conf.d/20-fileinfo.ini, /etc/php/7.3/apache2/conf.d/20-

Nun wechselt man ins Home Verzeichnis. Und installiert die Software für den GrovePi.

```
pi@z54:~ $ cd
pi@z54:~ $ sudop curl -kL dexterindustries.com/update_tools | bash
pi@z54:~ $ sudo git clone https://github.com/DexterInd/GrovePi.git
```

Danach wechseln wir in das Verzeichnis.

```
pi@z54:~ $ cd GrovePi/Script
```

Danach geben wir dem Installationsskript noch die Executive Rechte. Und anschliessend installieren wir GrovePi und i2C

```
pi@z54:~ $ sudo chmod +x install.sh
pi@z54:~ $ sudo ./install.sh
```

Nach einer erfolgreichen Installation rebooten wir den RasPi.

```
pi@z54:~ $ sudo reboot
```

Nach dem Neustart kann überprüfen ob alles richtig installiert wurde. In der Ausgabe sollte dann die Nummer 04 ersichtlich sein.

```
pi@z54:~ $ sudo i2cdetect -y 1
```

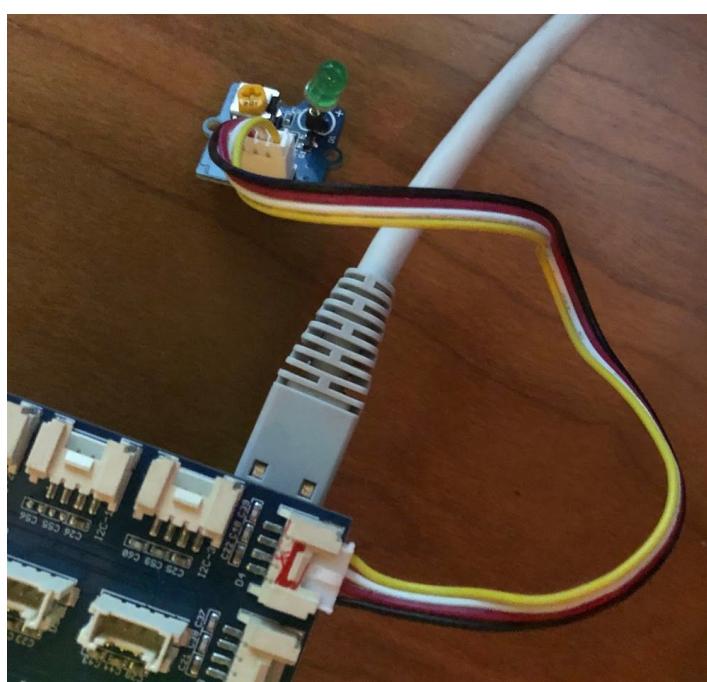
Nun ersetzt man die defekte Library grove.py mit dem alten auf dem [BSCW](#). Danach entfernen wir die defekte Datei und geben dem pi User den Owner Status für den Ordner.

```
pi@z54:~ $ sudo rm ~/GrovePi/Software/Python/grovepi.py
pi@z54:~ $ sudo chown pi: ~/GrovePi/Software/Python/ -R
```

Danach fügen wir die Datei aus dem BSCW via FileZilla oder WINSCP im Ordner hinzu. Zudem geben wir der neuen Datei executive Rechte.

```
pi@z54:~ $ sudo chmod +x ~/GrovePi/Software/Python/grovepi.py
```

Nun muss man am Port D4 eine LED anschliessen.



Nun starten wir das Example Script. Mit CTRL + C kann man das Programm dann wieder beenden.

```
pi@z54:~ $ cd
pi@z54:~ $ cd GrovePi/Software/Python
pi@z54:~ $ sudo python grove_led_blink.py
```

Nun wollen wir das Script mal anschauen, zuerst ändern wir die Zugriffsrechte.

```
pi@z54:~ $ sudo chmod 774 grove_led_blink.py
pi@z54:~ $ sudo chown pi: grove_led_blink.py
```

Ich habe dann das Script so geändert, dass die Lampe nur noch 5 mal blinkt. Zudem habe ich dann noch zwei Script geschrieben zum einschalten und zum ausschalten.

5 Blinken:

```
for x in range(0, 5):
    try:
        #Blink the LED
        digitalWrite(led,1)          # Send HIGH to switch on LED
        print ("LED ON!")
        time.sleep(1)

        digitalWrite(led,0)          # Send LOW to switch off LED
        print ("LED OFF!")
        time.sleep(1)

    except KeyboardInterrupt: # Turn LED off before stopping
        digitalWrite(led,0)
        break
    except IOError:           # Print "Error" if
        communication error encountered
        print ("Error")
```

Ein- und ausschalten:

```
import time
from grovepi import *

# Connect the Grove LED to digital port D4
led = 4

pinMode(led,"OUTPUT")
time.sleep(1)

print ("This example will blink a Grove LED connected to the GrovePi+ on
the port labeled D4.\nIf you're having trouble seeing the LED blink, be
```

```
sure to check the LED connection and the port number.\nYou may also try
reversing the direction of the LED on the sensor.")
print (" ")
print ("Connect the LED to the port labele D4!" )

digitalWrite(led,0)
print ("LED OFF!")

import time
from grovepi import *

# Connect the Grove LED to digital port D4
led = 4

pinMode(led,"OUTPUT")
time.sleep(1)

print ("This example will blink a Grove LED connected to the GrovePi+ on
the port labeled D4.\nIf you're having trouble seeing the LED blink, be
sure to check the LED connection and the port number.\nYou may also try
reversing the direction of the LED on the sensor.")
print (" ")
print ("Connect the LED to the port labele D4!" )

digitalWrite(led,0)
print ("LED OFF!")
```

Nun wechselt man in das Verzeichnis ~/GrovePi/Firmware und machen das Script ausführbar.

```
pi@z54:~ $ sudo chmod +x firmware_update.sh
pi@z54:~ $ sudo ./firmware_update.sh
```

Nun erstellen wir ein Projektordner im Apache Webserver.

```
pi@z54:~ $ cd /var/www/html
pi@z54:~ $ sudo chown -R pi: .
pi@z54:~ $ mkdir Demoprojekt
```

Nun kopieren wir die Dateien aus dem Python-Ordner.

```
pi@z54:~ $ cd
pi@z54:~ $ cd GrovePi/Software/Python
pi@z54:~ $ sudo cp grovepi.py /var/www/html/Demoprojekt
pi@z54:~ $ cp grove_led*.py /var/www/html/Demoprojekt
```

Nun erstellen wir im Demoprojekt Ordner eine LED4.php Datei.

```
<html>
  <head>
    <meta name="viewport" content="width=device-width">
    <title>GrovePi 1/4ber PHP schalten</title>
  </head>
  <body>GrovePi LED 4 schalten:<form method="get"
action="LED4.php">
  <input type="submit" value="LED4 ein" name="LED4ein">
  <input type="submit" value="LED4 aus" name="LED4aus">
</form>
<?php
if(isset($_GET['LED4ein'])){
$val= exec(escapeshellcmd('sudo python grove_led4_ein.py'));
echo "LED4 ein";
}
else if(isset($_GET['LED4aus'])){
$val= exec(escapeshellcmd('sudo python grove_led4_aus.py'));
echo "LED4 aus";
}
?>
</body>
</html>
```

Nun müssen wir dem www-data mit sudo-rechten versehen.

```
pi@z54:~ $ sudo visudo
```

Danach geben wir einfach zuunterst im File folgendes ein.

```
www-data ALL=(ALL) NOPASSWD: ALL
```

Den Demoprojekt ordner nun nur noch für die Gruppe www-data zugänglich machen.

```
pi@z54:~ $ sudo chown -R pi:www-data *
pi@z54:~ $ ls -al
pi@z54:~ $ sudo reboot
```

Danach öffnen wir den Webserver auf einem beliebigen Browser unter folgender Adresse:
http://IP_RASPI/Demoprojekt.

Danach kann man versuchen die Lampe ein und auszuschalten. Das sieht dann etwa so aus:
<https://m126.luis-luescher.com/wp-content/uploads/2020/02/sstg-A37C073D-0A98-4BAE-B628-3C6708AC0CE7.mp4>

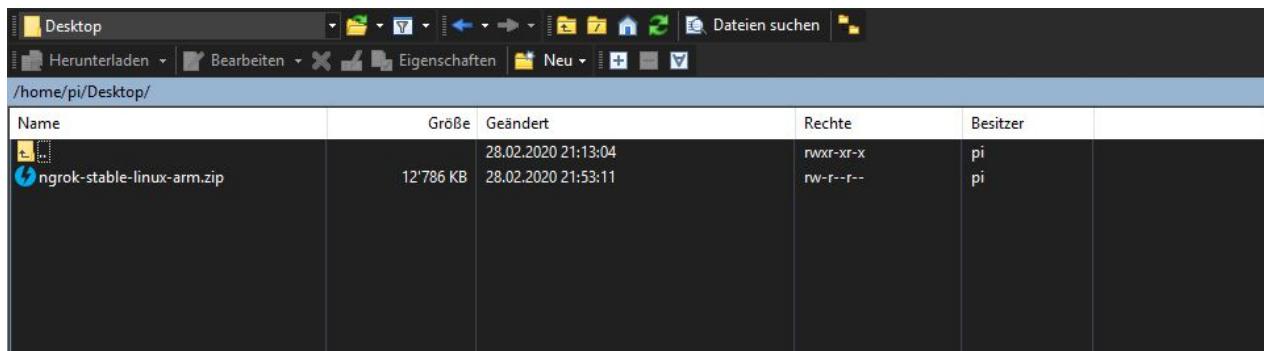
W01 ngrok Tunnel

Installation und Konfiguration

Zu Beginn geht man auf die Website von ngrok (<https://ngrok.com/>)
Hier kann man dann direkt das Linux (ARM) File herunterladen.

The screenshot shows the 'Setup & Installation' section of the ngrok website. It includes four numbered steps: 1. Download ngrok (with a 'Download for Windows' button), 2. Unzip to install (with a terminal command: \$ unzip /path/to/ngrok.zip), 3. Connect your account (with a terminal command: \$./ngrok auth <auth_token>), and 4. Fire it up (with a terminal command: \$./ngrok http 80). Below these steps is a 'How do I...?' section with links for 'Expose a local webserver', 'Test on mobile devices', 'Get a URL that doesn't change', and 'Build a webhook integration'.

Danach kann man das heruntergeladene File via WINSSCP auf den Raspberry Pi spielen.



Nun müssen wir noch das File extrahieren.

```
pi@z54:~ $ unzip ngrok-stable-linux-arm.zip
```

Zudem kann man noch einen Authentifizierungscode speichern, dieser ermöglicht es dann die Tunnel über die Ngrok zu managen. Aus Sicherheitsgründen habe ich den Authentifizierungscode durch den String TOKENCODE ersetzt.

```
pi@z54:~ $ ./ngrok authtoken <TOKENCODE>
```

Theoretisch kann man nun bereits den Tunnel starten.

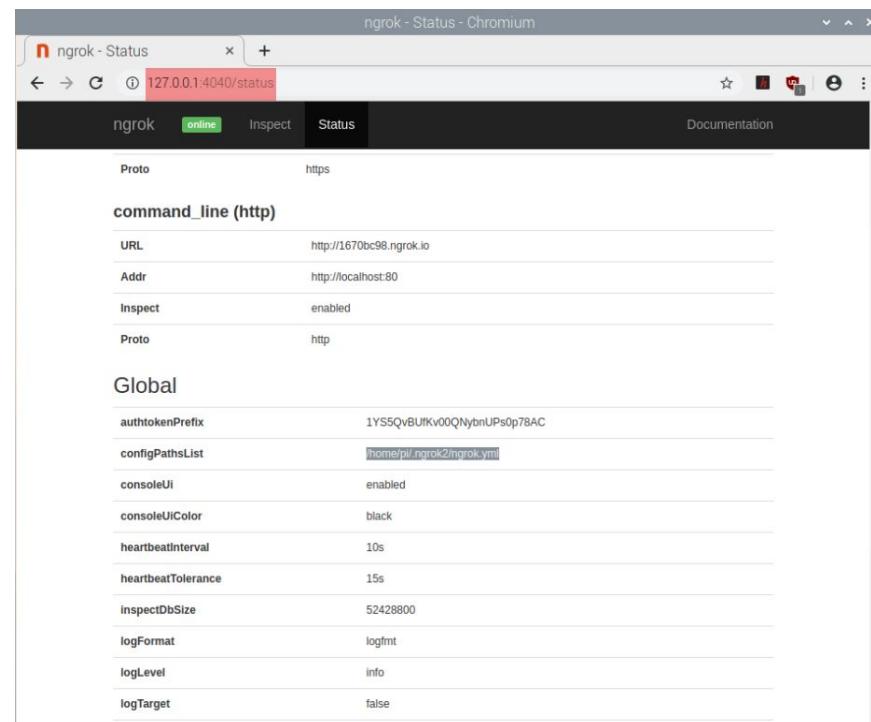
```
pi@z54:~ $ ./ngrok http 80
```

Hier kann man unter dem Punkt "Web Interface" die lokale Adresse für den Manage Bereich sehen. Zudem hat man unter dem Punkt "Forwarding" die öffentliche Adresse die man von überall aus der Welt erreichen kann.

```
ngrok by @inconshreveable

Session Status          online
Account                 luisricardoluescher@gmail.com (Plan: Free)
Version                2.3.35
Region                 United States (us)
Web Interface          http://127.0.0.1:4040
Forwarding             http://1335dl12c.ngrok.io -> http://localhost:80
Forwarding             https://1335dl12c.ngrok.io -> http://localhost:80

Connections            ttl     opn      r1l      rt5      p50      p90
                           0       0       0.00    0.00    0.00    0.00
```



Nun ist es leider so, dass wenn man die Website mittels öffentlicher Adresse öffnet kommt folgende Fehlermeldung:

Failed to complete tunnel connection

The connection to <https://e2e38662.ngrok.io> was successfully tunneled to your ngrok client, but the client failed to establish a connection to the local address **localhost:80**.

Make sure that a web service is running on **localhost:80** and that it is a valid address.

The error encountered was: **dial tcp [::1]:80: connect: connection refused**

Doch dies ist ganz einfach zu lösen. Einfach ein HTML File erstellen....

```
pi@z54:~ $ sudo nano index.html
```

Und einen beliebigen HTML Code einfügen.

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Document</title>
</head>

<body>
    <center>
        <h1>Dies ist ein Test</h1>
        <p>Ich denke es funktioniert lol</p>
        <p>Freundliche Grüsse <br>Luis Lüscher</p>
        <p>Besuche meine <a href="luis-luescher.com">Website</a></p>
    </center>
</body>
</html>
```

Nun kann man den Tunnel starten mit dem Pfad zum HTML File als Parameter.

```
pi@z54:~ $ ./ngrok http file:///home/pi/Desktop/
```

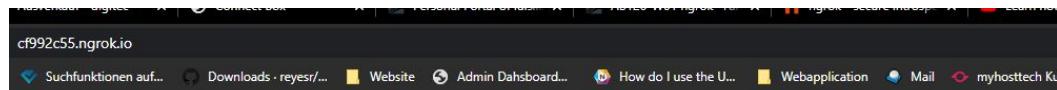
Dann den erstellten öffentlichen Link kopieren....

```
pi@raspberrypi: ~/Desktop
ngrok by @inconshreveable

Session Status          online
Account                luisricardoluescher@gmail.com (Plan: Free)
Version                2.3.35
Region                United States (us)
Web Interface          http://127.0.0.1:4040
Forwarding             http://cf992c55.ngrok.io -> file:///home/pi/Desktop
Forwarding             https://cf992c55.ngrok.io -> file:///home/pi/Desktop

Connections            ttl     opn      rtl      rt5      p50      p90
                        0       0       0.00     0.00     0.00     0.00
```

Im Browser öffnen und nun sieht man die vorhin erstellte Website.



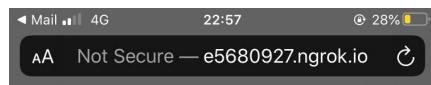
Dies ist ein Test

Ich denke es funktioniert lol

Freundliche Grüsse
Luis Lüscher

Besuche meine [Website](#)

Das gleiche auf dem Handy.



Dies ist ein Test

Ich denke es funktioniert lol

Freundliche Grüsse
Luis Lüscher

Besuche meine [Website](#)

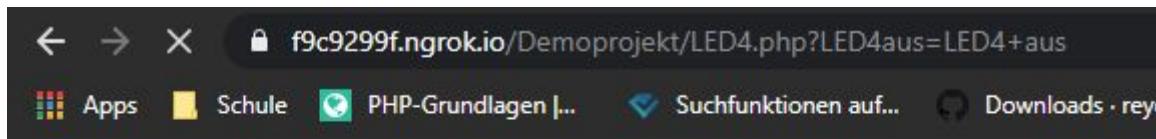


Nun wollen wir via externen Access unser Demoprojekt erreichen und die Lampe ein und ausschalten. Dafür einfach den folgenden befehl ausführen.

```
pi@z54:~ $ ./ngrok http 80
```

Nun müssen wir dann im Browser den Pfad angeben zum Demoprojekt. Folgenden Link im Browser eingeben.

```
<ERZEUGTER LINK>/Demoprojekt/LED4.php  
BSP: https://f9c9299f.ngrok.io/Demoprojekt/LED4.php
```



GrovePi LED 4 schalten:

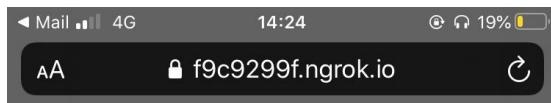
Nun sieht man das Demoprojekt, und es lässt sich auch ohne Probleme ein- und ausschalten.

GrovePi LED 4 schalten:

LED4 ein



Auch vom Handy aus via 4G geht es ohne Probleme.



GrovePi LED 4 schalten:

LED4 ein

Erklärung ngrok Tunnel - How it works

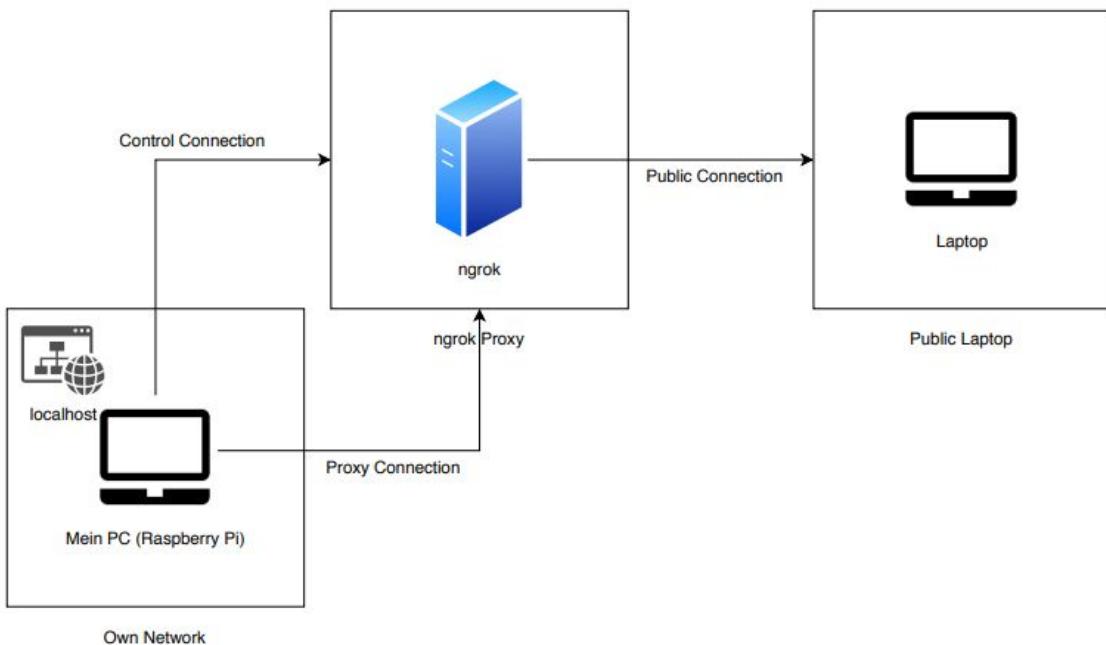
Es ist bereits klar, dass man über die öffentliche Adresse (zB. <http://345f3df54hf.ngrok.com>) an den lokalen Rechner auf Port 80 geht. Aber was ist, wenn ich hinter einer [NAT/Firewall](#) sitze, die eingehenden Verkehr blockiert (ein sehr häufiges Szenario). Veranlasst ngrok [Polling-Anfragen](#), um festzustellen, wann Daten empfangen wurden?

Da ein ngrok-Tunnel immer zuerst auf der Client-Seite initiiert wird, kann er auf diese Weise einen sicheren Kanal mit dem Server aushandeln. Es ist eine wirklich glatte Lösung, um konventionelle Firewall-Konfigurationen zu umgehen. Trotzdem ist es abzuraten ngrok in kritischen Firmennetzwerken zu verwenden, durch die sicher Initialisierungen mit dem Server, wird eine Art Loch durch die Firewall gebohrt. Durch dieses Loch läuft dann der ganze Traffic. So ist dann die Firewall für jeden Angreifer ohne Probleme passierbar.

Dies wird intern dadurch erreicht, dass der Client eine einzige langlebige TCP-Verbindung öffnet, bei der viele logische Sockets innerhalb einer physischen Socket-Verbindung erzeugt

werden. Diese Technik wird als [Stream-Multiplexing](#) bezeichnet. Bei dieser Einrichtung ist keinerlei Abfrage erforderlich, da Client und Server immer noch eine vollständig bidirektionale Kommunikation haben.

Client und Server bleiben dann mit einem [Heartbeat-Mechanismus](#) am Leben, der sicherstellt, dass die Verbindung offen ist und ordnungsgemäß funktioniert und sogar bei einem Fehler oder einer verlorenen/geschlossenen Verbindung wieder hergestellt wird.



W02 Print Server & static IP

Zu Beginn muss der Raspberry Pi eine fixe IP haben, dazu siehe folgende [Anleitung](#).

1. Installieren Sie CUPS und richten es anschließend ein.
2. Installieren Sie den Drucker-Treiber, falls notwendig.
3. Richten Sie den Drucker unter CUPS ein.
4. Drucken Sie eine Datei auf der Kommandozeile aus.

Zuerst führen wir ein Update der Paketlisten und ein Upgrade des Systems durch.

```
pi@z54:~ $ sudo apt-get update
pi@z54:~ $ sudo apt-get upgrade
```

Dann installieren wir CUPS. Der Parameter “-f” forced den Installationprozess.

```
pi@z54:~ $ sudo apt-get -f install cups
```

Es kann sein das der Download nicht funktioniert, dann einfach das Repository ändern.

```
pi@z54:~ $ sudo nano /etc/apt/sources.list.d/raspi.list
```

Und in diesem File muss man auch noch das Repository anpassen.

```
pi@z54:~ $ sudo nano /etc/apt/sources.list
```

In den File's muss man **nur** den **Link** anpassen. Die Links zu den Repositorys sind auf dieser [Website](#) einsehbar.

Nun installiert man die Treiber für sonstige Farb-Tintendrucker.

```
pi@z54:~ $ sudo apt-get install printer-driver-gutenprint
```

- printer-driver-gutenprint (enthält viele Treiber für sonstige Farb-Tintendrucker)

Jetzt kann der Dienst gestartet werden.

```
pi@z54:~ $ sudo /etc/init.d/cups start
```

Wenn man andere Benutzer für die Druckerkonfiguration berechtigen wollen, muss dieser benutzer in der Gruppe lpadmin sein.

```
pi@z54:~ $ sudo usermod -aG lpadmin pi
```

Es bietet sich ausserdem an, einen extra Druck-Benutzer anzulegen.

```
pi@z54:~ $ sudo useradd -c Druckuser -M -s /bin/false Drucker
pi@z54:~ $ sudo usermod -aG lpadmin Drucker
pi@z54:~ $ sudo passwd Drucker
```

Die Option “-M” legt dem Benutzer kein Homeverzeichnis an; die Option “-s /bin/false” verhindert das Einloggen auf der Shell, so dass das Passwort nicht sehr stark sein muss. (Gut für die Familie zum Merken) “-c Druckuser” ist einfach nur ein Kommentar.

Nun das Konfigurationsfile mit dem Nano editor öffnen.

```
pi@z54:~ $ sudo nano /etc/cups/cupsd.conf
```

Danach folgende Änderungen im File machen. (**Fett markiert**)

```
# Only listen for connections from the local machine.
# Listen localhost:631
# Listen /run/cups/cups.sock
#Allow remote access
Port 631
# Web interface setting...
```

WebInterface Yes

Restrict access to the server...

```
<Location />
  Require user @SYSTEM
  Order allow,deny
  Allow @local
</Location>
```

Restrict access to the admin pages...

```
<Location /admin>
  Order allow,deny
  Allow @Local
  Require user @SYSTEM
</Location>
```

Restrict access to configuration files...

```
<Location /admin/conf>
  AuthType Default
  Require user @SYSTEM
  Order allow,deny
  Allow @Local
</Location>
```

Restrict access to log files...

```
<Location /admin/log>
  AuthType Default
  Require user @SYSTEM
  Order allow,deny
</Location>
```

#Added by Luis Luescher

```
<Location /classes>
  Order allow,deny
  Allow @Local
  Require user @SYSTEM
</Location>
```

```
<Location /help>
```

```
  Order allow,deny
  Allow @Local
  Require user @SYSTEM
</Location>
```

```
<Location /jobs>
```

```
Order allow,deny  
Allow @Local  
Require user @SYSTEM  
</Location>
```

```
<Location /printers>  
Order allow,deny  
Allow @Local  
#Require user @SYSTEM  
#Sollte auskommentiert werden, damit Windows den Server findet  
</Location>
```

Und danach noch Cups neu starten.

```
pi@z54:~ $ sudo /etc/init.d/cups restart
```

Nun kann man die Weboberfläche öffnen via <http://<fixe-IP-Adresse>:631>.
Authentifizierung via “pi” und Benutzerpasswort oder “Drucker” und Benutzerpasswort
(vorhin angelegter User).

The screenshot shows the CUPS 2.2.10 web interface. At the top is a dark header bar with the CUPS.org logo and links for Home, Verwaltung, Klassen, Hilfe, Aufträge, and Drucker. Below this is a light-colored main content area divided into three columns:

- CUPS für Benutzer**: Contains links for Überblick über CUPS, Befehlszeilen Druck und Optionen, and Benutzerforum.
- CUPS für Administratoren**: Contains links for Drucker und Klassen hinzufügen, Betriebs-Richtlinie festlegen, Benutzung von Netzwerk-Druckern, and cupsd.conf Referenz.
- CUPS für Entwickler**: Contains links for Einführung in die CUPS Programmierung, CUPS API, Filter und Backend Programmierung, HTTP und IPP APIs, and Entwicklerforum.

At the bottom of the page is a footer bar with the text "CUPS und das CUPS Logo sind Warenzeichen der Apple Inc. Copyright © 2007-2018 Apple Inc. Alle Rechte vorbehalten."

Nun kann man eigentlich komplett auf das Terminal verzichten, da die meisten Konfigurationen im Webinterface möglich sind. So kann man unter dem Reiter “Verwaltung” die Konfigurationsdatei anpassen.

The screenshot shows the CUPS Management page. The top navigation bar includes links for Startseite, Verwaltung (which is highlighted in red), Klassen, Hilfe, Aufträge, and Drucker. The main content area is titled "Verwaltung" and contains several sections:

- Drucker**: Buttons for Drucker hinzufügen, Neue Drucker suchen, and Drucker verwalten.
- Klassen**: Buttons for Klasse hinzufügen and Klassen verwalten.
- Aufträge**: Button for Aufträge verwalten.
- Server**: A section with "Konfigurationsdatei bearbeiten", "Zugriffsprotokoll anzeigen", "Fehlerprotokoll anzeigen", and "Seitenprotokoll anzeigen". It also includes "Server-Einstellungen:" with an "Erweitert" link and several checkboxes for network and security options.

Nun müssen wir zuerst das Drucken aus dem Internet aktivieren. Dazu unter dem Punkt "Server-Einstellung" den Hacken bei "Drucken aus dem Internet zulassen" setzen. Um das ganze zu Speichern klicken wir dann auf Einstellung ändern.

Nun werden wir den Drucker via USB an den RasPi anschliessen. Dann kann man unter dem Reiter "Verwaltung" auf den Button "Drucker hinzufügen" klicken.

CUPS.org Startseite Verwaltung Klassen Hilfe Aufträge Drucker

Verwaltung

Drucker

Drucker hinzufügen Neue Drucker suchen Drucker verwalten

Nun sollte der Drucker erscheinen in der Auswahl. Diesen Auswählen und dann mit "Weiter" bestätigen.

CUPS.org Startseite Verwaltung Klassen Hilfe Aufträge Drucker

Drucker hinzufügen

Drucker hinzufügen (Schritt 1/5)

- Lokale Drucker:** CUPS-BRF (Virtual Braille BRF Printer)
 VNC Remote Printer (VNC Printer)
 Brother DCP-195C (Brother DCP-195C)

Gefundene Netzwerkdrucker:

- Andere Netzwerkdrucker:** Backend Error Handler
 Internet Printing Protocol (ipp)
 Internet Printing Protocol (ipp)
 Internet Printing Protocol (http)
 LPD/LPR-Host oder -Drucker
 AppSocket/HP JetDirect
 Internet Printing Protocol (https)

Weiter

Danach kann man dem Drucker einen Namen geben. Wichtig ist, dass man unter dem Punkt "Freigabe" den Haken bei "Drucker im Netzwerk freigeben" macht.

CUPS.org Startseite Verwaltung Klassen Hilfe Aufträge Drucker

Drucker hinzufügen

Drucker hinzufügen (Schritt 3/5)

Name: Brother_DCP-195C
(Darf alle druckbaren Zeichen außer "/", "#", und Leerzeichen enthalten)

Beschreibung: Brother DCP-195C
(Menschenlesbare Beschreibung wie etwa "HP LaserJet mit Duplexer")

Ort: 1OG Zimmer
(Menschenlesbarer Ort wie etwa "Labor 1")

Verbindung: usb://Brother/DCP-195C?serial=BROB1F154094|Brother DCP-195C

Freigabe: Drucker im Netzwerk freigeben

Im letzten Schritt muss man die Druckertreiber auswählen. Normalerweise sucht CUPS nach dem besten und schlägt diesen auch vor. Es kann aber sein, dass dies nicht so ist, dann kann man einen eigenen Treiber hochladen, via Webinterface oder man kann sich im Terminal den Treiber herunterladen. Den Schritt dann mit dem Klick auf den Button "Drucker hinzufügen" beenden.

Drucker hinzufügen

Drucker hinzufügen (Schritt 5/5)

Name: Brother_DCP-195C

Beschreibung: Brother DCP-195C

Ort: 1OG Zimmer

Verbindung: usb://Brother/DCP-195C?serial=BROB1F154094|Brother DCP-195C

Freigabe: Drucker im Netzwerk freigeben

Hersteller: Brother

Modell:
Brother DCP-195C CUPS (en)
Brother DCP-1200 - CUPS+Gutenprint v5.3.1 (en)
Brother DCP-8045D - CUPS+Gutenprint v5.3.1 (en)
Brother HL-4Ve - CUPS+Gutenprint v5.3.1 (en)
Brother HL-10h - CUPS+Gutenprint v5.3.1 (en)
Brother HL-10V - CUPS+Gutenprint v5.3.1 (en)
Brother HL-630 - CUPS+Gutenprint v5.3.1 (en)
Brother HL-660 - CUPS+Gutenprint v5.3.1 (en)
Brother HL-760 - CUPS+Gutenprint v5.3.1 (en)

Oder PPD-Datei bereitstellen: Keine ausgewählt

Nun kann man über das Interface eine Testseite drucken, um zu schauen ob alles so funktioniert, wie es sollte.

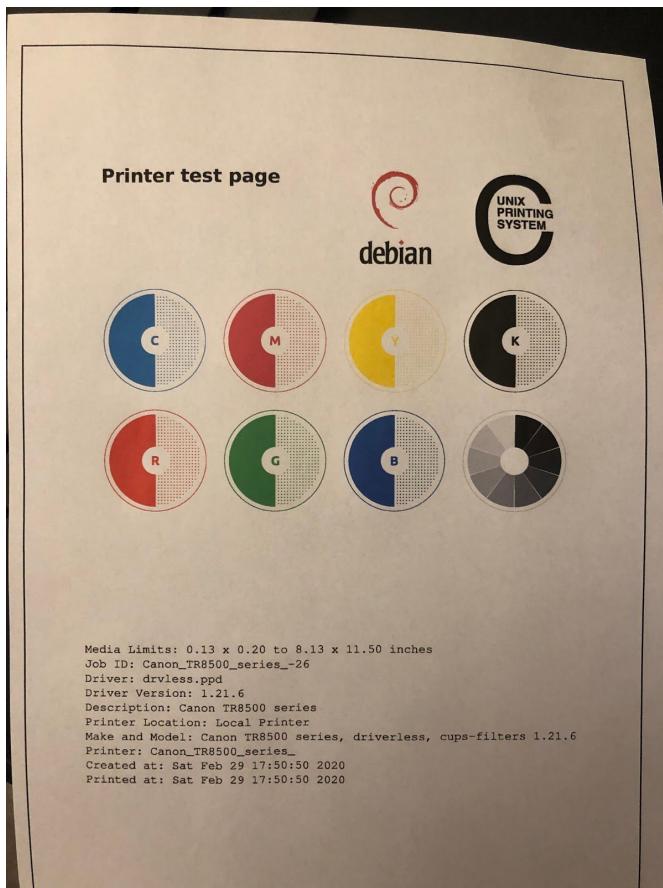
Canon_TR8500_series_

Canon_TR8500_series_ (Beschäftigt, Aufträge annehmen, Netzwerk)

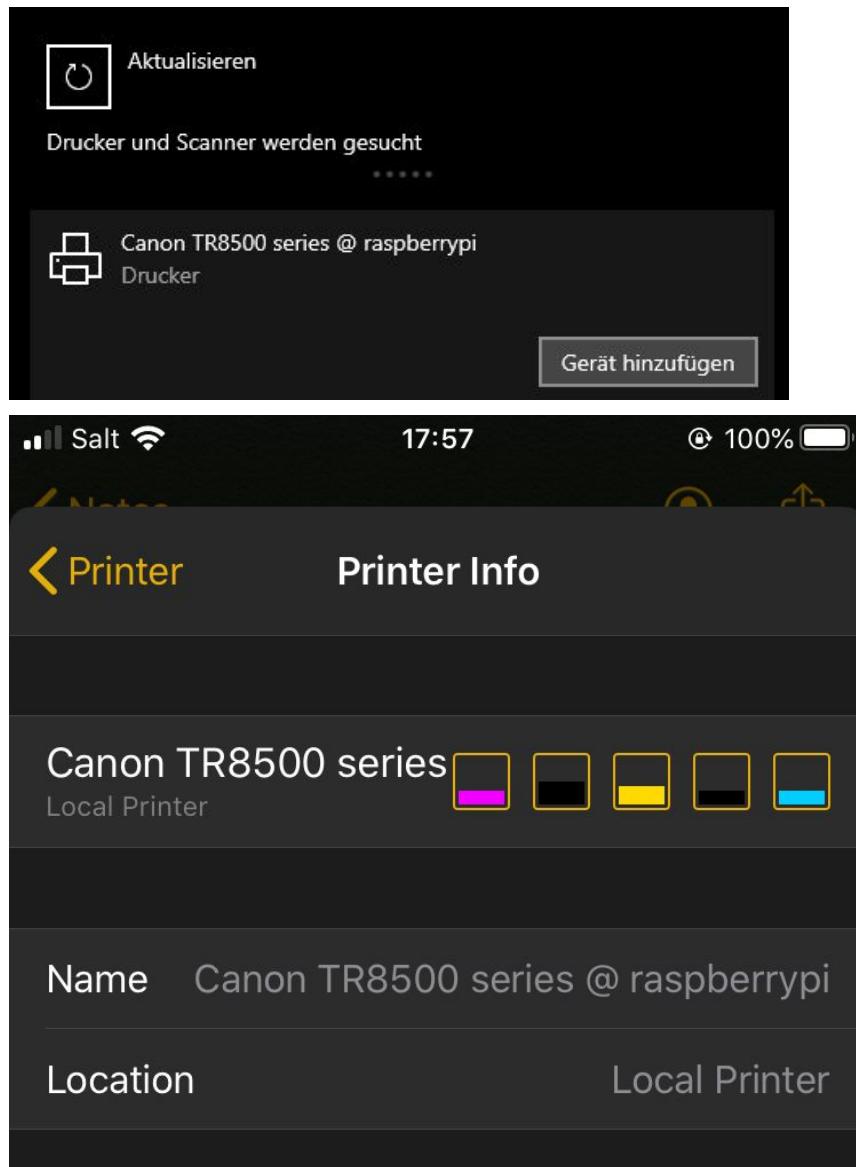
Wartung	▼	Verwaltung	▼
Wartung	:	Canon TR8500 series	
Testseite drucken	:	Local Printer	
Drucker anhalten	:	Canon TR8500 series, driverless, cups-filters 1.21.6 (farbig, 2-seitiges Drucken)	
Aufträge ablehnen	:	ipp://CD642B00000.local:631/ipp/print	
Alle Aufträge verschieben	:	job-sheets=none, none media=iso_a4_210x297mm sides=one-sided	
Alle Aufträge abbrechen	:		
Standardauflösungen:	:		

Aufträge

So würde dann die entsprechende Testseite aussehen.



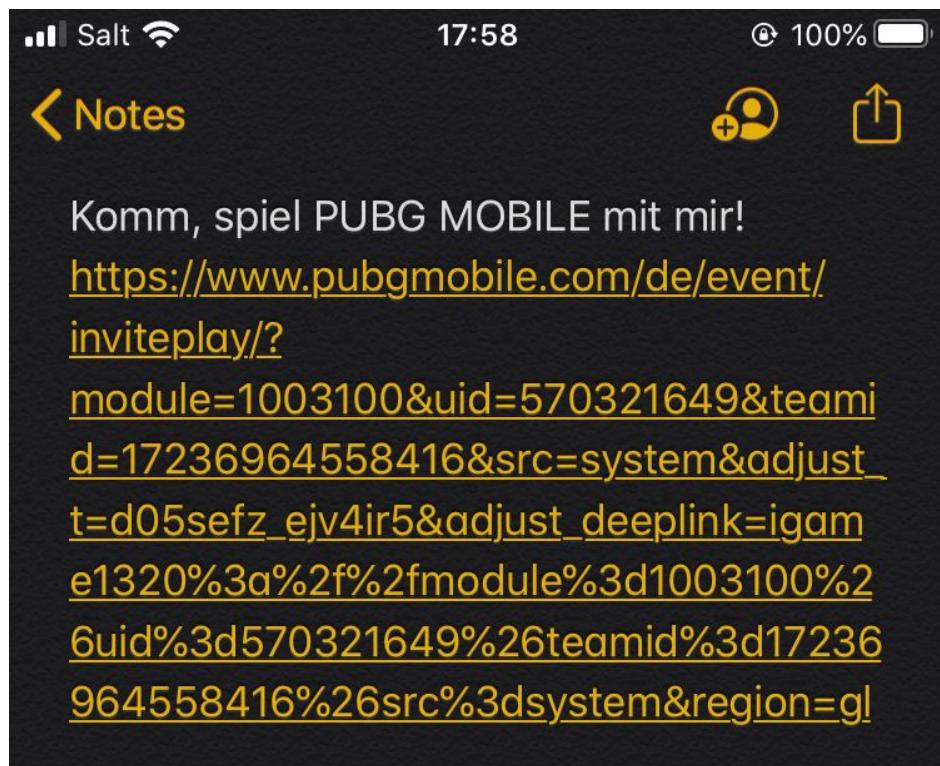
Wenn die Testseite erfolgreich gedruckt wurde, kann man sich dann mit dem Drucker verbinden. Auf seinem Handy oder auf seinem PC.



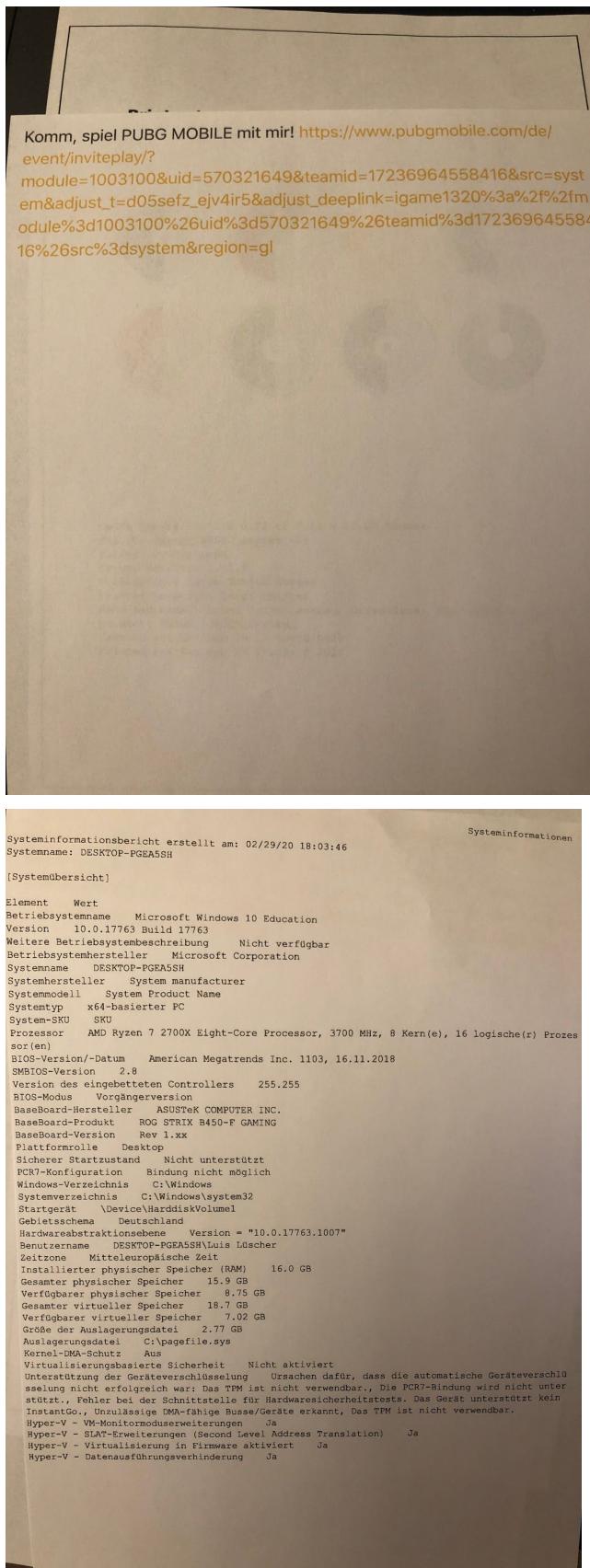
Wichtig ist halt, dass die jeweiligen Clients im selben Netzwerk sind wie die Drucker.
Das tolle ist nun kann man alle ausgedruckten Dokumente an einem zentralen Punkt sehen.

Jobs listed in descending order.						
ID	Name	Benutzer	Größe	Seiten	Status	Steuerung
Canon_TR8500_series_26	Test Page	Drucker	1k	1	beendet am Sat Feb 29 17:51:54 2020	<button>Auftrag neu drucken</button>
Canon_TR8500_series_27	Komm, spiel PUBG MOBILE mit mir! https://www.pubgmobile.com/de/event ...	mobile	11k	1	beendet am Sat Feb 29 17:57:17 2020	<button>Auftrag neu drucken</button>
Canon_TR8500_series_29	Systeminformationen 8.0	NT-AUTORITÄT\Lokaler Dienst	70k	1	beendet am Sat Feb 29 18:04:00 2020	<button>Auftrag neu drucken</button>
Canon_TR8500_series_30	tbz gut diese	mobile	1844k	3	beendet am Sat Feb 29 18:07:28 2020 "Unable to cancel print job."	<button>Auftrag neu drucken</button>

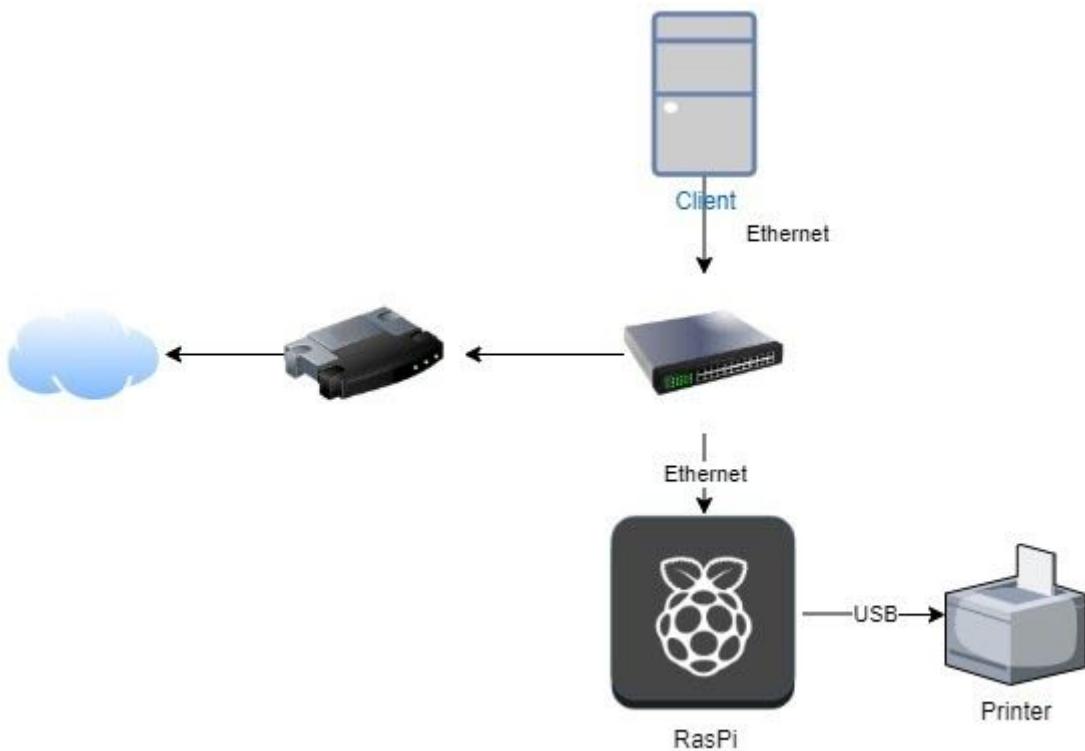
Hier unten sind die jeweiligen Dokumente einsehbar.



Luis Ricardo Lüscher
M126 - Peripheriegeräte im Netzwerkbetrieb einsetzen



Dies ist die ausgedruckte Systeminformation.



Einfacher Aufbau der Umgebung. Mit RasPi der als print-Server fungiert und Client der etwas ausdrucken möchte.

AirPrint:

AirPrint ist eine von Apple bereitgestellte Software, die es ermöglicht Apple-Geräte drahtlos über ein WLAN-Netzwerk mit geeigneten Druckern zu verbinden um Emails, Fotos oder auch einfach nur Dokumente auszudrucken. Ist ein Gerät mit dieser Software kompatibel, spricht man von einem AirPrint-Drucker. Drahtloses Drucken ist natürlich keine Neuheit, auch vor Services wie AirPrint gab es WLAN-Drucker, die das ermöglichen. Was hebt AirPrint also von herkömmlichen Drahtlosdruckern ab?

Die größte Besonderheit bei AirPrint-Druckern, im Vergleich zu gewöhnlichen WLAN-Druckern, besteht darin, dass zum Drucken mit AirPrint keine Treiber benötigt werden. Diese müssen normalerweise installiert werden, um ein verwendetes Endgerät (z.B. Computer, Smartphone, Tablet) mit dem Drucker zu koppeln.

Diese Notwendigkeit fällt mit der Nutzung von AirPrint komplett weg. Hier kommuniziert der Computer, das Tablet oder das Smartphone direkt mit dem Drucker – ohne einen erforderlichen Zwischenschritt. Der Workflow mit AirPrint wird somit schnell und unkompliziert gestaltet.

W04 RPI Monitor

Um den RPi-Monitor zu installieren, fügen wir zuerst den Public Key hinzu, somit müssen wir dem Repository zuerst vertrauen.

```
pi@z54:~ $ sudo apt-get install dirmngr
```

```
pi@z54:~ $ sudo apt-key adv --recv-keys --keyserver keyserver.ubuntu.com 2C0D3C0F
```

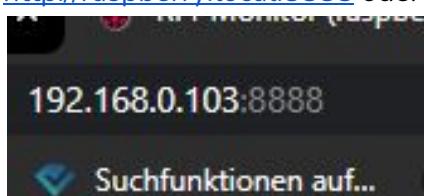
Mit folgendem Kommando fügen wir RPI-Monitor zur der Liste der Repositories.

```
pi@z54:~ $ sudo wget http://goo.gl/vewCLL -O /etc/apt/sources.list.d/rpimonitor.list
```

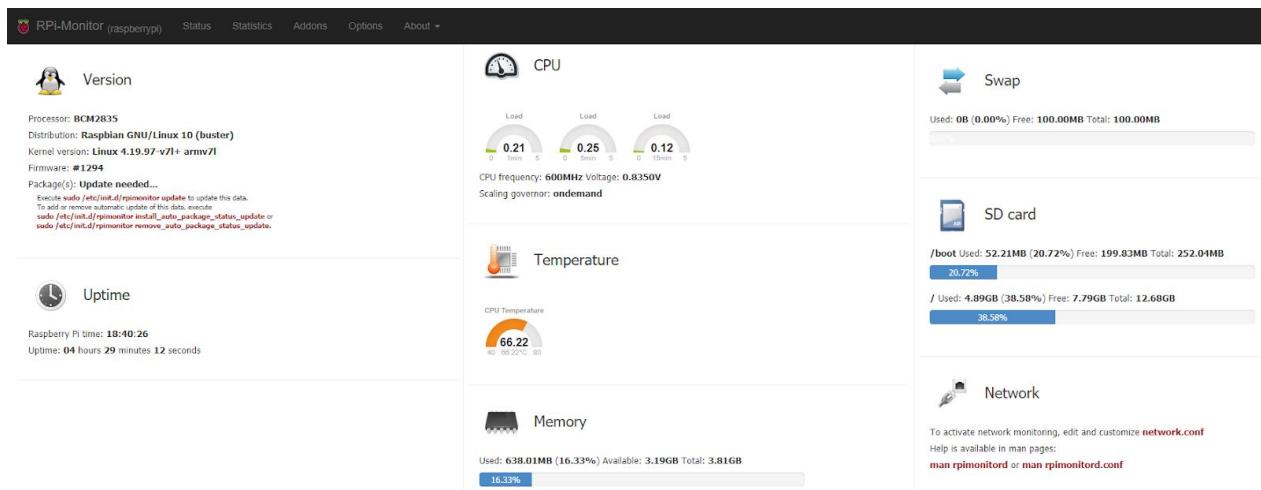
Nun installieren wir den RPi-Monitor, dafür folgendes Kommando verwenden.

```
pi@z54:~ $ sudo apt-get update  
pi@z54:~ $ sudo apt-get install rpimonitor
```

Nun ist es so das man entweder über folgende Links auf den RPi-Monitor zugreifen kann:
<http://raspberry.local:8888> oder <http://<IP-RASPI>:8888>.



Wenn man dann die Seite öffnet sieht es folgendermaßen aus.



Hie kann man alle wichtigen Informationen bezüglich des RasPi sehen, wie zB. momentaner CPU-Usage oder auch die CPU-Temperatur.

Nun fügen wir unter dem Punkt Addon ein Addon hinzu nämlich "Shellinabox". Shellinabox ist eigentlich genau, dass was der Name sagt. Es ist eine Shell in einem Web Gui. Nach unserer Konfiguration wird es möglich sein, über den Addons Bereich direkt auf dem Terminal, also der Shell zu arbeiten ohne das man extra ein Terminal Fenster öffnen muss. Zu Beginn installieren wir Shellinabox

```
pi@z54:~ $ sudo apt-get install shellinabox
```

Nun kann man einen beliebigen Browser öffnen und über folgenden Link auf die Shell in a Box zugreifen: https://IP_RASPI:4200/. Wichtig ist hierbei, dass man über das HTTPS

Protokoll geht und nicht nur über HTTP. Nun kann einfach über den Browser im Terminal also in der Shell arbeiten.

Damit wir alles lokaler haben, werden wir nun die Shell in a Box als Addon ins RPi Monitor Tool einbauen.

Dazu öffnen wir mit einem beliebigen Editor die Konfigurationsdatei.

```
pi@z54:~ $ sudo nano /etc/rpimonitor/data.conf
```

Nun kann man einfach den untenstehenden Code in sein Config-File einfügen, evtl. muss man die IP Adresse anpassen.

```
web.friends.1.name=NAS Synology
web.friends.1.link=http://192.168.0.100/
web.friends.2.name=NAS Netgear
web.friends.2.link=http://192.168.0.102/
#web.friends.3.name=Raspberry Pi
#web.friends.3.link=http://192.168.0.38/

web.page.icon='img/logo.png'
web.page.menutitle='RPi-Monitor <sub>('+data.hostname+')</sub>'
web.page.pagetitle='RPi-Monitor ('+data.hostname+')'

web.status.1.name=Raspberry Pi
#web.status.2.name=Home
web.statistics.1.name=Raspberry Pi
#web.statistics.2.name=page 2

web.addons.1.name=Addons
web.addons.1.addons=about
web.addons.1.title=Shellinabox
#web.addons.1.addons=shellinabox
web.addons.1.addons=custom
web.addons.1.showtitle=true
web.addons.1.url=https://192.168.0.103:4200/
web.addons.1.allowupdate=false

#web.addons.2.name=Addons example
#web.addons.2.addons=example
#web.addons.2.showTitle=0

#web.addons.3.name=Webcam - Hawkeye
#web.addons.3.showTitle=0
#web.addons.3.addons=custom
#web.addons.3.url=http://0:8000/

#web.addons.4.name=Custom addons
```

```
#web.addons.4.addons=custom
#web.addons.4.showTitle=0
#web.addons.4.url=/addons/custom/custominfo.html

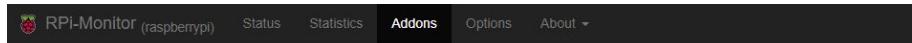
#web.addons.5.name=Top3
#web.addons.5.addons=top3

#include=/etc/rpimonitor/template/example.justgage.conf
#include=/etc/rpimonitor/template/example.badge_and_label.conf
#include=/etc/rpimonitor/template/example.progressbar.conf
#include=/etc/rpimonitor/template/example.visibility.conf
include=/etc/rpimonitor/template/example.alert.conf
#include=/etc/rpimonitor/template/example.interval.conf

include=/etc/rpimonitor/template/version.conf
include=/etc/rpimonitor/template/uptime.conf
include=/etc/rpimonitor/template/cpu.conf
include=/etc/rpimonitor/template/temperature.conf
include=/etc/rpimonitor/template/memory.conf
include=/etc/rpimonitor/template/swap.conf
include=/etc/rpimonitor/template/sdcard.conf
include=/etc/rpimonitor/template/network.conf

#include=/etc/rpimonitor/template/printer.conf
#include=/etc/rpimonitor/template/storage.conf
#include=/etc/rpimonitor/template/services.conf
#include=/etc/rpimonitor/template/wlan.conf
#include=/etc/rpimonitor/template/dht11.conf
#include=/etc/rpimonitor/template/entropy.conf
```

Wenn man möchte kann man nun den RasPi neu starten. (Wird empfohlen um die ganze Funktionalität zu testen) Danach sollte es ungefähr so Aussehen unter dem Punkt Addons.



Addons

```
raspberrypi login: pi
Password:
Last login: Sat Feb 29 21:08:38 CET 2020 from 192.168.0.157 on pts/1
Linux raspberrypi 4.19.97-v7l+ #1294 SMP Thu Jan 30 13:21:14 GMT 2020 armv7l

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.

Wi-Fi is currently blocked by rfkill.
Use raspi-config to set the country before use.

pi@raspberrypi:~ $ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
        inet 127.0.0.1/8 scope host lo
            valid_lft forever preferred_lft forever
        inet6 ::1/128 scope host
            valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc mq state UP group default qlen 1000
    link/ether dc:a6:32:48:52:98 brd ff:ff:ff:ff:ff:ff
        inet 192.168.0.103/24 brd 192.168.0.255 scope global noprefixroute eth0
            valid_lft forever preferred_lft forever
        inet6 fe80::6e66:5497:50bd:6441/64 scope link
            valid_lft forever preferred_lft forever
3: wlan0: <BROADCAST,MULTICAST> mtu 1500 qdisc noop state DOWN group default qlen 1000
    link/ether dc:a6:32:48:52:9b brd ff:ff:ff:ff:ff:ff
pi@raspberrypi:~ $
```

W05 Netttime NTP

Nun ist die Aufgabe den RasPi so zu konfigurieren, das er die Zeit mit einem Zeitserver abgleicht. Im neuen Raspian Stretch muss der Dienst NTPD installiert werden.

```
pi@z54:~ $ sudo apt-get update && sudo apt-get -y install ntp
```

Nach der erfolgreichen Installation kann man die Konfigurationsdatei öffnen.

```
pi@z54:~ $ sudo nano /etc/ntp.conf
```

Nun muss man die Debian NTP-Server auskommentieren. Mittels #

```
#pool 0.debian.pool.ntp.org iburst
#pool 1.debian.pool.ntp.org iburst
#pool 2.debian.pool.ntp.org iburst
#pool 3.debian.pool.ntp.org iburst
```

Danach kann man auf dieser Website den nächsten NTP-Server finden.

```
server 0.ch.pool.ntp.org
```

```
server 1.ch.pool.ntp.org
server 2.ch.pool.ntp.org
server 3.ch.pool.ntp.org
```

Danach muss man den Raspi neu starten

```
pi@z54:~ $ sudo reboot
```

Danach kann man mit folgenden Befehl die aktuellen NTP-Server anzeigen lassen.

```
pi@z54:~ $ ntpq -pn
```

Folgenden Output habe ich erhalten.

remote	refid	st	t	when	poll	reach	delay	offset	jitter
192.168.0.255	.BCST.	16	B	-	64	0	0.000	0.000	0.001
+45.91.92.111	185.134.196.169	2	u	58	64	177	37.679	9.895	5.939
81.94.123.17	85.158.25.74	2	u	64	64	143	12.302	-3.463	10.907
+212.25.1.1	194.242.34.149	2	u	57	64	173	9.844	3.157	3.385
*5.148.175.134	131.188.3.222	2	u	55	64	177	10.604	2.873	4.222

Das Sternchen (*) steht für die aktuelle Zeitquelle. Das Pluszeichen (+) steht für Source selected, included in final set (Für Mittelwertberechnung einbezogen). Insofern diese Zeichen noch nicht sichtbar sind, muss man noch ein bisschen warten. Denn dies ist ein Hinweis darauf, dass der NTP Dienst noch nicht genug lang gestartet ist. Der Titel remote bezeichnet den Zeitserver aus der ntp.conf Datei. Redif gibt an, woher der Remoteserver seine Zeit hat. Die Zeile st gibt an welcher Stratum Level der Server hat. NTP nutzt ein hierarchisches System verschiedener Strata. Als Stratum 0 bezeichnet man das Zeithormal, beispielsweise eine Atomuhr oder eine Funkuhr (Zeitzeichenempfänger via GNSS oder DCF77). Die unmittelbar mit ihm gekoppelten NTP-Server heißen Stratum 1. Jede weitere abhängige Einheit erhält bei der Bezeichnung eine höhere Nummer (Stratum 2, Stratum 3 usw.). Die NTP-Software auf Stratum 1, Stratum 2, Stratum 3 usw. ist zugleich Client des darüber liegenden Stratums als auch Server des darunter liegenden Stratums, sofern eines existiert. T steht für den Servertyp => l = local, u = unicast (most common), m = multicast, b = broadcast und - = netaddr. Die Werte von When geben an, vor wie vielen Sekunden eine Antwort erhalten wurde. Poll bezeichnet den Abruf Intervall in Sekunden. Reach ist wie oft der Server erreicht wurde (377 heisst zB. , dass der Server die letzten 8 Abfragen erfolgreich waren). Delay ist die Durchschnittszeit für Antworten in Millisekunden. Offset ist die Zeitdifferenz zwischen Client und Server in Millisekunden. Jitter ist wie stark die abgefragte Zeit zwischen den Anfragen schwankt in Millisekunden.

Nun wenn man den RasPi mit der Zeit synchronisiert hat, kann man die Informationen mit seinem Netzwerk teilen, dazu müssen wir den NTP-Server “öffnen”.

```
pi@z54:~ $ sudo nano /etc/ntp.conf
```

Nun suchen wir nach dem Eintrag mit restrict und passen den Standartgateway an.

```
restrict 192.168.0.0 mask 255.255.255.0
```

Zudem muss man folgende Zeilen entkommentieren oder zuunterst im File hinzufügen, wenn sie nicht vorhanden sind.

```
disable auth  
broadcastclient
```

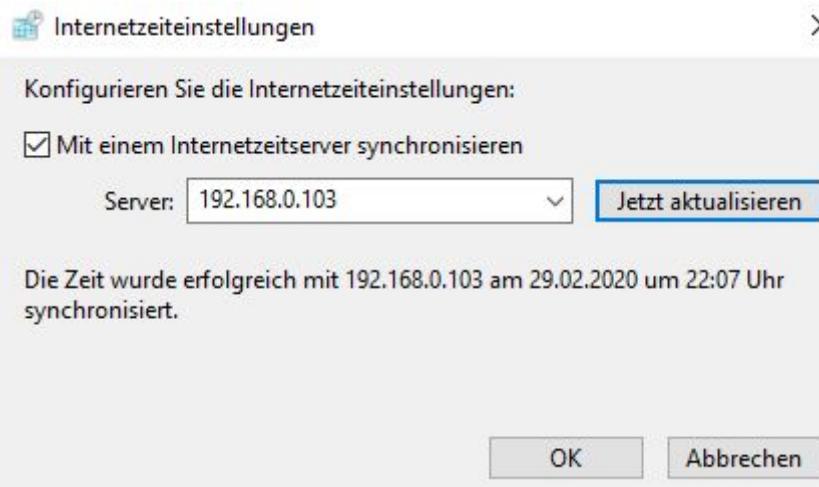
Zudem den Broadcast Eintrag #broadcast 192.168.123.255 auf das eigene Netzwerk anpassen.

```
broadcast 192.168.0.255
```

Nun alles speichern und dann den Dienst neustarten

```
pi@z54:~ $ sudo /etc/init.d/ntp restart
```

Nun kann man einen Client anpassen und mit dem NTP-Server verbinden. Unter Windows folgender pfad für die Einstellung: Systemsteuerung => Zeit, Sprache und Region => Datum und Uhrzeit => Internetzeit => Einstellung ändern (Administratorenrechte nötig!).



Dann kann man die IP des RasPi eingeben und auf Jetzt aktualisieren klicken. Wenn der Vorgang erfolgreich war, gibt es eine dementsprechende Meldung im GUI.

Fragen anhand der ntpq -pn Ausgabe

Welcher ntp-Server wird vom RasPi verwendet (hostname) ?

Mein RasPi verwendet den Server 5.148.175.134 als NTP-Server. Siehe nslookup

```
C:\Users\Luis>nslookup 5.148.175.134
Server:  dns.google
Address: 8.8.8.8

Name:   skarabae.draxinusom.ch
Address: 5.148.175.134
```

Erkennbar am * (Sternchen), dieses steht für den bevorzugten NTP-Server.

Welchen Stratum-Level hat dieser? Wo können Sie das ablesen?

Der NTP-Server hat ein Stratum-Level 2, erkennbar unter dem Punkt st und dann die entsprechende Nummer in diesem Fall 2.

Was bedeuten die Spalten delay und jitter?

Delay ist die Durchschnittszeit für Antworten in Millisekunden.

Jitter ist wie stark die abgefragte Zeit zwischen den Anfragen schwankt in Millisekunden.

Was ist eine Schaltsekunde? Warum braucht man diese?

Die Ortszeiten in Zeitzonen weltweit basieren auf der Koordinierten Weltzeit (UTC). Diese wird durch zwei Komponenten bestimmt:

Die Internationale Atomzeit (TAI) – ein sehr genauer Zeitstandard, der mithilfe von über 400 Atomuhren berechnet wird.

Die Universal Time (UT1) – eine Sonnenzeit, die anhand der tatsächlichen Erdrotation bestimmt wird.

Nun dreht sich die Erde immer langsamer, die Tage werden dadurch länger, doch unsere Atomuhren bleiben über Jahrmillionen nahezu konstant. So hinkt die UT1 der TAI immer mehr hinterher, momentan beträgt der Unterschied etwa 37 Sekunden. Um sicherzustellen, dass unsere Uhren die tatsächliche Tageslänge möglichst exakt widerspiegeln – dass also der Unterschied zwischen UT1 und UTC nicht zu groß wird – fügen wir zur UTC etwa alle anderthalb Jahre eine Sekunde hinzu.

Beschreiben Sie in eigenen Worten wie NTP funktioniert.

In Netzwerken und in Computern mit zeitkritischen Aufgaben ist eine genaue Zeit unerlässlich. Schon deshalb wurden sehr früh Mechanismen entwickelt, wie vernetzte Computer die Zeit untereinander austauschen können. NTP ist ein hierarchisches Protokoll über das Zeit-Server untereinander eine gemeinsame Zeit ermitteln können. Als Port wird 123 verwendet.

Auf Port 13 kann ein Server mittels "daytime" seine Zeit als ASCII-Klartext zur Verfügung stellen. Eine andere Möglichkeit ist der Port 37 über "time". Dieser liefert die verstrichenen Sekunden seit 1.1.1900 0 Uhr als 32-Bit-Binärwert zurück. Dieser Wert ist allerdings nur sekundengenau.

Diese beiden simplen Verfahren haben jedoch noch einen weiteren großen Nachteil. Sie berücksichtigen nicht die Datenpaketlaufzeit zum Ziel. Bis die Angabe der Zeit beim Empfänger "eintrifft" ist sie veraltet. Außerhalb von LANs sind diese Verfahren deshalb ungeeignet.

Die Mängel von "time" und "daytime" führten zur Entwicklung von NTP, das Paketlaufzeiten im Netz misst und ausgleicht. Der NTP-Dienst arbeitet parallel zur Systemuhr als eigenständige Uhr.

W06 NAS & SMB

Zu Beginn aktualisieren wir erst einmal den Raspberry Pi. Dazu geben Sie folgende Befehle in das Terminal ein. Der erste Befehl aktualisiert die Programm - Bibliothek und der zweite

aktualisiert das Betriebssystem des Pi.

```
pi@z54:~ $ sudo apt upgrade && sudo apt update
```

Nun öffnen wir das Terminal und installieren Samba via folgenden Befehl. Die aufkommende Meldung mit Nein bestätigen.

```
pi@z54:~ $ sudo apt-get install samba samba-common smbclient cifs-utils
```

Nach der erfolgreich Installation überprüfen wir den Status.

```
pi@z54:~ $ sudo service smbd status
pi@z54:~ $ sudo service nmbd status
```

The screenshot shows two terminal windows side-by-side. The left window displays the output of the command `sudo service smbd status`, which shows the Samba SMB Daemon is active and ready to serve connections. The right window displays the output of `sudo service nmbd status`, showing the Samba NMB Daemon is also active and ready to serve connections. Both windows show logs from March 21, 2020, at 12:21:46.

Die mitgelieferte Konfigurationsdatei müssen wir später noch anpassen. Zuerst wird sie mit folgendem Befehl gesichert:

```
pi@z54:~ $ sudo cp /etc/samba/smb.conf /etc/samba/smb.conf.bak
```

Nun überprüfen wir ob die Freigabe verfügbar ist.

```
pi@z54:~ $ smbclient -N -L //192.168.0.103/
```

Nun überprüft man ob man sich mit dem pi User anmelden kann. Wenn man nach einem Passwort gefragt wird, ist alles i.o.

```
pi@z54:~ $ smbclient -U pi -L //192.168.0.103/
```

Nun setzt man die Freigabe für das Verzeichnis mit folgenden Befehl. Chmod 1770 (chmod a+rwx,o-rwx,ug+s,+t,u-s,g-s) setzt die Berechtigungen so, dass (U)ser / Eigentümer lesen, schreiben und ausführen kann. (G)roup lesen, schreiben und ausführen kann. (O)thers kann nicht lesen, kann nicht schreiben und kann nicht ausführen.

```
pi@z54:~ $ mkdir /home/pi/smb
pi@z54:~ $ sudo chmod 1770 /home/pi/smb
```

Danach öffnen wir die Samba Konfigurationsdatei mit dem Befehl.

```
pi@z54:~ $ sudo nano /etc/samba/smb.conf
```

Und fügen dort die Freigabe ein.

```
[pi]
comment = Freigabe pi
path = /home/pi/smb
valid users = pi
Browseable = yes
Writable = Yes
only guest = no
create mask = 0777
directory mask = 0777
Public = no
Guest ok = no
```

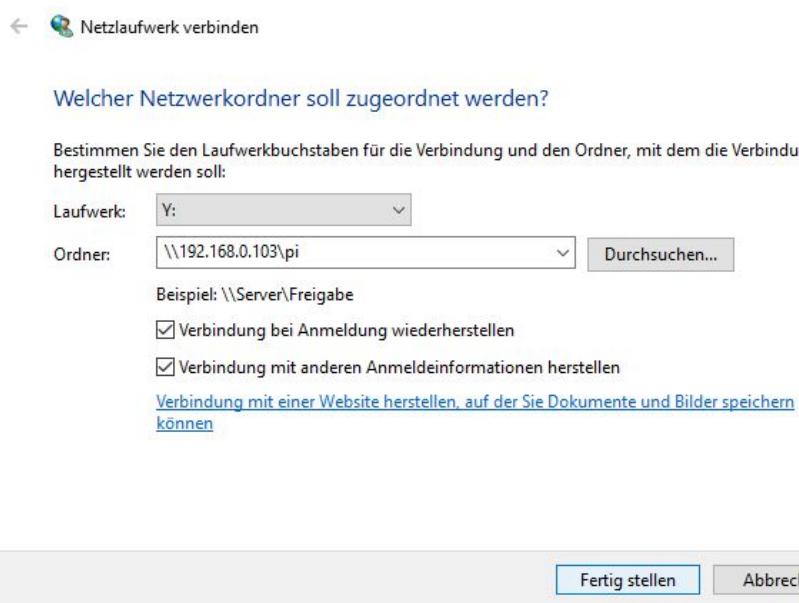
Danach startet man den Service neu, dadurch wird die Konfiguration getestet.

```
pi@z54:~ $ sudo service smbd restart
pi@z54:~ $ testparm
```

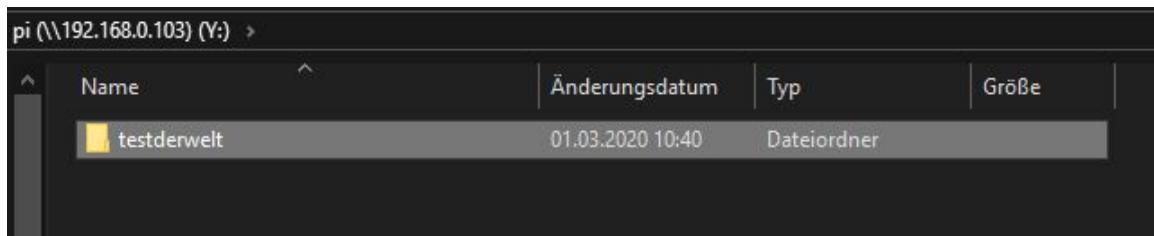
Nun erstellen wir ein Double des Pi Users. Und geben dem Pi User ein Passwort.

```
pi@z54:~ $ sudo smbpasswd -a pi
```

Nun versuchen wir uns mit dem Netzwerklaufwerk zu verbinden.



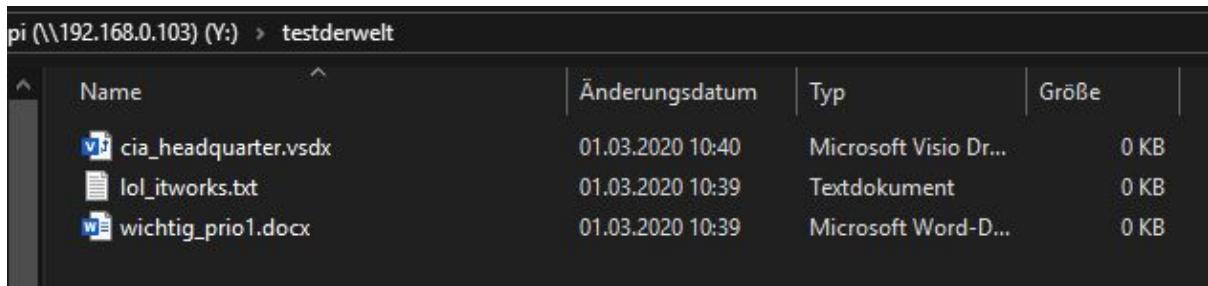
Nun erstellen wir einen Ordner auf dem Laufwerk.



Den Ordner sieht man auch auf dem Terminal mittels Befehl ls.

```
pi@raspberrypi:~/test $ ls
testderwelt
pi@raspberrypi:~/test $
```

In dem Ordner erstelle ich einige Files.



Diese Files kann man dann auch im Terminal sehen.

```
pi@raspberrypi:~/test/testderwelt $ ls
cia_headquarter.vsdx  lol_itworks.txt  wichtig_priol.docx
pi@raspberrypi:~/test/testderwelt $
```

Nun überprüfen wir ob der NTFS Treiber installiert ist.

```
pi@z54:~ $ ls -l /lib/modules/$ (uname -r)/kernel/fs
```

Nun installiere wir den NTFS Treiber um auch zu schreiben können.

```
pi@z54:~ $ sudo apt-get -y install ntfs-3g hfsutils hfsprogs exfat-fuse
```

Nun kann man den USB Stick einstecken und folgenden Befehl ausführen.

```
pi@z54:~ $ sudo blkid -o list -w /dev/null
```

Durch den Befehl habe ich folgenden Output erhalten.

```
pi@raspberrypi:~/test/testderwelt $ sudo blkid -o list -w /dev/null
/dev/device                               fs_type      label      mount point
                                         (in use)
/dev/mmcblk0                             vfat        RECOVERY   (not mounted)
                                         SETTINGS   (not mounted)
                                         boot       /boot
                                         root       /
                                         ERSATZ LUIS /media/pi/ERSATZ LUIS
/dev/mmcblk0p1                           9AE3-8BA0
                                         ext4
                                         SETTINGS   (not mounted)
                                         boot       /boot
                                         root       /
                                         ERSATZ LUIS /media/pi/ERSATZ LUIS
/dev/mmcblk0p5                           ald317dd-077b-41c7-8600-d2651b6dfbf0
                                         ext4
                                         boot       /boot
                                         root       /
                                         ERSATZ LUIS /media/pi/ERSATZ LUIS
/dev/mmcblk0p6                           055E-C48A
                                         ext4
                                         boot       /boot
                                         root       /
                                         ERSATZ LUIS /media/pi/ERSATZ LUIS
/dev/mmcblk0p7                           9d78249b-5843-4195-b79d-b18425e54055
                                         ext4
                                         boot       /boot
                                         root       /
                                         ERSATZ LUIS /media/pi/ERSATZ LUIS
/dev/sdal                                5480BD7680BDE4
                                         ntfs
                                         ERSATZ LUIS /media/pi/ERSATZ LUIS
pi@raspberrypi:~/test/testderwelt $
```

Nun erstellen wir einen Mountpunkt für den USB Stick.

```
pi@z54:~ $ sudo mkdir /media/nas
```

Dann öffnen wir die Automount Datei.

```
pi@z54:~ $ sudo nano /etc/fstab
```

Im File muss man dann folgenden Einzeiler hinzufügen. Nachher sollte man einen reboot machen.

```
UUID=5480BD7680BD5EE4 /media/nas/ ntfs-3g  
utf8,uid=root,gid=sambashare,noatime 0
```

Nun richten wir die Benutzergruppen und zwei Benutzer ein. Gruppe gibt bereits.

```
pi@z54:~ $ sudo addgroup sambashare  
pi@z54:~ $ sudo useradd user1 -G sambashare  
pi@z54:~ $ sudo useradd user2 -G sambashare  
pi@z54:~ $ sudo smbpasswd -a user1  
pi@z54:~ $ sudo smbpasswd -a user2
```

Jetzt erstellt man den Freigabeordner share im Mount-Point.

```
pi@z54:~ $ sudo mkdir /media/nas/share  
pi@z54:~ $ ls /media/nas/share -al
```

Nun öffnen wir das Konfigurationsfile mit folgenden Befehl.

```
pi@z54:~ $ sudo nano /etc/samba/smb.conf
```

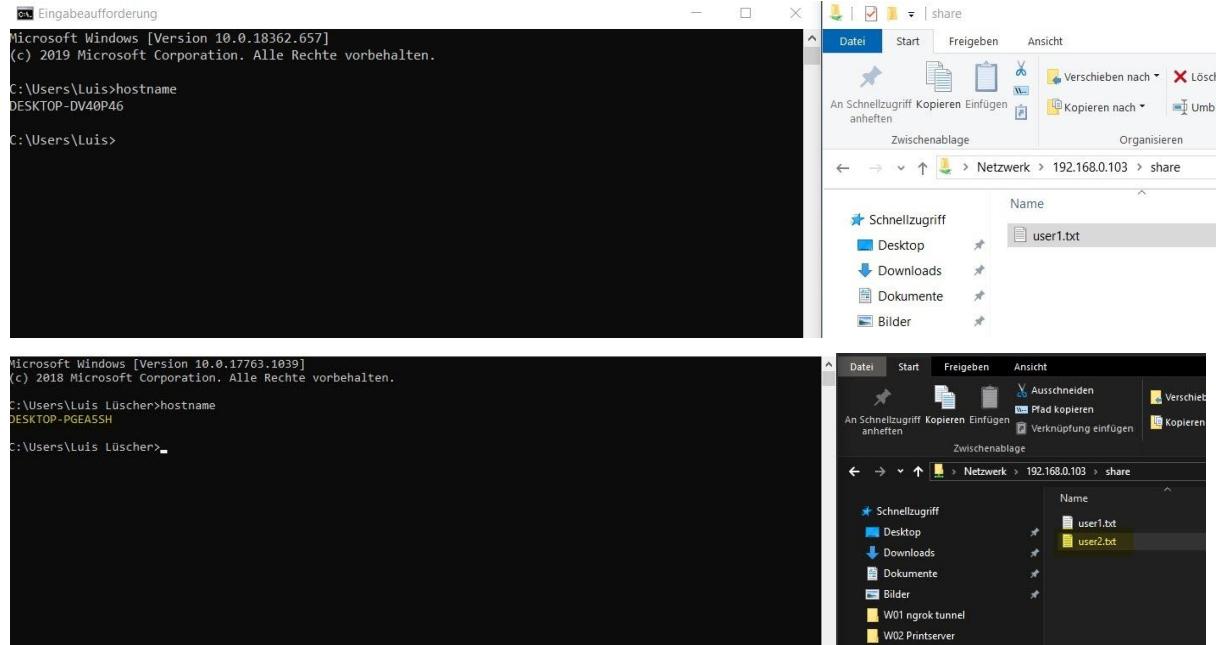
Und fügen die Freigabe dort ein.

```
[share]  
Comment = Pi shared folder  
Path = /media/nas/share  
valid users = @sambashare  
Browseable = yes  
Writable = Yes  
only guest = no  
create mask = 0777  
directory mask = 0777  
Public = yes  
Guest ok = yes
```

Nun testen wir wieder die Konfiguration.

```
pi@z54:~ $ sudo service smbd restart
pi@z54:~ $ testparm
```

Nun greife ich von zwei verschiedenen PC auf den Share zu und dies dann entweder mit user1 oder user2.



Zudem kann man überprüfen, ob die Daten wirklich auf dem USB Stick sind, wenn man den Stick an einem PC anschliesst und dann die vorhin erstellten Files sieht.

ERSATZ LUIS (E:) > share				
	Name	Aenderungsdatum	Typ	Größe
	user1.txt	01.03.2020 11:17	Textdokument	0 KB
	user2.txt	01.03.2020 11:18	Textdokument	0 KB

Zum Ende installieren wir noch einen FTP-Server. und wechseln in das jeweilige Verzeichnis.

```
pi@z54:~ $ sudo apt-get install proftpd
pi@z54:~ $ cd /etc/proftpd/
```

Nun erstellen wir den User lslschr.

```
pi@z54:~ $ sudo ftppasswd --passwd --name lslschr --gid 33 --uid 33
--home /var/www/ --shell /bin/false
```

Dann muss man noch den User freischalten.

```
pi@z54:~ $ sudo nano /etc/proftpd/proftpd.conf
```

Und fügen folgenden Code ein.

```
DefaultRoot ~  
AuthOrder mod_auth_file.c mod_auth_unix.c  
AuthUserFile /etc/proftpd/ftp.passwd  
AuthPAM off  
RequireValidShell off
```

Danach starten wir den Server neu.

```
pi@z54:~ $ sudo /etc/init.d/proftpd restart
```

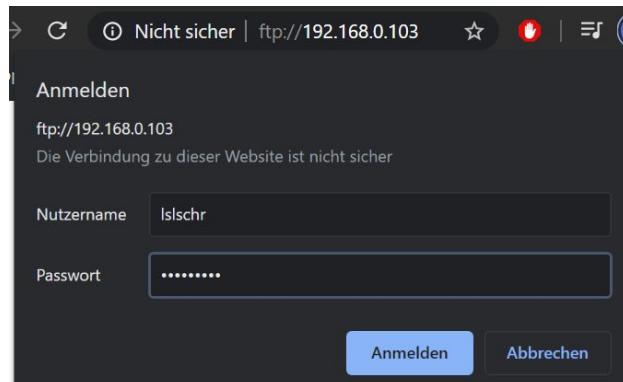
Es kann sein, dass man dann immer noch nicht zugreifen kann, dann einfach folgende Berechtigungen geben.

```
pi@z54:~ $ sudo chmod 775 /var/www  
pi@z54:~ $ sudo chmod g+s /var/www
```

Man kann dann einen beliebigen Browser öffnen und folgenden Link eingeben.

```
ftp://192.168.0.103
```

Nun muss man sich mit dem vorhin erstellten User einloggen.



Index von /ftp/

[[übergeordnetes Verzeichnis]]

	Name	Größe	Änderungsdatum
📄	ftpserver	0 B	21.03.20, 14:15:00
📄	luescher	0 B	21.03.20, 14:15:00
📄	luis	0 B	21.03.20, 14:15:00
📄	m126	0 B	21.03.20, 14:15:00

Ich habe zur demonstration einige Files erstellt um die Funktion des FTP Server zu zeigen. Wenn man fertig ist sollte man noch den mount-Point in fstab löschen, aufgrund der Funktionalität.

W07 Webmin

Zu Beginn müssen wir noch einige Dinge installieren, um auf den Raspi via SSL zugreifen zu können.

```
pi@z54:~ $ sudo apt-get install libnet-ssleay-perl libio-socket-ssl-perl
```

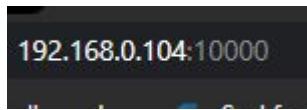
Danach installieren wir die aktuelle Version von Webmin. Auf dieser [Website](#) kann man nachschauen welche die aktuelle Version ist.

```
pi@z54:~ $ wget
http://prdownloads.sourceforge.net/webadmin/webmin-1.941.tar.gz
```

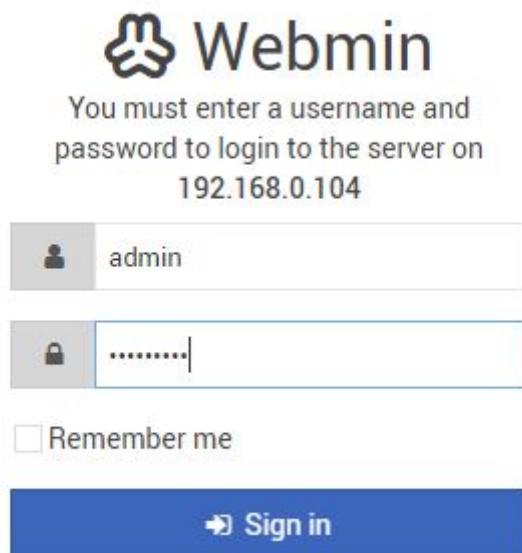
Dann extrahiert man das vorhin heruntergeladenen File. Und wechseln dann in das neue Verzeichnis und starten das Setup Script. Während dem Setup Script kann man sich an die Default Werte halten.

```
pi@z54:~ $ tar -zxvf webmin-1.941.tar.gz
pi@z54:~ $ cd webmin-1.941/
pi@z54:~ $ sudo ./setup.sh
```

Nun kann man auf das Webmin Tool zugreifen. http://RASPI_IP:10000



Danach muss man sich mit den vorhin im Setup Script definierten Credentials anmelden.



You must enter a username and password to login to the server on 192.168.0.104

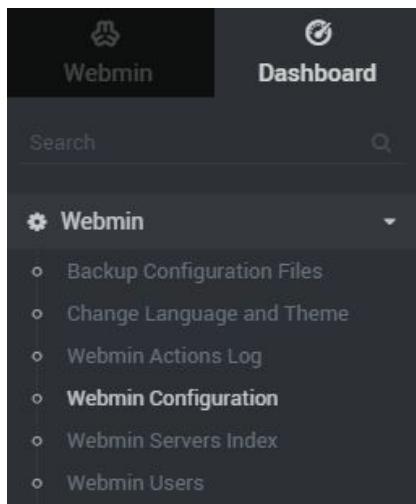
admin

.....|

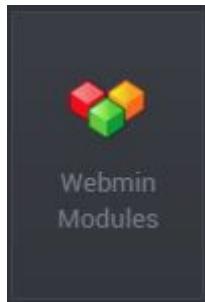
Remember me

Sign in

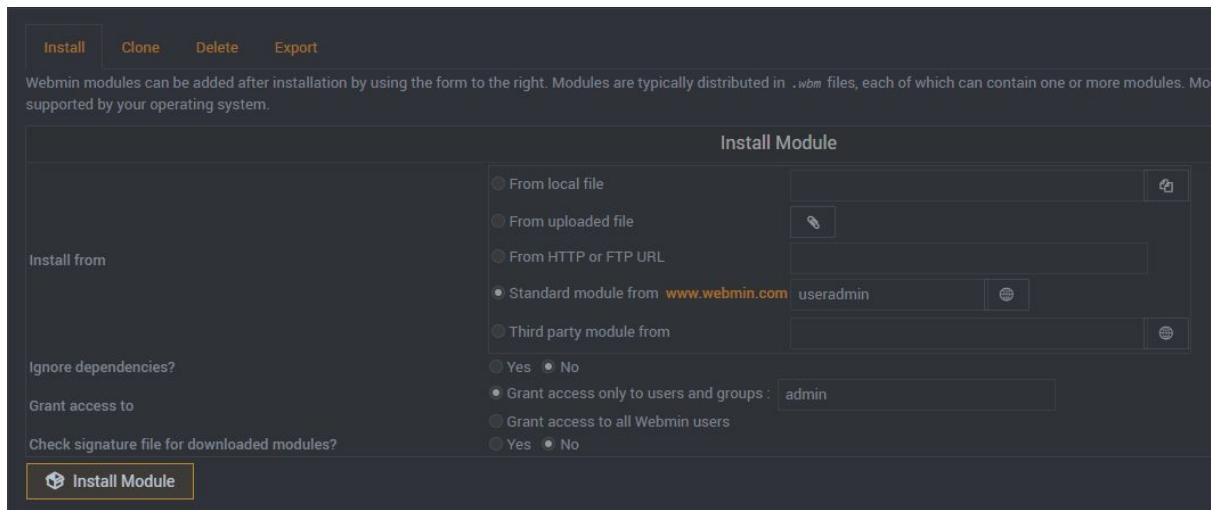
Auf der Seite navigieren wir dann unter dem Punkt "Webmin" zu "Webmin Configuration".



Hier öffnen wir dann die Webmin Modules.



Und erhalten wir folgendes Formular. Hier wählen wir das Standard Modul und fügen das useradmin Modul hinzu. Unsere Auswahl speichern wir dann mittels Install Module.



The screenshot shows the 'Install Module' dialog in Webmin. At the top, there are four buttons: 'Install' (highlighted in yellow), 'Clone', 'Delete', and 'Export'. Below them is a note: 'Webmin modules can be added after installation by using the form to the right. Modules are typically distributed in .wbb files, each of which can contain one or more modules. Mo...' followed by 'Supported by your operating system.' The main area is titled 'Install Module' and contains several options:

- From local file (with a file input field)
- From uploaded file (with a file input field)
- From HTTP or FTP URL (with a text input field)
- Standard module from www.webmin.com useradmin (with a file input field)
- Third party module from (with a text input field)

Below these are two sections:

- 'Ignore dependencies?' with radio buttons 'Yes' (selected) and 'No'
- 'Grant access to' with radio buttons 'Grant access only to users and groups : admin' (selected) and 'Grant access to all Webmin users'

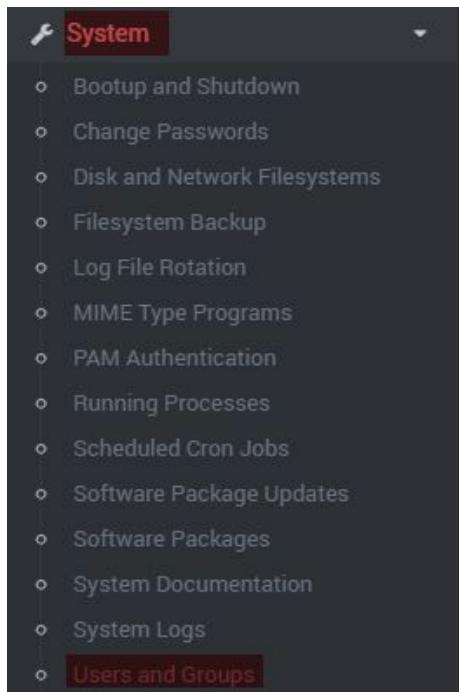
At the bottom left is a 'Check signature file for downloaded modules?' checkbox, which is unchecked. At the bottom right is a large 'Install Module' button, which is highlighted with a yellow border.

Wenn das Modul erfolgreich installiert wurde,

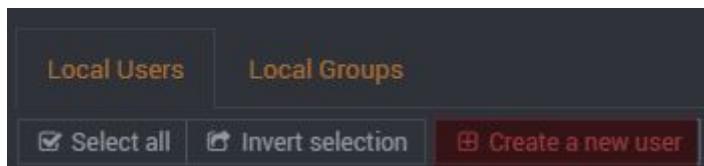
The following modules have been successfully installed and added to your access control list :

Users and Groups in `/home/pi/webmin-1.941/useradmin` (5168 kB) under category System

kann man dann unter dem Punkt "System" und dann unter "Users and Groups" einen neuen user erstellen.



Hier dann einfach auf Create a new User klicken.



Und dann folgende Werte setzen:

Einen Username in diesem Fall user1, einen Real name in diesem Fall Ricardo Lüscher und dann noch ein Passwort setzen. Danach kann man den User mit "Create" erstellen.

Luis Ricardo Lüscher
M126 - Peripheriegeräte im Netzwerkbetrieb einsetzen

The screenshot shows a user configuration interface for creating a new user account. The user has chosen the username 'user1'. Other settings include:

- User ID:** Automatic (selected), value 1001.
- Real name:** Ricardo Lüscher.
- Home directory:** Automatic (selected).
- Shell:** /bin/sh (selected).
- Password:** Normal password set to Admin1234.
- Groups:** Primary group is Existing group 'users'.
- Group Membership:** Secondary groups listed include root, daemon, bin, sys, adm, and tty.
- Upon Creation:** Options for creating home directory, copying template files, and creating users in other modules are present.

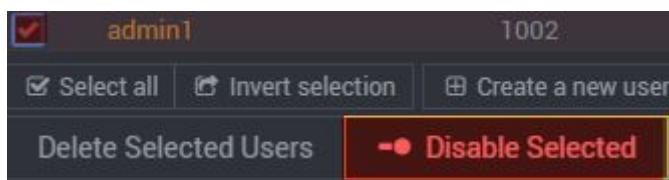
A red box highlights the 'Create' button at the bottom left of the form.

Nun werden wir noch einen Admin User erstellen. Auch hier einen Username in diesem Fall admin1, einen Real name in diesem Fall Luis Lüscher und dann noch ein Passwort setzen. Zudem wählen wir noch eine zweite Gruppe für diesen User aus, hier wählen wir "adm" dies steht für admin. Danach kann man den User mit "Create" erstellen.

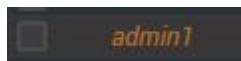
Luis Ricardo Lüscher
M126 - Peripheriegeräte im Netzwerkbetrieb einsetzen

The screenshot shows the configuration of a new user named 'admin1'. The 'Primary group' is set to 'Existing group' with 'users'. In the 'Secondary groups' section, 'adm' is selected in the 'In groups' list. Under 'Upon Creation...', 'Create home directory?' and 'Copy template files to home directory?' are checked.

Nun werden wir den erstellten Admin User disabeln. Dafür wählen wir ihn aus und klicken auf “disable selected”.

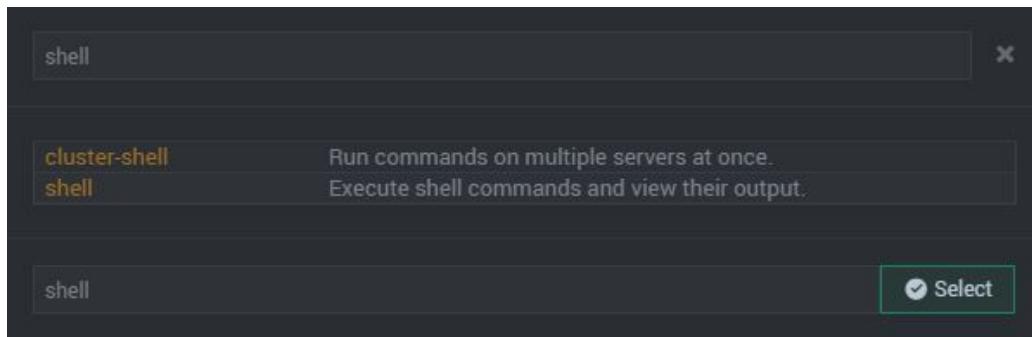


Wenn man alles richtig gemacht hat, ist der Name des Users nun kursiv.



Nun werden wir noch einige Module installieren.

Zuerst werden wir Shell installieren. Dazu suchen wir nach "shell" wählen dieses auch aus und bestätigen unsere Auswahl mittels "Select".



Nach einer erfolgreichen Installation finden wir die Shell unter dem Punkt "Others" => "Command Shell".

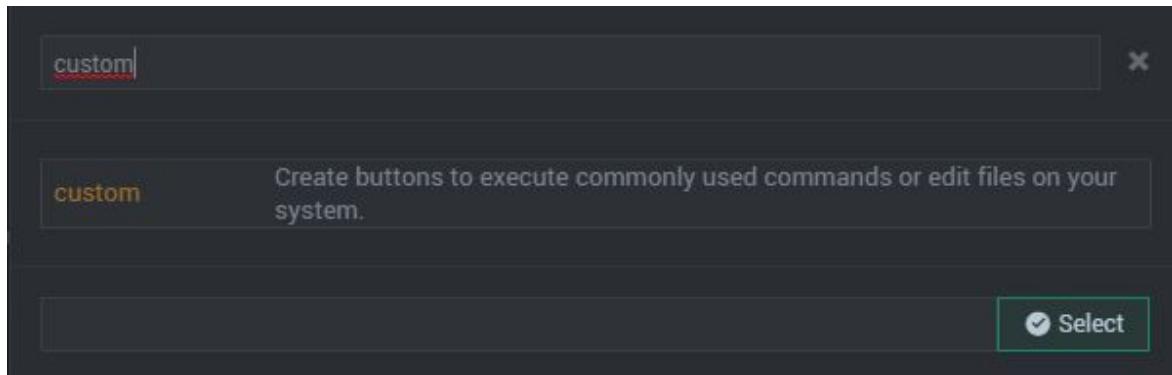


Mittels diesem Modul können wir Commands über das Web Tool ausführen und erhalten auch den output direkt über die Website. Im Beispiel führe ich den Befehl "ip a" aus. Und erhalte direkt die Ausgabe aus dem Terminal.

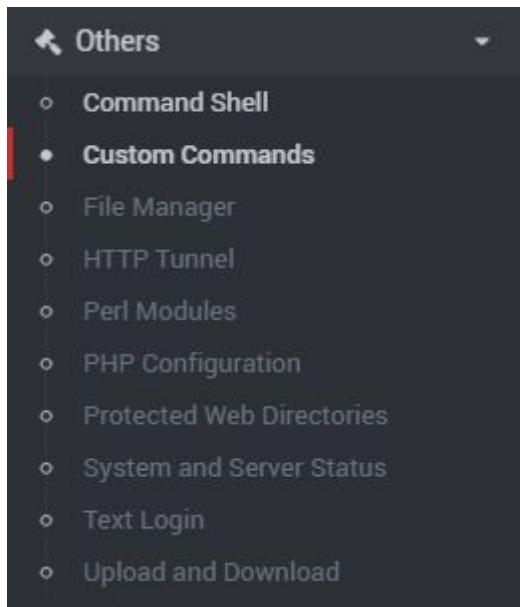
The screenshot shows a terminal window with the command 'ip a' entered. The output is displayed below:

```
> ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default qlen 1000
    link/ether b8:27:eb:2c:72:2d brd ff:ff:ff:ff:ff:ff
    inet 192.168.0.104/24 brd 192.168.0.255 scope global noprefixroute eth0
        valid_lft forever preferred_lft forever
    inet6 fe80::46b7:6be9:8a1c:3483/64 scope link
        valid_lft forever preferred_lft forever
3: wlan0: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 qdisc pfifo_fast state DOWN group default qlen 1000
    link/ether b8:27:eb:79:27:78 brd ff:ff:ff:ff:ff:ff
```

Nun installieren wir das Modul "custom".



Nach der erfolgreichen Installation finden wir das Modul unter dem Punkt "Others" als "Custom Commands".



Meiner Meinung nach ähnelt dies der "alias" Funktion im Terminal. Man kann dem Command einen Namen geben und im Hintergrund einen Befehl ausführen lassen der evtl. sehr lange ist und man diesen darum nicht immer eingeben möchte. Mit Save speichert man seine Parameter.

The screenshot shows the 'Command' configuration page in Webmin. The 'Command ID' is 1584875979 and the 'Description' is 'show echo string'. The 'HTML description' field contains the command 'echo "Hey my name is Luis Luescher"'. Under 'Run in directory', 'Default' is selected. Under 'Run as user', 'Webmin user' is selected with 'pi' entered. 'Command outputs HTML?' has 'No' selected. 'Hide command when executing?' has 'No' selected. 'Maximum time to wait for command?' has 'Default' selected. 'Output style' has 'Show in Webmin UI' selected. At the bottom, there are buttons for 'Save' (highlighted), 'Clone', and 'Delete'.

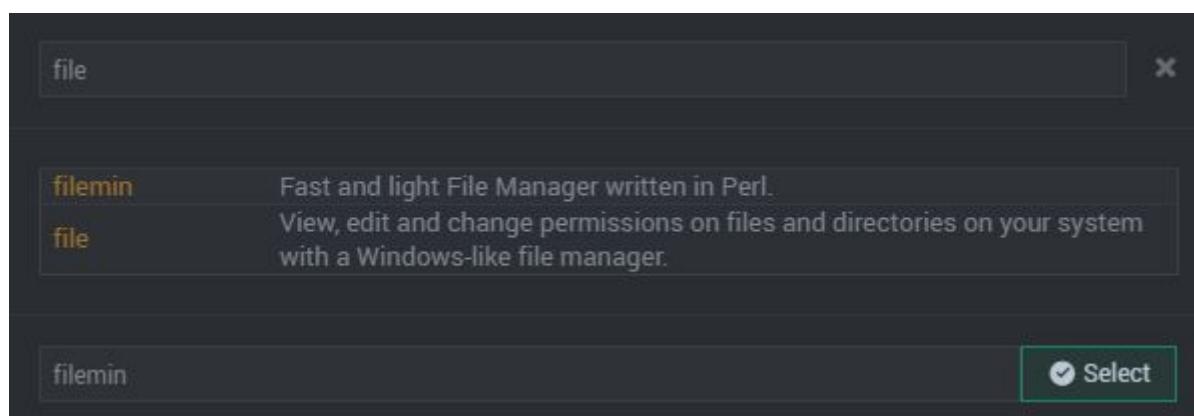
Nun führen wir den erstellten Befehl aus.

▶ show echo string

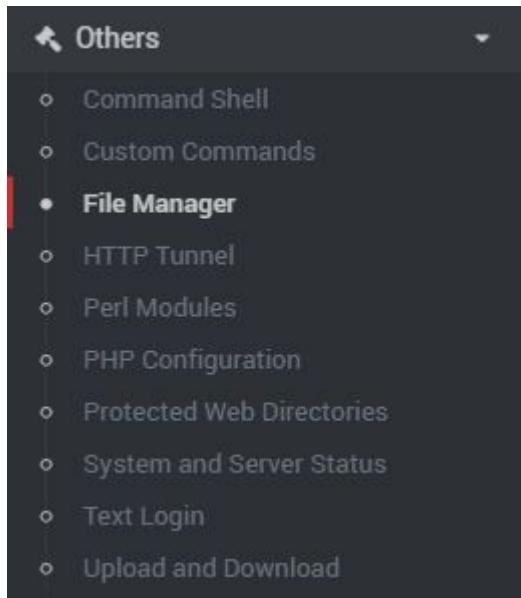
Und wir sehen der Output ist korrekt.

The output window displays the command 'echo "Hey my name is Luis Luescher"' followed by its execution result: 'Hey my name is Luis Luescher'.

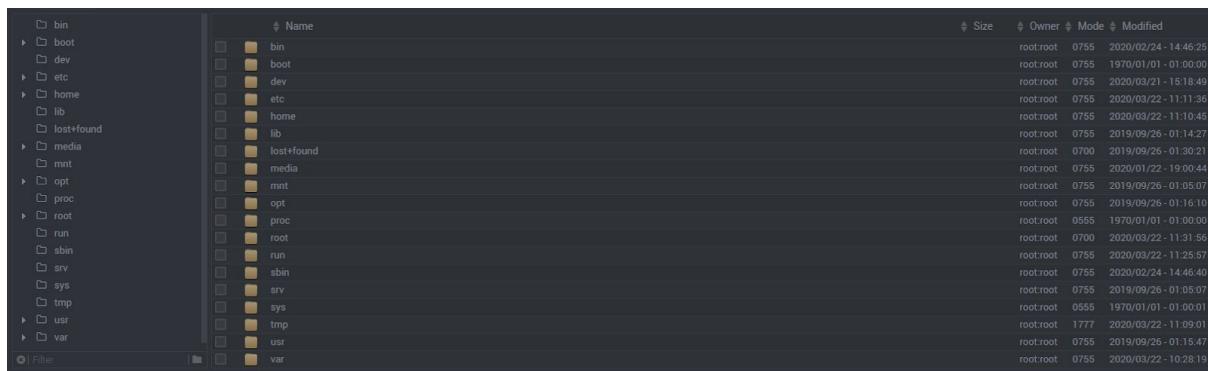
Dies ist das letzte Modul welches ich installiert habe. Wir suchen nach "file" und wählen filemin.



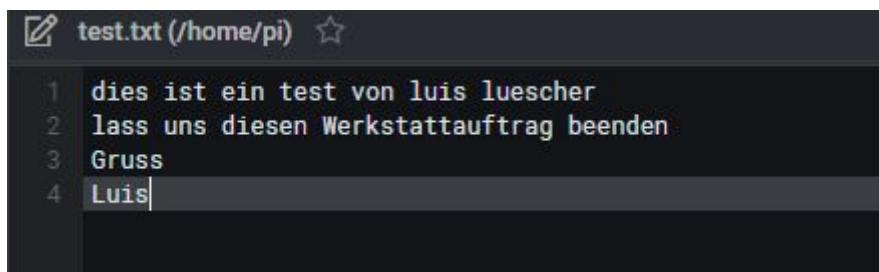
Nach einer erfolgreichen Installation finden wir den File Manager unter dem Punkt "Others".



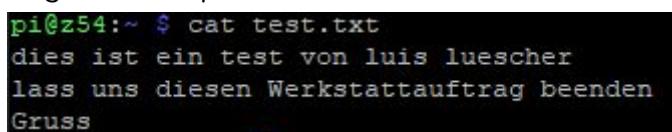
Nun hat man einen File Explorer so wie man ihn von Windows, Mac OS etc kennt.



Um die Funktionalität zu testen, erstellen wir ein Test File.



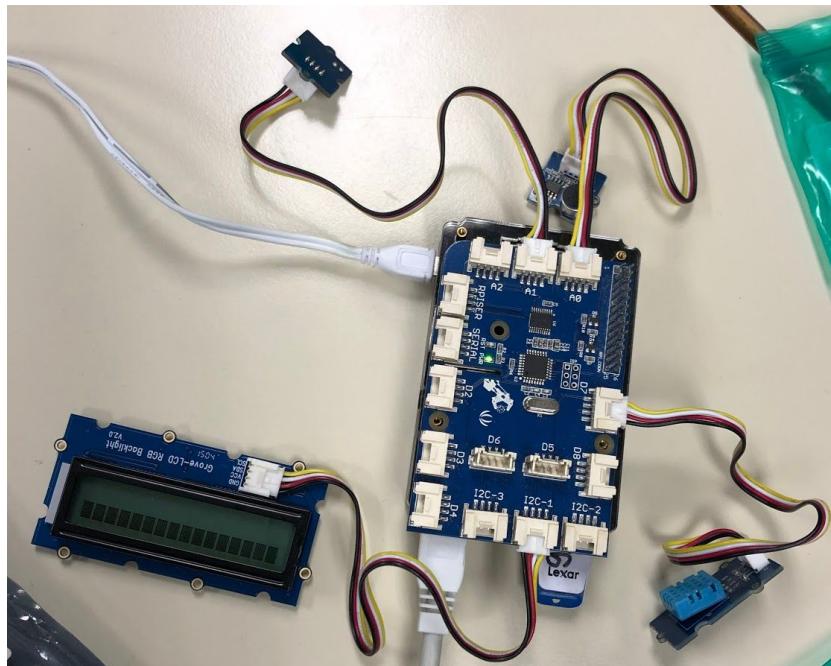
Folgenden Output erhalten wir im Terminal. Es scheint ohne Probleme zu funktionieren.



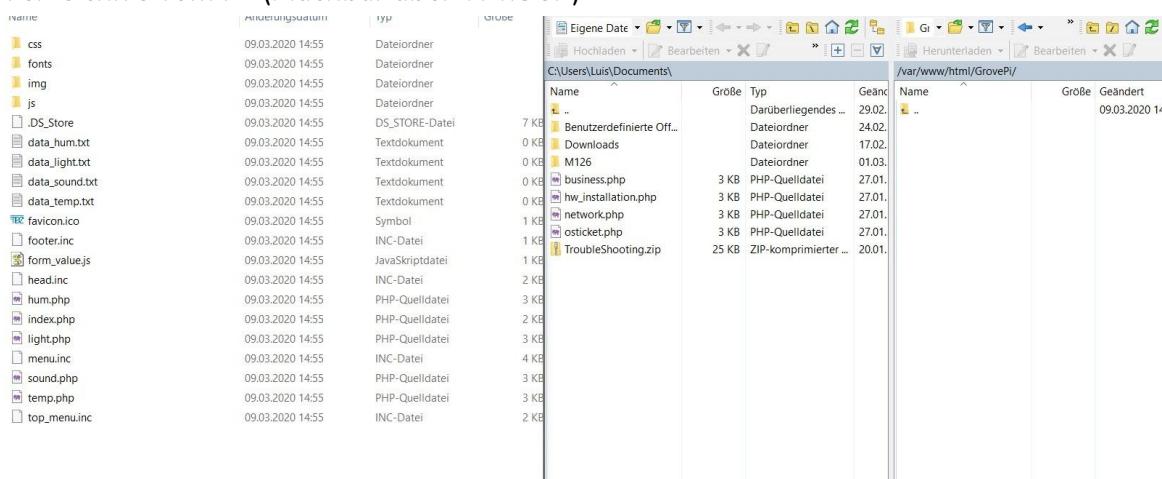
W10a GrovePi Einführung

Zu Beginn am GrovePi die Sensoren folgendesmassen anschliessen.

- Sound Sensor = A0
- Licht Sensor = A1
- Temp & Luftfeuchtigkeit Sensor = D7
- Grove LCD RGB Backlight = I2C - 1



Nun laden wir die bereits erstellte Version der Website herunter (Website.zip). Diese entpackt man und verschiebt den darin enthaltenen Ordner GrovePi ins var/www/html Verzeichnis. Danach kopiert man den Ordner Scripts in den Ordner ~/PyScripts im Home Verzeichnis vom Pi. (Machbar über WinSCP)



Danach öffnet man das GrovePi_Run.py

```
pi@z54:~ $ sudo nano GrovePi_Run.py
```

Dann muss man die jeweiligen Pfade überprüfen. Eventuell noch anpassen.

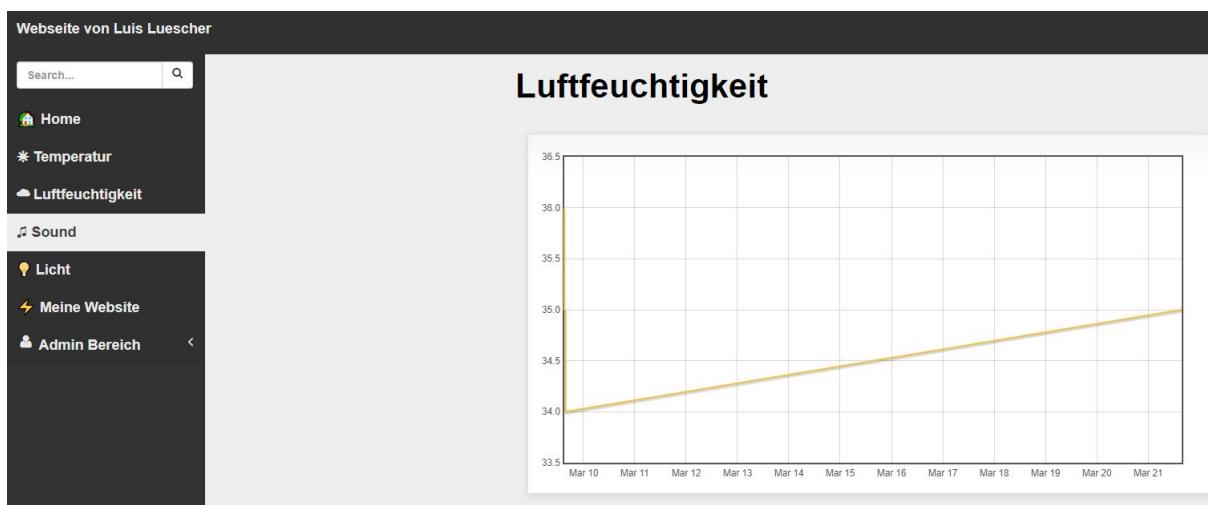
```
while True:  
    try:  
        #Dateien öffnen  
        f_temp=open("/var/www/html/GrovePi/data_temp.txt", "a")  
        f_hum=open("/var/www/html/GrovePi/data_hum.txt", "a")  
        f_sound=open("/var/www/html/GrovePi/data_sound.txt", "a")  
        f_light=open("/var/www/html/GrovePi/data_light.txt", "a")  
        #Werte auslesen  
        [ temp,hum ] = grovepi.dht(dht_sensor_port,dht_sensor_type)  
        sensor_value_sound = grovepi.analogRead(sound_sensor) #use A  
        sensor_value_light = grovepi.analogRead(light_sensor)  
        #shel Ausgabe  
        print "temp =", temp, "C\thumidity =", hum,"%"
```

Danach kann man mal das Python Script zu starten.

```
pi@z54:~ $ sudo python GrovePi_Run.py
```

Nun sollte der LCD Display angehen und die aktuellen Daten mitteilen. Nun kann man auch die Website öffnen und sieht dann das die Daten dann entsprechend aktualisiert werden.





W10b GrovePi Sensor

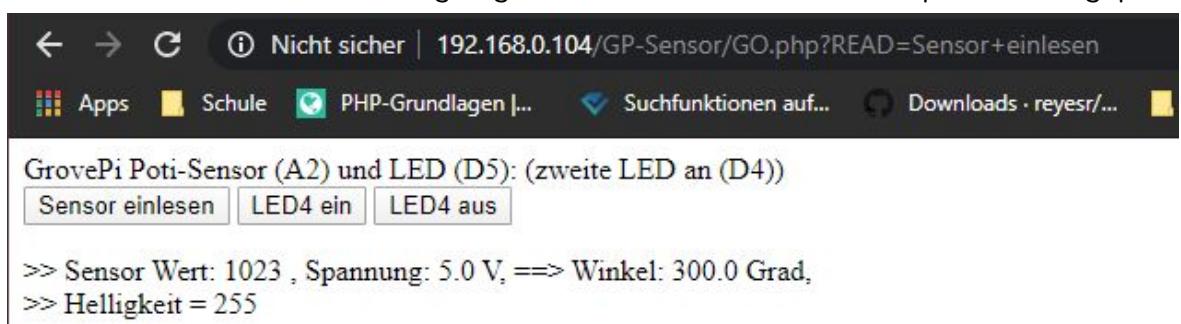
Zu Beginn dieses Auftrages muss man die HW folgendermassen anschliessen.

- Dreh-Sensor = A2
- LED4 = D4

Nun kopieren wir alle Daten des Demoprojekts in einen neuen Ordner.

```
pi@z54:~ $ sudo mkdir /var/www/html/GP-Sensor
pi@z54:~ $ sudo cp /var/www/html/Demoprojekt/* /var/www/html/GP-Sensor/
```

Danach habe ich das Script so verändert, dass nur noch die try Funktion enthalten ist.
Danach habe ich das im Demoprojekt erstelle LED4.php File kopiert und in GO.php genannt.
Nun nur noch einen Button hinzugefügt und noch den Befehl dementsprechend angepasst.



Das Resultat sieht dann so aus. Die Daten des Sensors werden dann im Paragraph angezeigt. So sieht mein Script aus.

```
import time
import grovepi

# Connect the Grove Rotary Angle Sensor to analog port A2
# SIG,NC,VCC,GND
potentiometer = 2
```

```
# Connect the LED to digital port D5
# SIG,NC,VCC,GND
led = 5

grovepi.pinMode(potentiometer,"INPUT")
grovepi.pinMode(led,"OUTPUT")
time.sleep(1)

# Reference voltage of ADC is 5v
adc_ref = 5

# Vcc of the grove interface is normally 5v
grove_vcc = 5

# Full value of the rotary angle is 300 degrees, as per it's specs (0 to
300)
full_angle = 300

try:
    # Read sensor value from potentiometer
    sensor_value = grovepi.analogRead(potentiometer)

    # Calculate voltage
    voltage = round((float)(sensor_value) * adc_ref / 1023, 2)

    # Calculate rotation in degrees (0 to 300)
    degrees = round((voltage * full_angle) / grove_vcc, 2)

    # Calculate LED brightness (0 to 255) from degrees (0 to 300)
    brightness = int(degrees / full_angle * 255)

    # Give PWM output to LED
    grovepi.analogWrite(led,brightness)

    print ">> Sensor Wert:", sensor_value, ", Spannung:", voltage, "V,
=> Winkel:", degrees, "Grad, </Br>> Helligkeit = ", brightness
except IOError:
    print ("IO-Error")
```

W10c GrovePi Twitter Feed

Für diese Werkstatt wird ein Twitter Account vorausgesetzt. Dazu muss man dann eine App erstellen. Zuerst muss man unter developer.twitter.com eine App erstellen bzw. beantragen. Dazu klickt man auf "Create an app".

The screenshot shows the Twitter Developers website at developer.twitter.com/en/apps. The top navigation bar includes links for YouTube, Maps, TBZ, ESXI, Info & Menüplan, and Twitter Developers. Below the bar, there's a purple header with the Twitter logo and the text "Developer Use cases Products Docs More Labs Dashboard Testacc02192281". A message box says "#welcome We have sunset apps.twitter.com. You can manage any of your existing apps in all of the same ways through this site." At the bottom left, there's a link "Create an app" and a blue underline over the word "Apps".

Danach öffnet sich eine Seite auf der man einige Fragen beantworten muss zu seiner Applikation, grundsätzlich kann man hier irgendetwas hineinschreiben. Ich habe einfach beschrieben wofür dieser Zugang wäre und was ich damit vor habe. Danach wird der hellblaue Button "Create" blau und somit kann man seine Parameter an Twitter senden.

The form has fields for "App name (required)" with a placeholder "My Awesome App" and a character limit of 32. Below it is a "Application description (required)" field with a placeholder "Please be detailed." and a character limit of between 10 and 200. A note says "Share a description of your app. This description will be visible to users so this is a good place to tell them what your app does."

Wann dann Twitter die Anfrage approved hat, sieht man unter seinem Profil seine neue App.



The screenshot shows the 'Keys and tokens' section of the Twitter developer dashboard for the app 'W10c Grove Pi Twitter Feed'. It displays two sets of API keys: 'Consumer API keys' and 'Access token & access token secret'. The 'Consumer API keys' section includes fields for 'API key' and 'API secret key', both of which are redacted with black bars. A 'Regenerate' button is located to the right. The 'Access token & access token secret' section has a 'Generate' button. Below these sections, a note states: 'This is where you can create a token to auth into your Twitter account.'

Unter Keys und Tokens kann man dann seine API Keys erhalten. Wenn wir diese haben können wir effektiv mit dem Auftrag anfangen.

Nun werden wir Versuchen mit einem Python Skript die Angaben der Sensor auf Twitter via Tweet zu teilen. Dafür hat man in folgenden Ordner bereits ein Sample => /GrovePi/Projects/Sensor/Sensor_Twitter_Feed. Jedoch ist diese Version relativ schlecht darum habe ich das Script verändert, sodass es auch ohne Probleme zB. mit Python 2 funktioniert. Das Script sieht dann folgendermassen aus. Dieses Script hat dann ebenfalls noch die Daten des Drehpotentiometer eingelesen und geteilt.

Zu Beginn installieren wir noch Tweepy.

```
pi@z54:~ $ sudo pip install tweepy
```

Folgendes Script habe ich verwendet:

```
import tweepy
import time
import grovepi
import math

# Connections
sound_sensor = 0          # port A0
light_sensor = 1           # port A1
temperature_sensor = 2    # port D2
```

```
led = 3                      # port D3

intro_str = "DI Lab's"

# Authenticate to Twitter
auth = tweepy.OAuthHandler("opFVEGL9dxkNloDiBhdJ9ij55",
"Qf5LCmHXaB6Nwdf1DKhywlyrtZ88TBeFE2L30Vh1bFH吳CgvSD")
auth.set_access_token("1237028090221596672-71wcdFHSLoMwt963auTs7dMYNGiWuP
","2hE6UePyPfexOwqln8LmVb7qAlpWts32KRGdOUCbbINM")
api = tweepy.API(auth)
# test authentication
grovepi.pinMode(led,"OUTPUT")
grovepi.analogWrite(led,255)  #turn led to max to show readiness

# Connect the Grove Rotary Angle Sensor to analog port A2
# SIG,NC,VCC,GND
potentiometer = 2

# Connect the LED to digital port D5
# SIG,NC,VCC,GND
led = 5

grovepi.pinMode(potentiometer,"INPUT")
grovepi.pinMode(led,"OUTPUT")
time.sleep(1)

# Reference voltage of ADC is 5v
adc_ref = 5

# Vcc of the grove interface is normally 5v
grove_vcc = 5

# Full value of the rotary angle is 300 degrees, as per it's specs (0 to
300)
full_angle = 300

while True:

    # Error handling in case of problems communicating with the GrovePi
    try:
        # Read sensor value from potentiometer
        sensor_value = grovepi.analogRead(potentiometer)
```

```
# Calculate voltage
voltage = round((float)(sensor_value) * adc_ref / 1023, 2)

# Calculate rotation in degrees (0 to 300)
degrees = round((voltage * full_angle) / grove_vcc, 2)

# Calculate LED brightness (0 to 255) from degrees (0 to 300)
brightness = int(degrees / full_angle * 255)

# Give PWM output to LED
grovepi.analogWrite(led,brightness)

# Get value from light sensor
light_intensity = grovepi.analogRead(light_sensor)

# Give PWM output to LED
grovepi.analogWrite(led,light_intensity/4)

# Get sound level
sound_level = grovepi.analogRead(sound_sensor)

time.sleep(0.5)

# Get value from temperature sensor
[t,h]=[0,0]
[t,h] = grovepi.dht(temperature_sensor,0)

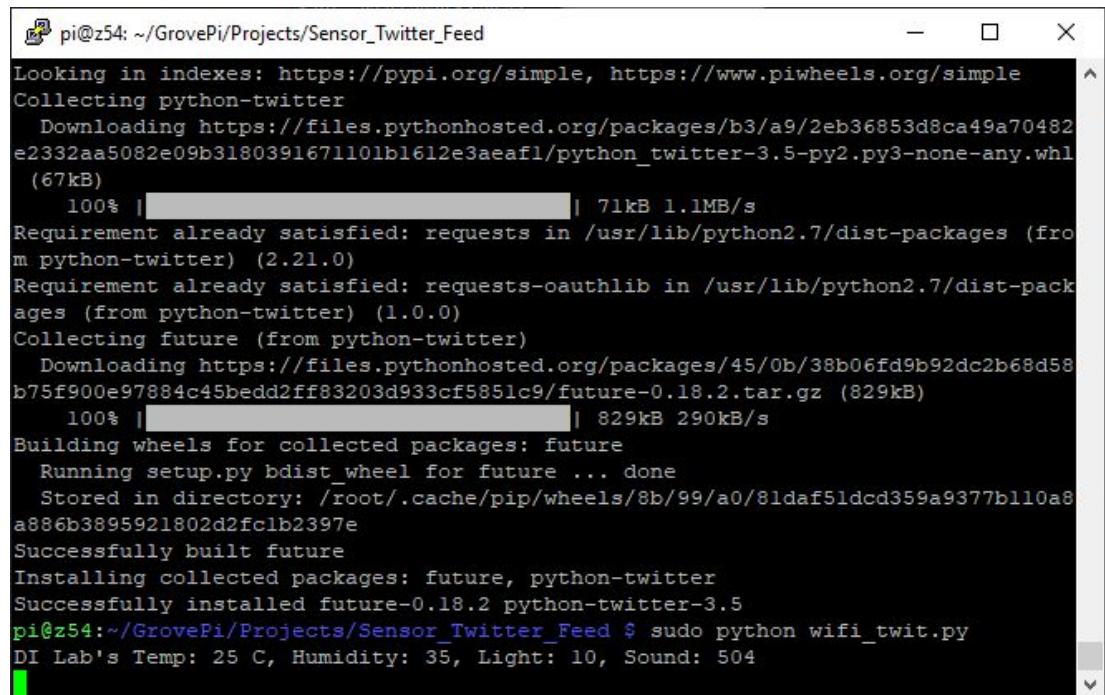
# Post a tweet
out_str ="%s Temp: %d C, Humidity: %d, Light: %d, Sound: %d ,
Sensor Wert: %d, Spannung: %d V, Winkel: %d Grad, Helligkeit = %d"
%(intro_str,t,h,light_intensity/10,sound_level,sensor_value,voltage,degrees,brightness)
    print (out_str)
    api.update_status(out_str)
except IOError:
    print("Error")
except KeyboardInterrupt:
    exit()
except Exception as e:
    print("Duplicate Tweet or Twitter Refusal: {}".format(e))

time.sleep(15)
```

Wenn wir dann das Script ausführen.

```
pi@z54:~ $ sudo python wifi_twit.py
```

Im Terminal sieht dann das so aus.

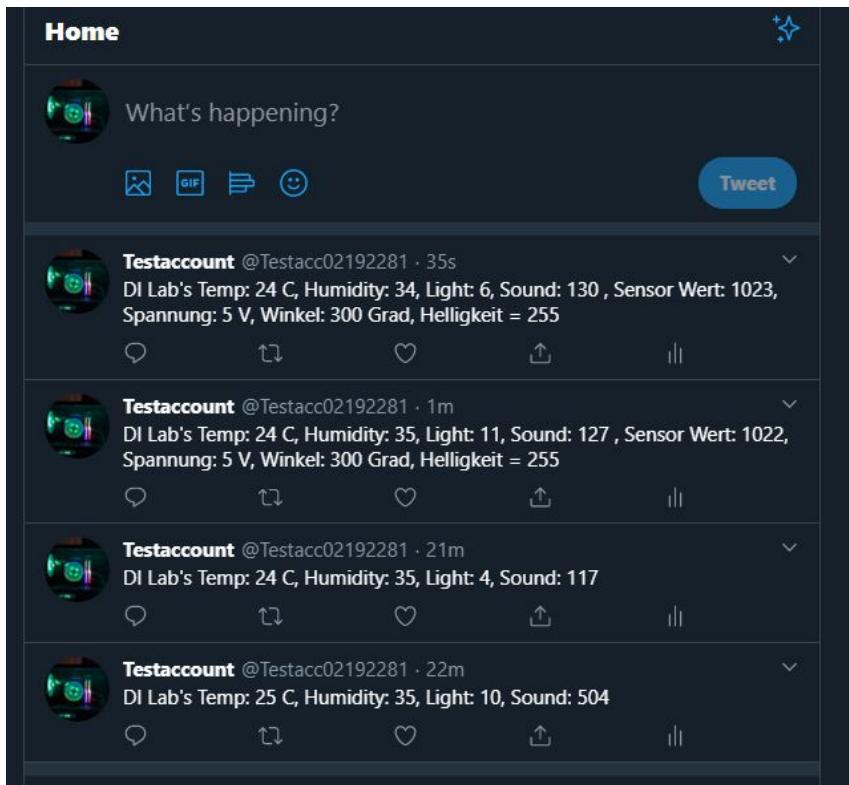


```
pi@z54: ~/GrovePi/Projects/Sensor_Twitter_Feed
Looking in indexes: https://pypi.org/simple, https://www.piwheels.org/simple
Collecting python-twitter
  Downloading https://files.pythonhosted.org/packages/b3/a9/2eb36853d8ca49a70482e2332aa5082e09b3180391671101b1612e3aeaf1/python_twitter-3.5-py2.py3-none-any.whl (67kB)
    100% |██████████| 71kB 1.1MB/s
Requirement already satisfied: requests in /usr/lib/python2.7/dist-packages (from python-twitter) (2.21.0)
Requirement already satisfied: requests-oauthlib in /usr/lib/python2.7/dist-packages (from python-twitter) (1.0.0)
Collecting future (from python-twitter)
  Downloading https://files.pythonhosted.org/packages/45/0b/38b06fd9b92dc2b68d58b75f900e97884c45bedd2ff83203d933cf5851c9/future-0.18.2.tar.gz (829kB)
    100% |██████████| 829kB 290kB/s
Building wheels for collected packages: future
  Running setup.py bdist_wheel for future ... done
  Stored in directory: /root/.cache/pip/wheels/8b/99/a0/81daf51dcd359a9377b110a8a886b3895921802d2fc1b2397e
Successfully built future
Installing collected packages: future, python-twitter
Successfully installed future-0.18.2 python-twitter-3.5
pi@z54:~/GrovePi/Projects/Sensor_Twitter_Feed $ sudo python wifi_twit.py
DI Lab's Temp: 25 C, Humidity: 35, Light: 10, Sound: 504
```

Und auf Twitter dann folgendermassen



Das Script wird automatisch alle 60 Sekunden ausgeführt. Hier sieht man dann auch die Daten des Drehpotentiometers.



Im Terminal sieht es dann folgendermassen aus.

```
pi@z54: ~/GrovePi/Projects/Sensor_Twitter_Feed
a886b3895921802d2fc1b2397e
Successfully built future
Installing collected packages: future, python-twitter
Successfully installed future-0.18.2 python-twitter-3.5
pi@z54:~/GrovePi/Projects/Sensor_Twitter_Feed $ sudo python wifi_twit.py
DI Lab's Temp: 25 C, Humidity: 35, Light: 10, Sound: 504
DI Lab's Temp: 24 C, Humidity: 35, Light: 4, Sound: 117
^X^CTraceback (most recent call last):
  File "wifi_twit.py", line 91, in <module>
    time.sleep(60)
KeyboardInterrupt
pi@z54:~/GrovePi/Projects/Sensor_Twitter_Feed $ sudo python wifi_twit.py
  File "wifi_twit.py", line 115
    out_str ="%s Temp: %d C, Humidity: %d, Light: %d, Sound: %d",Sensor Wert:",
    %d, ", Spannung:", %d, "V, ==> Winkel:", %d, "Grad, Helligkeit = ", %d" %(intro_
str,t,h,light_intensity/10,sound_level,sensor_value,voltage,degrees,brightness)
                                         ^
SyntaxError: invalid syntax
pi@z54:~/GrovePi/Projects/Sensor_Twitter_Feed $ sudo python wifi_twit.py
DI Lab's Temp: 24 C, Humidity: 35, Light: 11, Sound: 127 , Sensor Wert: 1022, Sp
annung: 5 V, Winkel: 300 Grad, Helligkeit = 255
DI Lab's Temp: 24 C, Humidity: 34, Light: 6, Sound: 130 , Sensor Wert: 1023, Sp
annung: 5 V, Winkel: 300 Grad, Helligkeit = 255
```

Zusatz Digitec Preis Twitter

Nun werden wir zuerst die folgenden Libraries installieren.

```
pi@z54:~ $ sudo apt-get install python-pandas
pi@z54:~ $ sudo apt-get install python-bs4
```

Danach habe ich folgendes Script erstellt.

```
import tweepy
import time
import grovepi
import math
import requests
from bs4 import BeautifulSoup
from pandas import DataFrame

intro_str = "Price for the best Monitor:"

# Connect to Twitter
auth = tweepy.OAuthHandler("opFVEGL9dxkNloDiBhdJ9ij55",
                            "Qf5LCmHXaB6Nwdf1DKhywlyrtZ88TBeFE2L30Vh1bFHVuCgvSD")
auth.set_access_token("1237028090221596672-71wcdFHSLoMwt963auTs7dMYNGiWuP",
                        "2hE6UePyPfexOwqln8LmVb7qAlpWts32KRGRdOUCbBINM")
api = tweepy.API(auth)

URL =
'https://www.digitec.ch/de/s1/product/aoc-c24g1-24-1920-x-1080-pixels-monitor-9104879'

headers = {"User-Agent": 'Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/78.0.3904.97 Safari/537.36'}

def check_price():
    page = requests.get(URL, headers=headers)

    soup = BeautifulSoup(page.content, 'html.parser')
    price = soup.select("strong.ZZcn")[0].get_text()
    product = soup.select("h1.productName.ZZdf")[0].get_text()

    converted_price = float(price[0:5])

    # Post a tweet
    out_str = "%s Price: %d CHF, Name: %s"
    %(intro_str,converted_price,product)
    print (out_str)
    api.update_status(out_str)

    time.sleep(30)

check_price()
```

Dieses Script geht auf eine Seite und holt die im Script definierten Werte und gibt diese mittels eines Twitter Tweets aus. Dies ist der erste tweet den ich damit gemacht habe.



Als Beweis ist dies der Artikel auf Digitec und man sieht die Daten sind identisch.

The image shows a product page for the AOC C24G1 monitor on the Digitec website. The main product image displays the monitor with its curved screen and a fiery motorcycle graphic. Key features highlighted include 'Curved' and '144Hz FreeSync2'. The price is listed as 229.-. The monitor has a rating of 4.5 stars based on 38 reviews and 45 questions. It is categorized under 'AOC · Monitor'. A yellow callout box notes that despite high demand, the company is working to maintain delivery by March 27th. A green checkmark indicates a delivery date of Friday, March 27, 2020, if ordered by Monday at 13:00. A blue button allows users to add the item to their shopping cart. Other buttons for 'Vergleichen' (Compare) and 'Merken' (Bookmark) are also visible.

W10e GrovePi LCD

Ich habe für diesen Werkstattauftrag insgesamt drei Scripts geschrieben. Bei diesem Werkstattauftrag hatte ich keine Vorgaben, das Entwickelte Programm sollte einfach innovativ sein und etwas neues. Insgesamt habe ich etwa drei Stunden daran gesessen.

Erstes Script:

```
from grove_rgb_lcd import *
import time

setRGB(0,255,0)
buf=list("My Name is Luis Luescher")
setText("".join(buf))
time.sleep(1)

for i in range(len(buf)):
    buf[i] ="."
    setText_norefresh("".join(buf))
    time.sleep(.1)
```

Das dazugehörige Video auf meiner [Website](#).

Zweites Script ca. 150 Zeilen:

```
from grove_rgb_lcd import *

import random

try:
    setRGB(0,255,0)

    setText("Hi my Name is Luis")
    time.sleep(2)

    setText("I'm 17 years old")
    time.sleep(2)

    setText("Random colors")
    for i in range(0,51):

        setRGB(random.randint(0,255),random.randint(0,255),random.randint(0,255))
            time.sleep(.1)
        time.sleep(1)

        # ascii char 255 is the cursor character
        setRGB(255,255,255)
        setText(chr(255)*32)
```

```
time.sleep(2)

# typewriter
setRGB(255,127,0)
str = "Hello TBZ"
for i in range(0,12):
    setText(str[:i])
    time.sleep(.2)
time.sleep(2)

chars = ""
for a in range(32,256):
    chars += chr(a)
    if len(chars) == 32:
        setText(chars)
        chars = ""
        time.sleep(2)

setRGB(0,255,0)
setText("Solid colors")
time.sleep(2)

setText("Red")
setRGB(255,0,0)
time.sleep(.5)

setText("Green")
setRGB(0,255,0)
time.sleep(.5)

setText("Blue")
setRGB(0,0,255)
time.sleep(.5)

setText("Yellow")
setRGB(255,255,0)
time.sleep(.5)

setText("Magenta")
setRGB(255,0,255)
time.sleep(.5)

setText("Cyan")
setRGB(0,255,255)
time.sleep(.5)

setText("White")
```

```
setRGB(255,255,255)
time.sleep(.5)

setText("Black")
setRGB(0,0,0)
time.sleep(.5)

setText("Grey")
setRGB(127,127,127)
time.sleep(.5)

setRGB(255,255,255)
setText("Raspberry Pi")
time.sleep(2)

setText("Grove Pi")
time.sleep(2)

setText("modul 126")
time.sleep(2)

setText("Guten Tag")
time.sleep(2)

setText("www.luis-luescher.com")
time.sleep(2)

setText("Shades of red")
for c in range(0,255):
    setRGB(255,255-c,255-c)
    time.sleep(.01)

setText("Shades of green")
for c in range(0,255):
    setRGB(255-c,255,255-c)
    time.sleep(.01)

setText("Shades of blue")
for c in range(0,255):
    setRGB(255-c,255-c,255)
    time.sleep(.01)

setText("Shades of yellow")
for c in range(0,255):
    setRGB(255,255,255-c)
    time.sleep(.01)
```

```
setText("Shades of magenta")
for c in range(0,255):
    setRGB(255,255-c,255)
    time.sleep(.01)

setText("Shades of cyan")
for c in range(0,255):
    setRGB(255-c,255,255)
    time.sleep(.01)

setText("Shades of grey")
for c in range(0,255):
    setRGB(c,c,c)
    time.sleep(.01)

except KeyboardInterrupt:
    setText("KeyboardInterrupt")
    setRGB(255,0,0)
except IOError:
    setText("IOError")
    setRGB(255,0,0)

time.sleep(1)
setText("Go away")
setRGB(0,255,0)
```

Das dazugehörige Video auf meiner [Website](#).

Zudem habe ich mein Digitec Script so optimiert, dass nun der Preis auf dem LCD Bildschirm angezeigt.

```
from grove_rgb_lcd import *
import tweepy
import time
import grovepi
import math
import requests
from bs4 import BeautifulSoup
from pandas import DataFrame

URL =
'https://www.digitec.ch/de/s1/product/raspberry-pi-gehaeuse-zu-raspberry-pi-4-model-b-gehaeuse-elektronikzubehoer-gehaeuse-12257287'

headers = {"User-Agent": 'Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/78.0.3904.97 Safari/537.36' }

def check_price():
```

```
page = requests.get(URL, headers=headers)

soup = BeautifulSoup(page.content, 'html.parser')
price = soup.select("strong.ZZcn")[0].get_text()
product = soup.select("h1.productName.ZZdf")[0].get_text()

converted_price = float(price[0:5])

out_str ="Price: %d CHF, Name: %s" %(converted_price,product)
print (out_str)
setText(out_str)
setRGB(0,128,64)

check_price()
```

Folgenden Output hat es dann im LCD Display gegeben:



W16 HiFi-Berry & WebRadio

Zu Beginn werden wir den HiFiberry konfigurieren. Und danach werden wir einer Example WAV Datei machen. Wir öffnen nun folgendes File. Und kommentieren eine spezifische Zeile aus und fügen dann eine Codezeile hinzu.

```
pi@z54:~ $ sudo nano /boot/config.txt
#dtparam=audio=on
dtoverlay=hifiberry-digi
```

Danach erstellen wir ein neues File und fügen folgenden Code ein.

Luis Ricardo Lüscher
M126 - Peripheriegeräte im Netzwerkbetrieb einsetzen

```
pi@z54:~ $ sudo nano /etc/asound.conf
pcm.!default {
    type hw card 0
}
ctl.!default {
    type hw card 0
}
```

Nun installieren wir den mPlayer.

```
pi@z54:~ $ sudo apt-get install mplayer
```

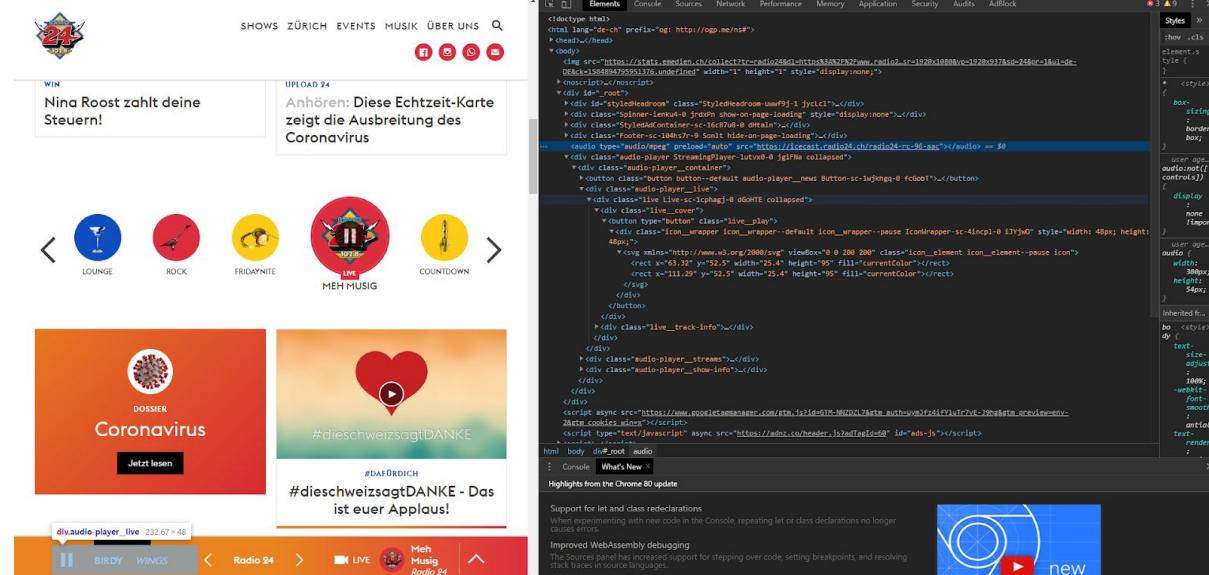
Danach führen wir eine Example WAV Datei aus. Folgender [Videobeweis](#).

```
pi@z54:~ $ aplay file_example_WAV_2MG.wav
```

Nun lassen wir noch einen lokalen Radiosender laufen. Folgender [Videobeweis](#).

```
pi@z54:~ $ mplayer
https://energybasel.ice.infomaniak.ch/energybasel-high.mp3
```

Die Links kann man auf der Seite der jeweiligen Radiosender beim Untersuchen des HTML Codes finden.



The screenshot shows a web browser's developer tools with the 'Elements' tab selected. The page content displays a radio station's website with various buttons for different music genres like Lounge, Rock, and Countdown. Below this is a section for the coronavirus, and at the bottom, there are links for 'Birdy' and 'Radio 24'. The developer tools' sidebar on the right shows the current file path: 'index.html' under 'HTML'. The main pane shows the HTML code for the page, with a specific line highlighted:

```
<audio type="audio/mpeg" preload="auto" src="https://icecast.radio24.ch/radio24-rc-96-aac"></audio>
```

This line represents the audio stream being played by the mplayer command.

Nun werden wir noch einen Musikserver einrichten.

Dazu werden wir zuerst die nötigen gpg Schlüssel hinzufügen und noch das Repository zu den Paketquellen hinzufügen.

```
pi@z54:~ $ wget -q -O - https://apt.mopidy.com/mopidy.gpg | sudo apt-key add -
```

```
pi@z54:~ $ sudo wget -q -O /etc/apt/sources.list.d/mopidy.list  
https://apt.mopidy.com/stretch.list
```

Nun aktualisieren wir die Paketquelle und installieren dann mopidy

```
pi@z54:~ $ sudo apt-get update  
pi@z54:~ $ sudo apt-get install mopidy
```

Danach öffnen wir das Konfigurationsfile und fügen folgende Parameter ein.

```
pi@z54:~ $ sudo nano .config/mopidy/mopidy.conf  
[spotify]  
username = luisricardoluescher@gmail.com  
password = SAFEPASS  
client_id = XXXXXXXXXXXXXXXXXXXXXXXXX  
client_secret = XXXXXXXXXXXXXXXXXXXXXXXXX  
  
[spotify_tunigo]  
enabled = true  
  
[mopify]  
enabled = true  
debug = false
```

Und danach Installieren wir noch folgende Pakete.

```
pi@z54:~ $ sudo apt-get install mopidy-spotify-tunigo  
pi@z54:~ $ sudo pip install Mopidy-Mopify  
pi@z54:~ $ sudo python3 -m pip install mopidy-spotify-tunigo
```

Wann alles erfolgreich installiert wurde muss man den Service neustarten.

```
pi@z54:~ $ sudo service mopidy restart
```

Danach kann man im Browser den Server öffnen über den Port 6680.



localhost:6680/mopidy/

Danach sollte unter dem Punkt "Web clients" ein wählbares Modul sein. In diesem Fall mopify.

Mopidy

This web server is a part of the Mopidy music server. To learn more about Mopidy, please visit www.mopidy.com.

Web clients

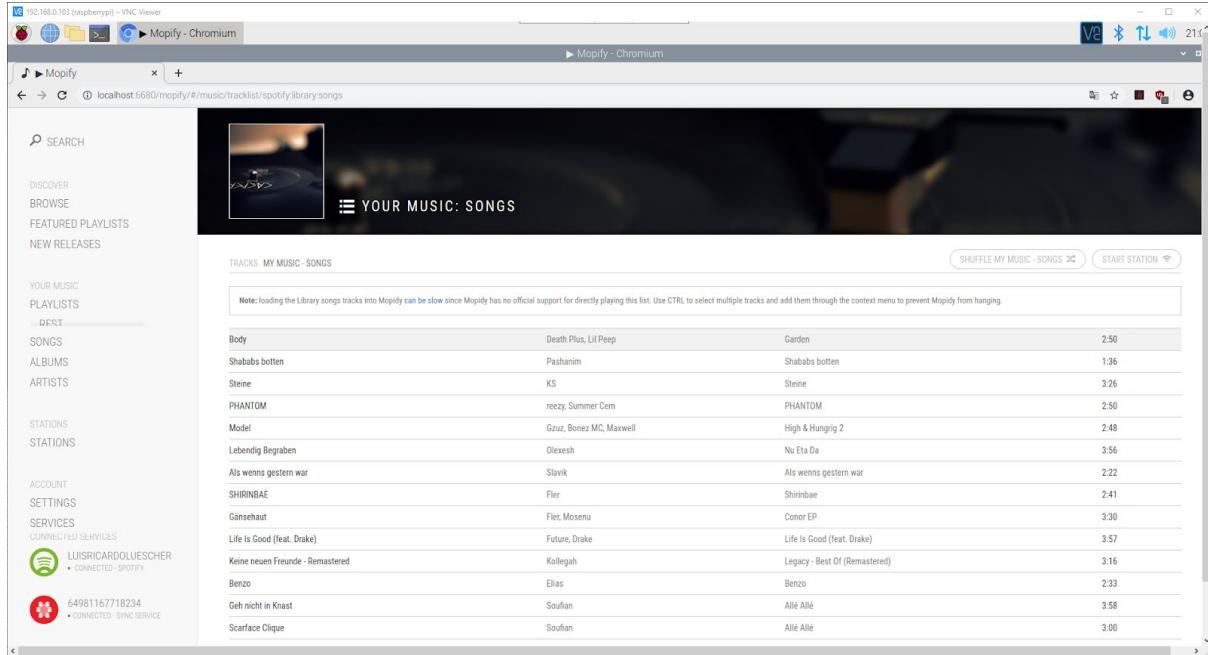
- [mopify](#)

Web clients which are installed as Mopidy extensions will automatically appear here.

Danach verbindet man sich mit seinem Spotify Account. Man muss nur schnell den Zugriff der App bzw. des Server bestätigen.

The screenshot shows the Mopidy web interface. On the left sidebar, there are several menu items: YOUR MUSIC, PLAYLISTS, REST, SONGS, ALBUMS, ARTISTS, STATIONS, ACCOUNT, SETTINGS, SERVICES, and CONNECTED SERVICES. Under CONNECTED SERVICES, it says "You have no connected services." and "Click here to connect a service." In the main content area, there is a "SERVICES" section with a note: "Note: browsers are likely to block the login popups". Below this is a large green button featuring the white Spotify logo. To the right of the button, the word "Spotify" is followed by a small "•" and the text "Search and manage playlists and get the latest charts".

Danach sollte man dann ohne Problem können auf seine Playlists zugreifen und Musik hören.



W24 RASPAP

Mithilfe von RaspAP kann man ganz einfach ein eigenes Gastnetzwerk machen, insofern der eigene Router eine solche Funktion nicht anbietet. Das Programm RaspAP übernimmt die Installation und vereinfacht die Konfiguration über eine Weboberfläche.

Zu Beginn werden wir das Installationsskript aus dem Internet herunterladen. Dafür verwenden wir folgenden Befehl.

```
pi@z54:~ $ wget -q https://git.io/v0EUQ -O /tmp/raspap
```

Danach geben wir dem Installationsskript noch die Ausführungsrechte (x für execute.)

```
pi@z54:~ $ chmod +x /tmp/raspap
```

Und danach kann man das Script ausführen.

```
pi@z54:~ $ /tmp/raspap
```

Danach kann man alle im Installationsprozess auftauchenden Parameter beim Default hier im Terminal als recommend markiert lassen.

Luis Ricardo Lüscher
M126 - Peripheriegeräte im Netzwerkbetrieb einsetzen

The terminal window shows the following steps:

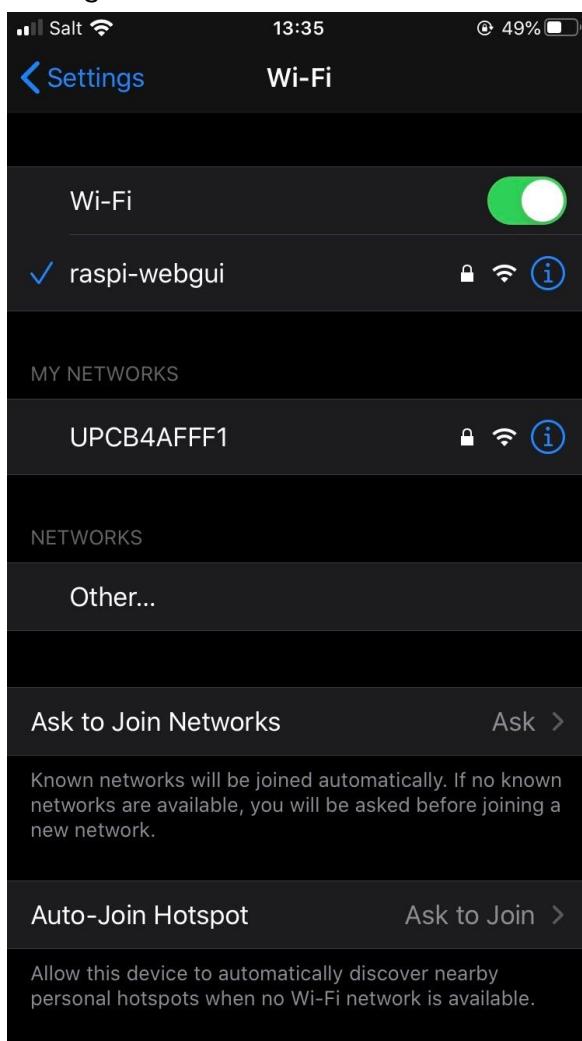
```
pi@raspberrypi: ~
BEendet --2020-03-23 13:05:46--
Verstrichene Zeit: 1.2s
Geholt: 1 Dateien, 4.5K in 0s (17.1 MB/s)
pi@raspberrypi:~ $ wget -q https://git.io/voEUQ -O /tmp/raspap
pi@raspberrypi:~ $ chmod +x /tmp/raspap
pi@raspberrypi:~ $ /tmp/raspap

 888888ba          .d888888  888888ba
 88   8b          d8   88   88   8b
a88aaaaa8P' .d8888b. .d8888b. 88d888b. 88aaaaaa8a a88aaaaa8P
 88   8b. 88   88 Y8oooooo. 88   88   88   88
 88   88 88. 88   88 88. 88 88   88   88
dP   dP 888888P 888888P 88Y8888P 88   88   dP
                                88
                                dP      version 2.2

The Quick Installer will guide you through a few easy steps

RaspAP Install: Configure installation
Detected Raspbian 10.0 (Buster)
Install directory: /etc/raspap
Install to Lighttpd root directory: /var/www/html? [Y/n]:
```

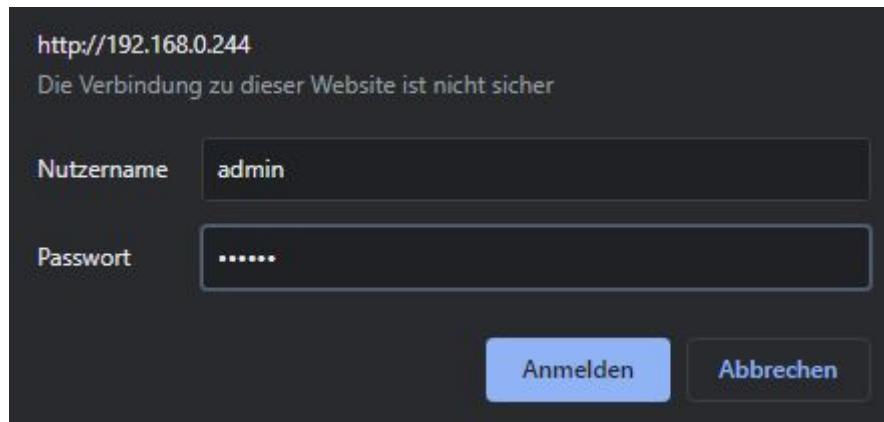
Nun kann man sich bereits mit dem Access Point verbinden. Das Passwort dazu ist.
"ChangeMe"



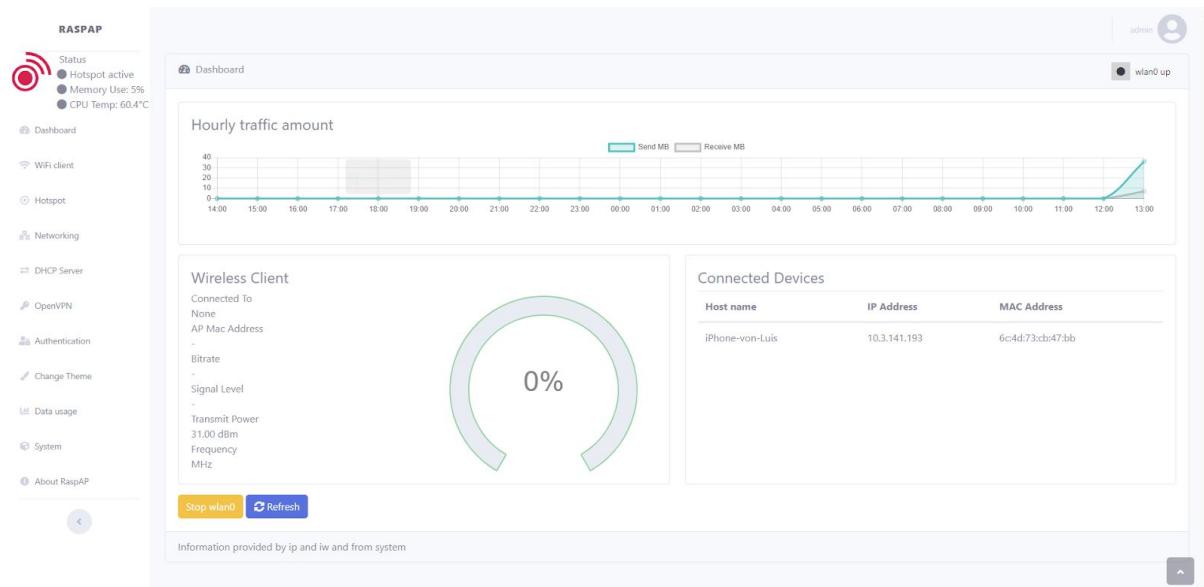
Nun kann man dann auch auf die Weboberfläche zugreifen. Dazu einfach die Weboberfläche via http://RASPI_IP öffnen.



Beim öffnen der Website wird man nach einem Login gefragt. Folgende Credentials:
Username: admin PW: secret.



Und nach der erfolgreichen Authentifizierung öffnet sich dann die Weboberfläche.



Leider ist es so, dass nun nur die sich im Netzwerk befindlichen Geräte kommunizieren können. Das Aufrufen einer Website ist in diesem Stand nicht möglich. Aufgrund dessen, dass das Weiterleiten von Paketen von der WLAN Schnittstelle auf den Ethernet Port und somit ins Internet nicht vorhanden ist. (Masquerading)
Dafür öffnen wir zuerst eine Konfigurationsdatei. Und entfernen die rauten vor
`#net.ipv4.ip_forward=1`

```
pi@z54:~ $ sudo nano /etc/sysctl.conf
net.ipv4.ip_forward=1
```

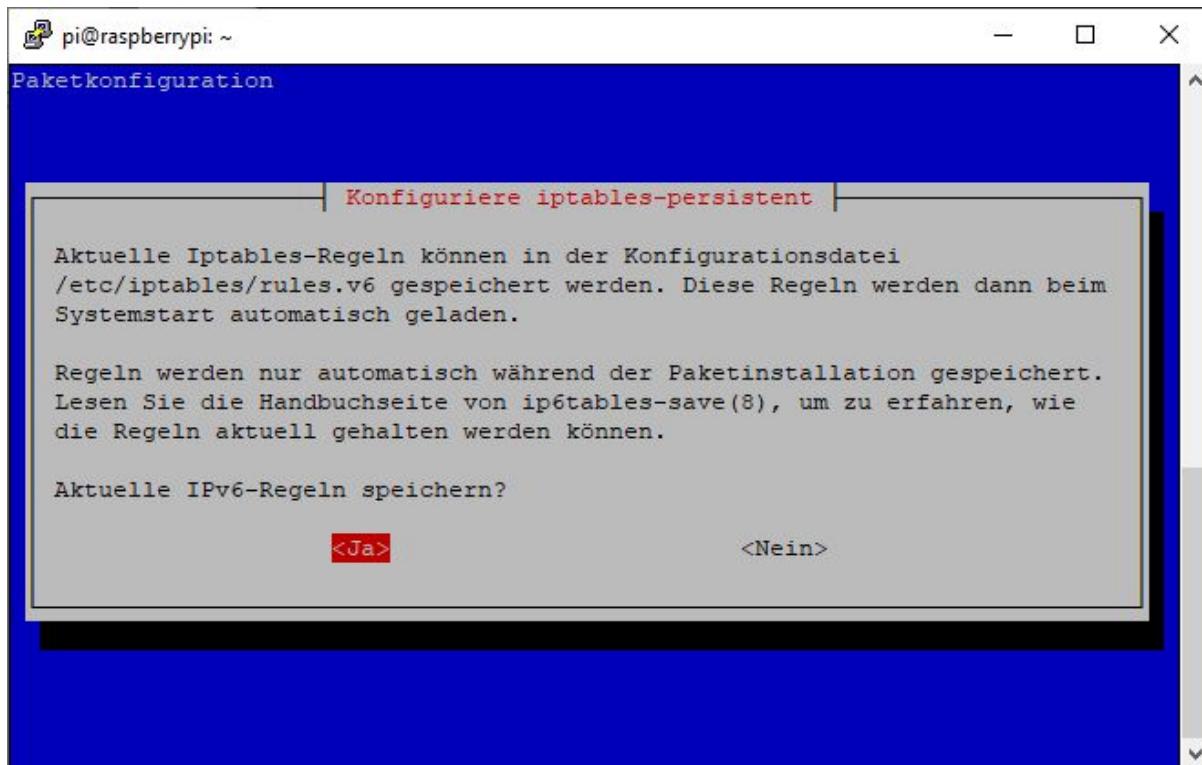
Zudem müssen wir die Iptables-Firewall noch updaten. Durch die Drop Anweisungen geben wir den Geräten die sich mit dem Gast AP verbinden, keine Chance sich mit dem eigenen Netz zu verbinden.

```
pi@z54:~ $ sudo iptables -t nat -A POSTROUTING -j MASQUERADE
pi@z54:~ $ sudo iptables -I FORWARD -d 10.0.0.0/8 -j DROP
pi@z54:~ $ sudo iptables -I FORWARD -d 172.16.0.0/12 -j DROP
pi@z54:~ $ sudo iptables -I FORWARD -d 192.168.0.0/16 -j DROP
```

Nun installieren wir noch das iptables Resistent Paket für die permanente Hinterlegung der Firewall Regeln.

```
pi@z54:~ $ sudo apt install iptables-persistent
```

Während der Installation öffnet sich ein Dialogfenster. Diese beiden Meldungen einfach mit "Yes" bestätigen.



Danach werden wir die Firewall Regeln noch abspeichern.

```
pi@z54:~ $ sudo iptables -F
pi@z54:~ $ sudo sh -c "iptables-save > /etc/iptables/rules.v4"
```

Nun werden wir noch die SSID und das Passwort ändern, damit der Zugang etwas sicherer ist. Unter dem Punkt "Hotspot" findet man dann unter "Security" das Feld mit dem Passwort (PSK), dieses kann man nach seinem Empfinden ändern und dann mit "Save settings"

speichern.

The screenshot shows the 'Security' tab selected in the top navigation bar. On the left, a sidebar lists various configuration options: Dashboard, WiFi client, Hotspot (selected), Networking, DHCP Server, OpenVPN, Authentication, Change Theme, Data usage, and System. The main content area is titled 'Security settings' and contains three dropdown menus: 'Security type' (set to WPA2), 'Encryption Type' (set to CCMP), and 'PSK' (set to 'ChangeMe'). At the bottom are three buttons: 'Save settings' (blue), 'Stop hotspot' (yellow), and 'Restart hotspot' (yellow).

Zudem können wir noch die SSID ändern unter dem Punkt Basic.

The screenshot shows the 'Basic' tab selected in the top navigation bar. The main content area is titled 'Basic settings' and contains several configuration fields: 'Interface' (set to wlan0), 'SSID' (set to 'm126-raspiap'), 'Wireless Mode' (set to '802.11g - 2.4 GHz'), and 'Channel' (set to 1). At the bottom are three buttons: 'Save settings' (blue), 'Stop hotspot' (yellow), and 'Restart hotspot' (yellow).

Die Funktionalität des AP kann mittels diesem [Video](#) bewiesen werden.

Eigene Projekte

Im Abschnitt “Eigene Projekte” werde ich eigenen projekten nachgehen in Bezug auf den RasPi. Da ich einige Punkte der LernMap als unnötig empfinde, arbeite ich selber lieber an einem Projekt welches mich herausfordert und mir auch einen persönlichen Nutzen daraus bringt. Bereits jetzt hat mein Vater 2 RasPi’s gekauft damit ich einige Dinge für unsere IT-Infrastruktur realisieren kann. Diese Projekte werde ich in diesem Abschnitt dokumentieren und erklären.

W95 PIVPN

Ein VPN erweitert im Wesentlichen Ihr privates lokales Netzwerk auf das Internet, und wenn es richtig eingerichtet ist, ist es sehr sicher. Leider kann diese Einrichtung für den Unerfahrenen ein wenig kompliziert sein. PiVPN beseitigt diese Komplexität auf Ihrer Raspberry Pi, indem es einen Installations- und Konfigurationsprozess bietet, der in einem einzigen Befehl automatisiert wurde. zuerst müssen wir iptable anpassen.

```
pi@z54:~ $ sudo iptables -F
pi@z54:~ $ sudo iptables -P INPUT ACCEPT
pi@z54:~ $ sudo iptables -P OUTPUT ACCEPT
pi@z54:~ $ sudo iptables -P FORWARD ACCEPT
```

Nun können wir die PiVPN Installationsdatei herunterladen.

```
pi@z54:~ $ curl -L https://install.pivpn.io/ | bash
```

Danach startet der Installationsprozess. Diese Meldung mit "OK" bestätigen.

A terminal window titled "pi@raspberrypi: ~". The title bar has icons for minimize, maximize, and close. The window content is a blue-tinted text area. At the top, it says "Welcome". Below that, there is a large amount of placeholder text consisting mostly of 'x' characters. In the center, the text "PiVPN Automated Installer" is printed in red. Below it, the message "This installer will transform your Raspberry Pi into an OpenVPN or WireGuard server!" is displayed. At the bottom center is a red button labeled "<OK>".

Der Pi braucht eine statische IP Adresse. Diese Meldung mit "OK" bestätigen.

A terminal window titled "pi@raspberrypi: ~". The title bar has icons for minimize, maximize, and close. The window content is a blue-tinted text area. At the top, it says "Initiating network interface". Below that, there is a large amount of placeholder text consisting mostly of 'x' characters. In the center, the text "Static IP Needed" is printed in red. Below it, the message "The PiVPN is a SERVER so it needs a STATIC IP ADDRESS to function properly." is displayed. Further down, the message "In the next section, you can choose to use your current network settings (DHCP) or to manually edit them." is shown. At the bottom center is a red button labeled "<OK>".

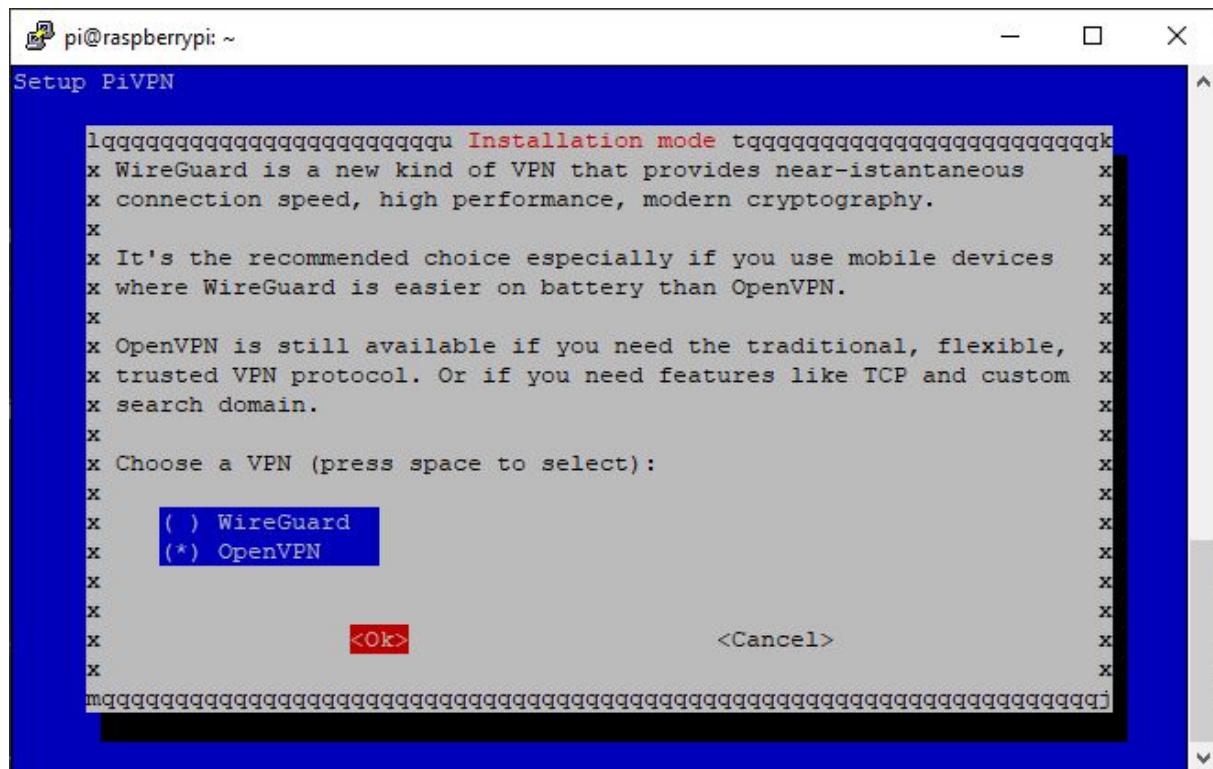
Nun kann man seine aktuelle IP-Konfiguration sehen. Ich habe in diesem Fenster "Yes" ausgewählt, da ich auf der DHCP Adresse verbleiben möchte.

A terminal window titled "pi@raspberrypi: ~". The title bar shows the session name and the command prompt. The window content displays a configuration menu for a network interface. It asks if the user is using DHCP Reservation on their router/server and provides current network settings (IP address: 192.168.0.244/24, Gateway: 192.168.0.1). It also offers three choices: Yes (Keep using DHCP reservation), No (Setup static IP address), and Don't know what DHCP Reservation is? Answer No. The "Yes" option is highlighted in red. At the bottom of the window are two buttons: "<Yes>" and "<No>".

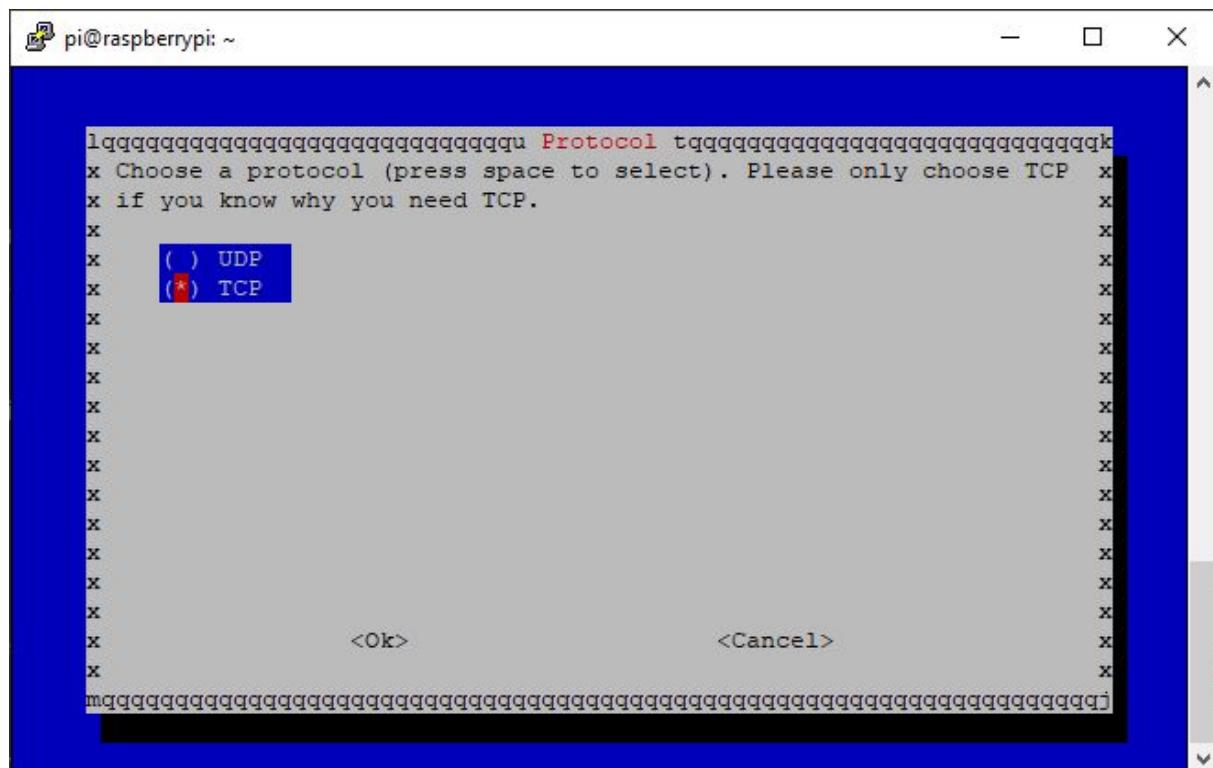
Nun muss man einen user für die VPN Konfiguration auswählen, in meinen Fall den Standartuser "pi".

A terminal window titled "pi@raspberrypi: ~". The title bar shows the session name and the command prompt. The window content displays a menu to choose a user for a VPN configuration. It asks to choose a user by pressing space to select and lists available users. The user "pi" is selected and highlighted in blue. At the bottom of the window are two buttons: "<Ok>" and "<Cancel>".

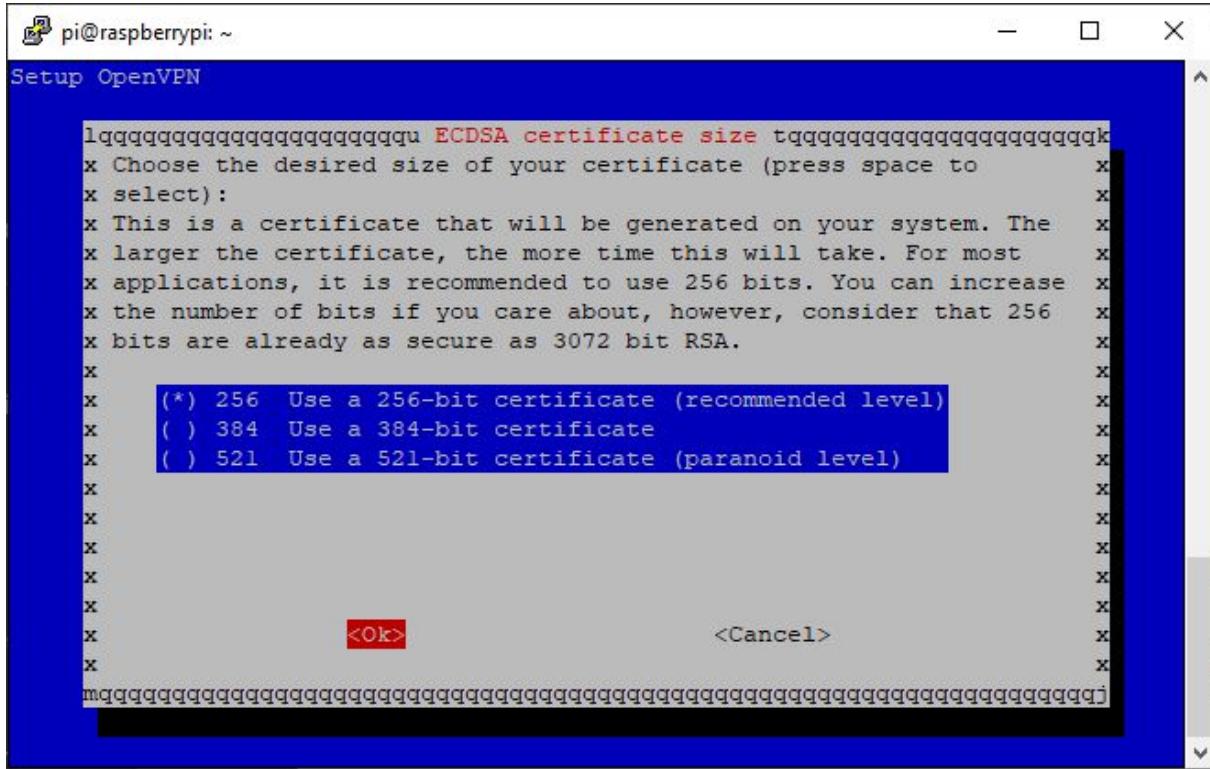
Nun kann man einen VPN auswählen und dann mit "OK" bestätigen. Meine Wahl ist bewusst auf OpenVPN gefallen, da ich bereits ein tiefes Wissen bezüglich OpenVPN habe (Bereits vor einem Jahr eine weit komplizierte Variante des Pi VPN erstellt).



Nun muss man das jeweilige Protokoll auswählen. Ich habe mich für TCP entschieden. Ist der Standard bei VPN's.



Nun muss man die entsprechende Grösse der zertifikate auswählen. Ich habe mich für eine 256-bit Version entschieden, da durch dieses Level der Speed nicht allzu stark beeinträchtigt wird.



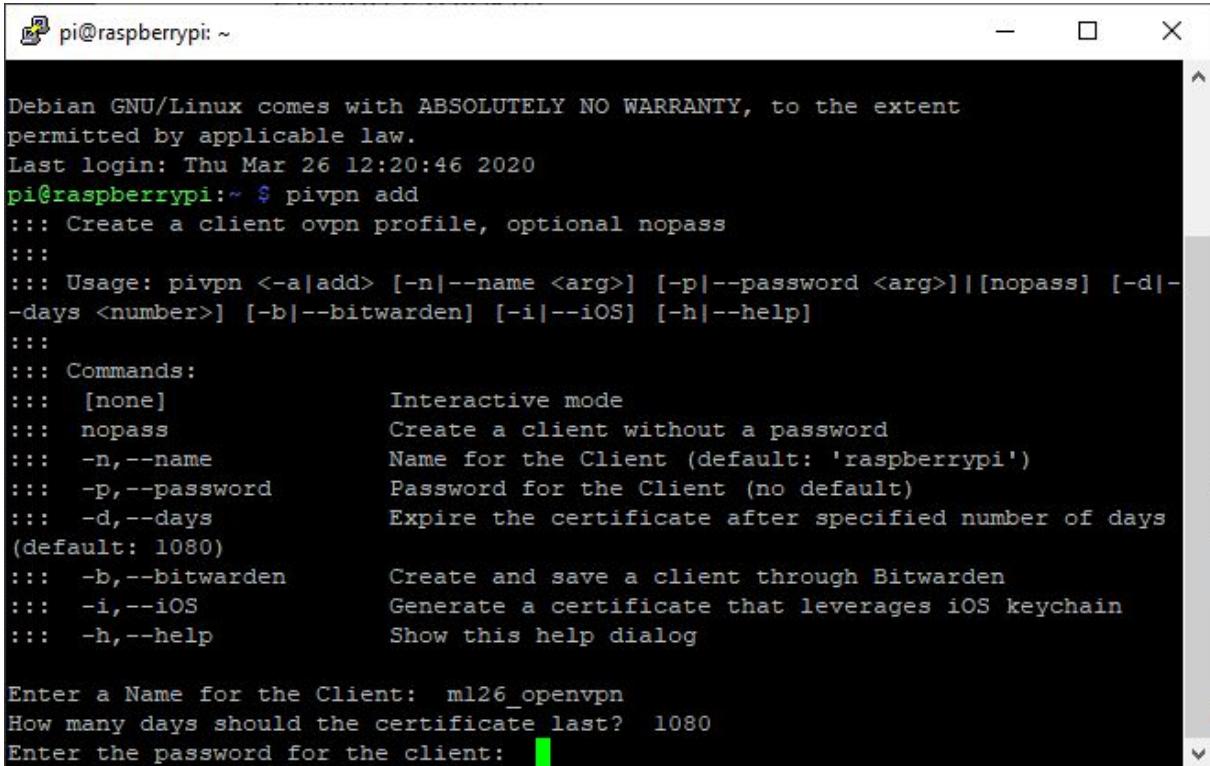
Danach sollte man den RasPi neustarten.



Nun muss man folgenden Befehl ausführen:

```
pi@z54:~ $ pivpn add
```

Nun muss man einige Parameter mitgeben, einerseits den Namen des Zertifikats, dann die Gültigkeit und dann noch das entsprechende passwort für mehr Schutz.

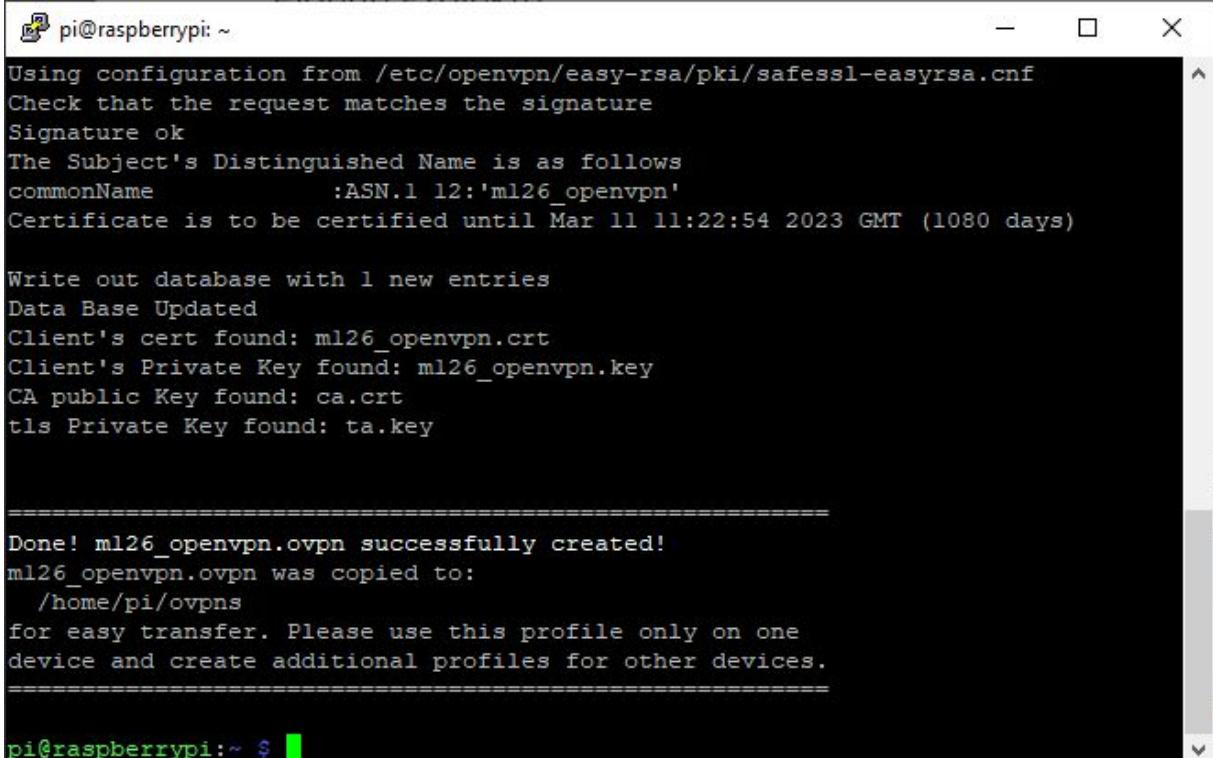


A terminal window titled "pi@raspberrypi: ~" showing the execution of the pivpn add command. The window displays the Debian warranty message, the last login information, and the pivpn help output. It then prompts for the client name, certificate expiration days, and password.

```
pi@raspberrypi:~ $ pivpn add
::: Create a client ovpn profile, optional nopass
:::
::: Usage: pivpn <-a|add> [-n|--name <arg>] [-p|--password <arg>] | [nopass] [-d|-days <number>] [-b|--bitwarden] [-i|--iOS] [-h|--help]
:::
::: Commands:
:::   [none]           Interactive mode
:::   nopass          Create a client without a password
:::   -n,--name        Name for the Client (default: 'raspberrypi')
:::   -p,--password   Password for the Client (no default)
:::   -d,--days        Expire the certificate after specified number of days
:::   (default: 1080)
:::   -b,--bitwarden  Create and save a client through Bitwarden
:::   -i,--iOS         Generate a certificate that leverages iOS keychain
:::   -h,--help        Show this help dialog

Enter a Name for the Client: ml26_openvpn
How many days should the certificate last? 1080
Enter the password for the client: █
```

Wenn alle Daten korrekt sind, sollte eine positive Meldung im Terminal erscheinen.



```
pi@raspberrypi: ~
Using configuration from /etc/openvpn/easy-rsa/pki/safessl-easyrsa.cnf
Check that the request matches the signature
Signature ok
The Subject's Distinguished Name is as follows
commonName          :ASN.1 12:'ml26_openvpn'
Certificate is to be certified until Mar 11 11:22:54 2023 GMT (1080 days)

Write out database with 1 new entries
Data Base Updated
Client's cert found: ml26_openvpn.crt
Client's Private Key found: ml26_openvpn.key
CA public Key found: ca.crt
tls Private Key found: ta.key

=====
Done! ml26_openvpn.ovpn successfully created!
ml26_openvpn.ovpn was copied to:
/home/pi/ovpns
for easy transfer. Please use this profile only on one
device and create additional profiles for other devices.
=====

pi@raspberrypi: ~ $
```

Nun kann man, das neu erstellte opvn-File von seinem Pi auf den eigenen PC transferieren.

Name	Änderungsdatum	Typ	Größe
Bilder	28.11.2019 21:09	Dateiordner	
Daten Notebook	09.02.2020 20:21	Dateiordner	
Drive	20.06.2019 19:26	Dateiordner	
Iso Files	03.01.2020 19:45	Dateiordner	
License	07.02.2020 10:03	Dateiordner	
Projekte	12.01.2020 02:26	Dateiordner	
SIX	24.02.2020 09:30	Dateiordner	
TBZ	09.03.2020 21:52	Dateiordner	
Vorlagen	24.02.2020 09:29	Dateiordner	
Website	22.02.2020 17:25	Dateiordner	
Big_Bang_Theory.txt	09.02.2020 15:00	Textdokument	1 KB
Kroatien.docx	21.02.2020 23:08	Microsoft Word-D...	93 KB
password_camera.txt	14.12.2019 20:02	Textdokument	1 KB
windowsnewclient.ovpn	11.01.2020 19:17	OpenVPN Config ...	5 KB
ml26_openvpn.ovpn	26.03.2020 12:22	OpenVPN Config ...	3 KB

Nun muss man noch auf seinem Router die entsprechende Portweiterleitung setzen.

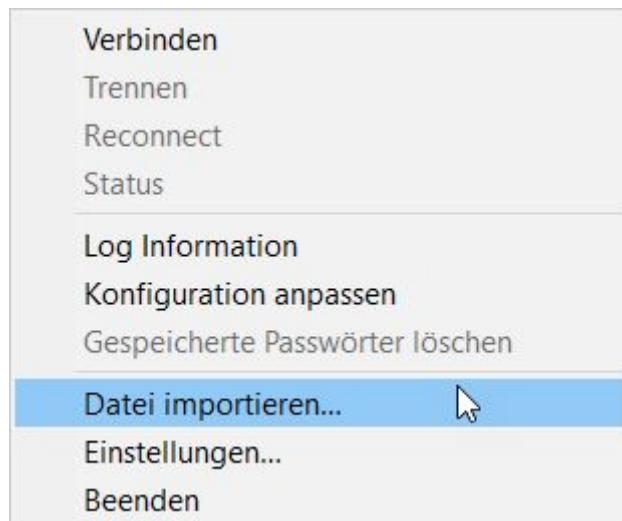
Mit dieser Funktion können eingehende Anfragen auf bestimmte Port-Nummern erlaubt werden, um Web-Server, FTP-Server, Mail-Server usw. zu erreichen:

Lokale IP	192.168.0. <input type="text" value="244"/>
Lokaler Start Port	<input type="text" value="1134"/>
Lokaler End Port	<input type="text" value="1134"/>
Externer Start Port	<input type="text" value="1134"/>
Externer End Port	<input type="text" value="1134"/>
Protokoll	<input type="text" value="TCP"/>
Aktiviert	<input type="text" value="An"/>

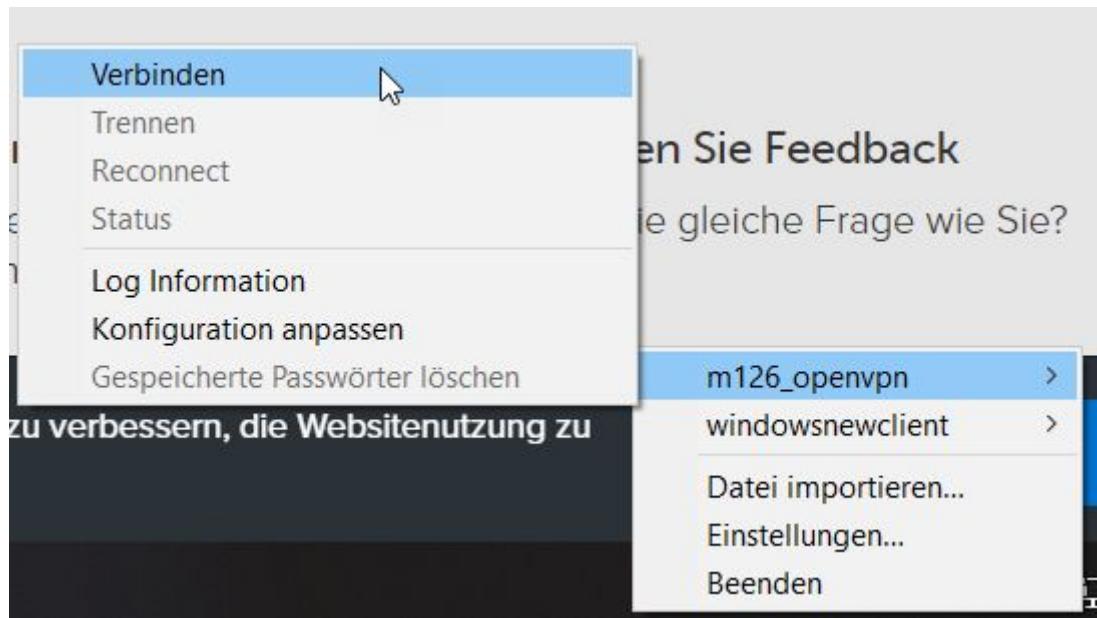
Abbrechen

Regel hinzufügen

Nun muss man das OpenVPN Gui installieren von folgender [Website](#). Und danach kann man das erstellte OpenVPN File in das Gui importieren.



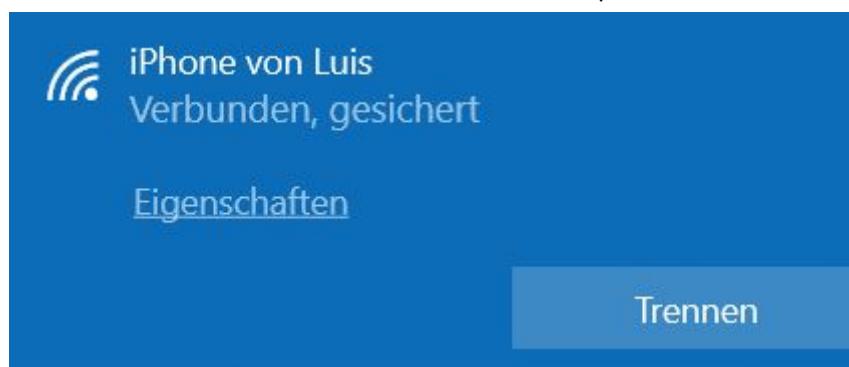
Danach kann man sich mit der jeweiligen Datei in sein Netzwerk verbinden.



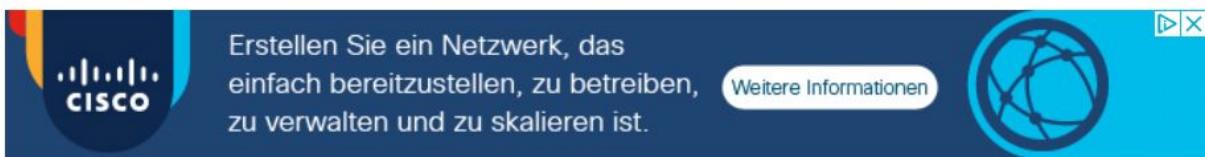
Nun noch das entsprechende Passwort eingeben.



Dann habe ich mich mit meinem Mobile-Hotspot verbunden.



Zuvor hatte ich noch eine öffentliche IP meines Provider, nämlich Salt.



Your Public IPv4 Address Is: 213.55.224.249

Your Public IPv6 is: Not Detected

What Is My Public IP Address?

Und nach der erfolgreichen Verbindung habe ich eine öffentliche Ip meines Heimnetzwerkes.

A screenshot of the WhatIsMyIP.com website. At the top, there's a navigation bar with links for "Ask A Question", "Questions & Answers", "IP Tools", "How To", "Knowledge Base", "General", and "Resources". Below the navigation is a Cisco advertisement banner with the text "Bénéficiez d'un réseau facile à déployer, à exploiter, à gérer et à faire évoluer." and a "En savoir plus" button. The main content area shows the text "Your Public IPv4 Address Is: 77.58.97.212" and a section titled "What Is My Public IP Address?". This section contains a detailed explanation of what a public IP address is and why it's important. At the bottom of the page are buttons for "Port Scanner", "Hide My IP", "IP Blacklist Check", and "Speed Test". A small note at the bottom left says "Warten auf s02mdn.net...".

W96 Plesk Media Server

Nun fügen wir zuerst die entsprechenden Keys hinzu und fügen das neue Repository zu unseren Ressourcen hinzu, sprich um die Paketliste zu aktualisieren.

```
pi@z54:~ $ sudo apt-get update && sudo apt-get upgrade
pi@z54:~ $ sudo apt-get install apt-transport-https
pi@z54:~ $ curl https://downloads.plex.tv/plex-keys/PlexSign.key | sudo apt-key add -
pi@z54:~ $ echo deb https://downloads.plex.tv/repo/deb public main | sudo tee /etc/apt/sources.list.d/plexmediaserver.list
pi@z54:~ $ sudo apt-get update
pi@z54:~ $ sudo apt-get install plexmediaserver
```

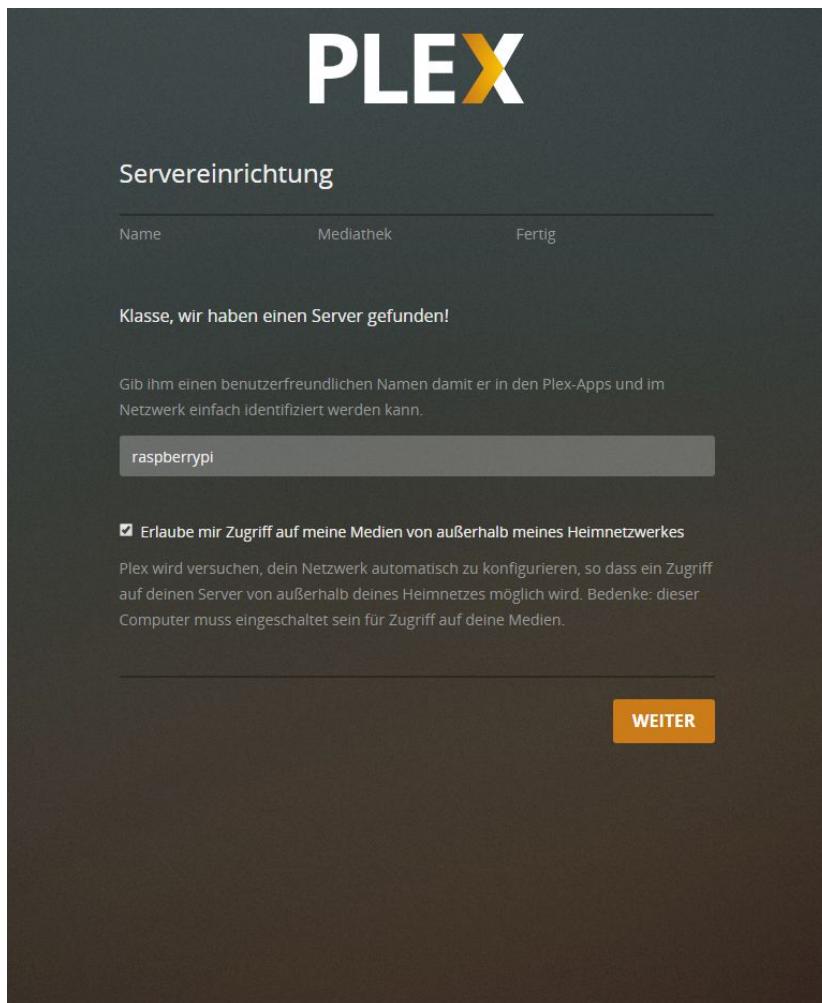
Nun kann man das Plex Media Server Paket installieren.

```
pi@z54:~ $ sudo apt-get install plexmediaserver
```

Danach kann man den Plex Server auf einem beliebigen Browser öffnen.

192.168.0.244:32400/web/

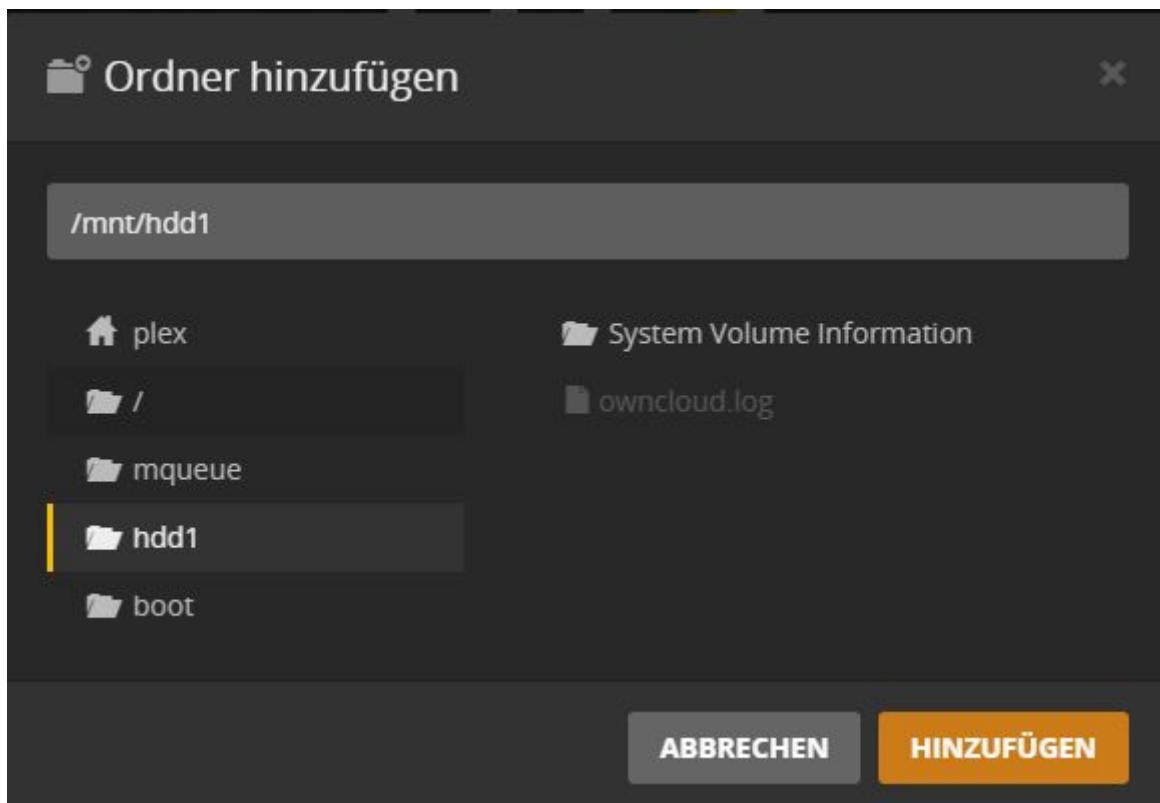
Nun kann man den Server über das Web konfigurieren. Hier einfach den gewünschten Servernamen angeben.



Danach den entsprechenden Ordner bzw. Mediatheke auswählen..



Im öffnenden Fenster dann den entsprechenden ordner auf dem System auswählen.

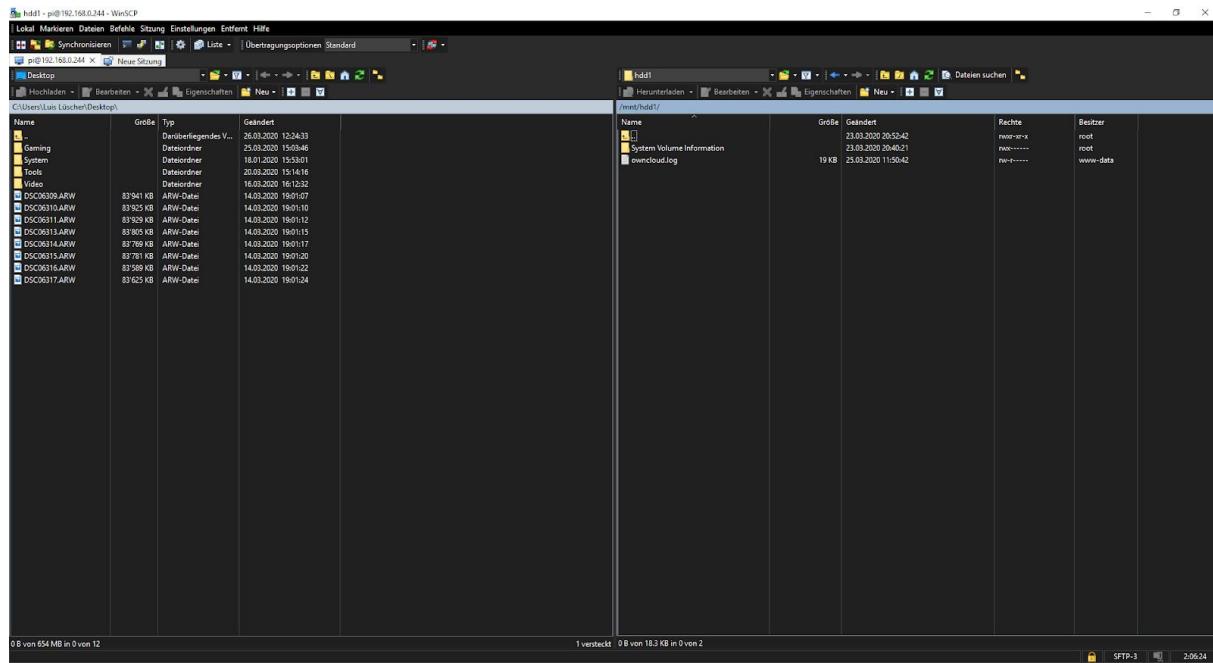


Danach kann man noch die Plex Media Applikation herunterladen zB. auf seinen Computer oder sein Handy.

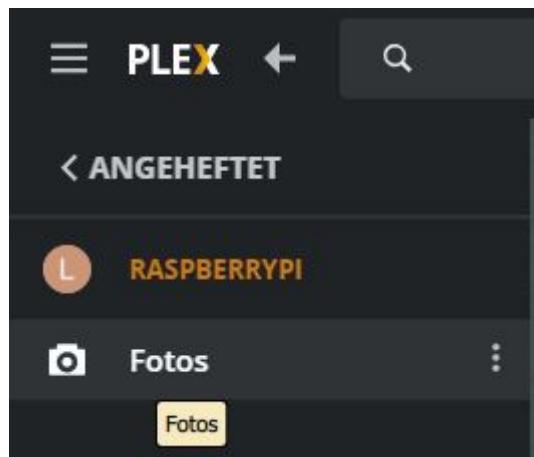
The screenshot shows a web page for downloading Plex for Windows. On the left, there's a sidebar with "Apps und Geräte" and a dropdown menu set to "Windows". Below it, a note says "Hol dir einen **Plex Pass**, um auf exklusive Vorschau-Downloads zuzugreifen.". On the right, there's a section for "Plex für Windows" with a Windows logo, the text "Windows 10 or later (64-bit only). 1.6.3.1009-57cf57c8 January 30, 2020", and a large orange "64 Bit herunterladen" button. At the bottom, a small note reads "Mit dem Herunterladen dieser Software bestätigst du, dass du die [Allgemeinen Geschäftsbedingungen](#) akzeptierst und mindestens 13 Jahre alt bist."

Luis Ricardo Lüscher
M126 - Peripheriegeräte im Netzwerkbetrieb einsetzen

Nun habe ich einige Medien in den vorhin definierten Ordner gelegt.

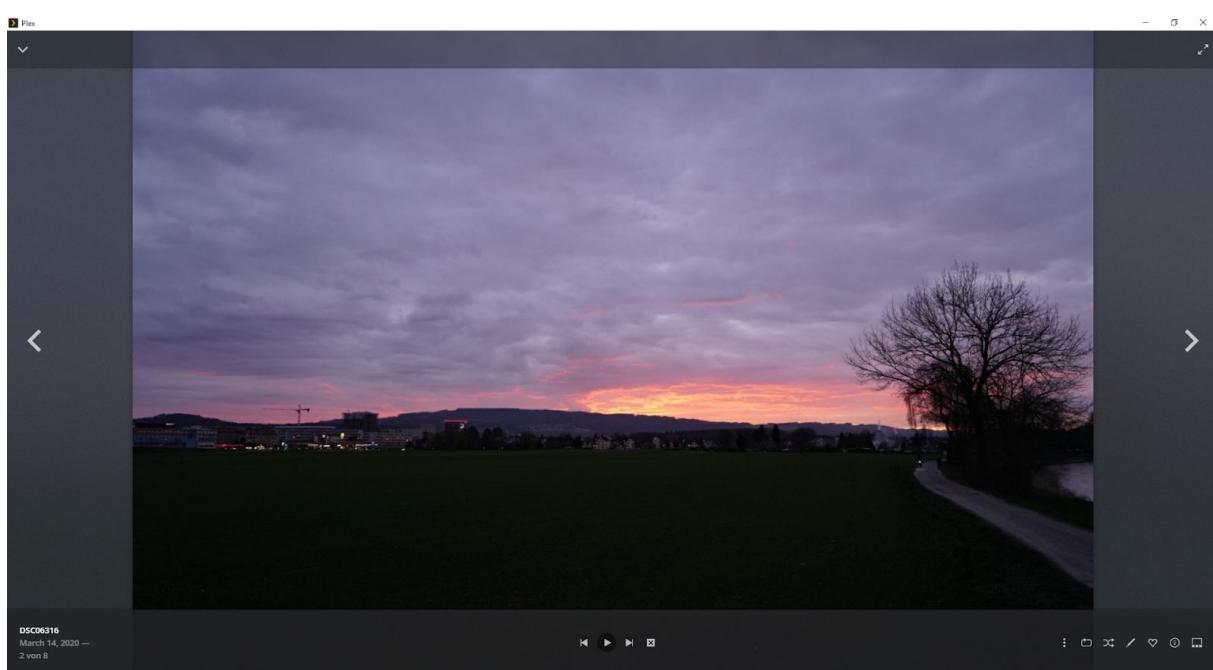


Nun kann man unter den angehefteten Medien den Punkt Fotos finden.



Danach findet man die vorhin in den Ordner kopierten Medien.

Luis Ricardo Lüscher
M126 - Peripheriegeräte im Netzwerkbetrieb einsetzen



Man kann auch die Bilder via Slideshow darstellen, dafür einfach alle Bilder auswählen und dann auf den Play Button klicken.

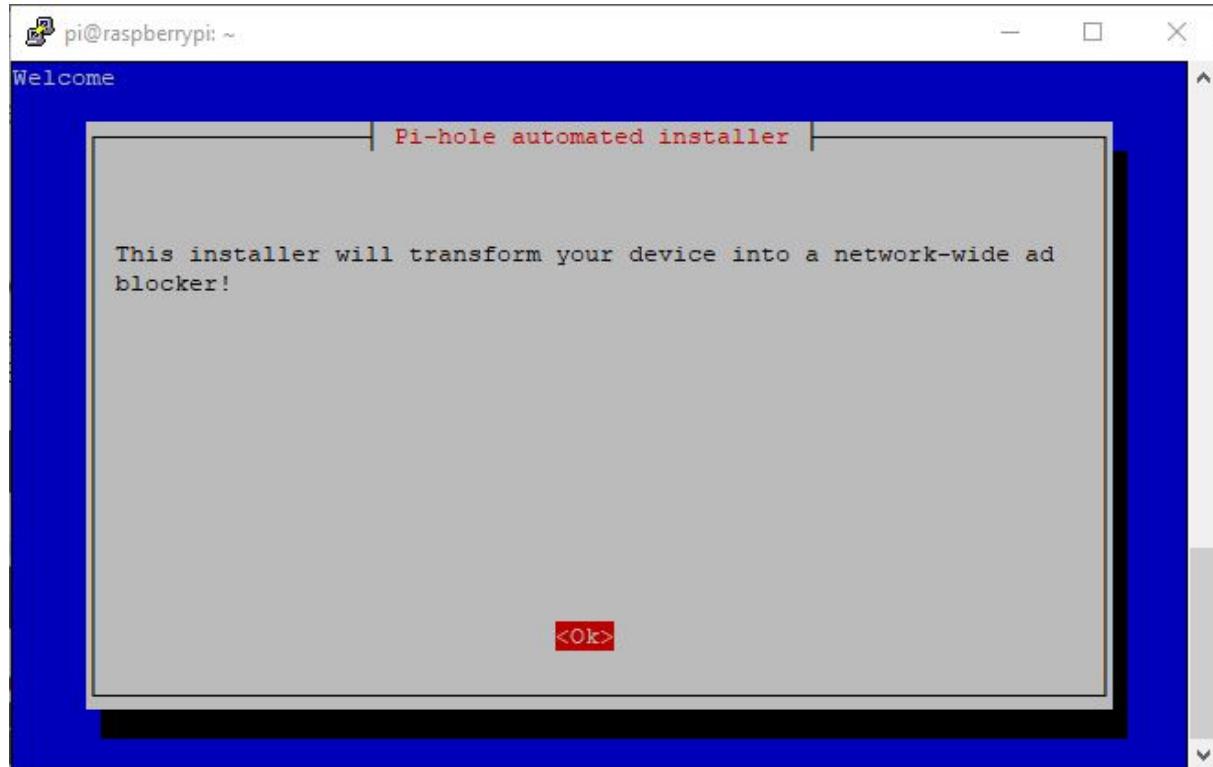
A screenshot of the Plex media center interface. On the left, there's a sidebar with sections for 'ANGEHEFTET' (Attached), 'RASPBERRYPI', 'Fotos' (selected), 'PLEX', 'Web Shows', 'Nachrichten', 'Podcasts', and 'TIDAL'. The main area is titled 'ZULETZT HINZUGEFÜGTE FOTOS' (Recently Added Photos) and shows eight thumbnail images of the sunset landscape, each with a yellow checkmark in the top-left corner. Above the thumbnails, it says '✓ 8 Elemente ausgewählt' (8 items selected) and 'Alles abwählen' (Deselect All). At the bottom of the main area, there's a small thumbnail of the fifth image with the file name 'DSC06313' and the date 'March 14, 2020 — 5 von 8'. The top right of the main area has a 'WERDE PREMIUM-MITGLIED' (Become Premium Member) button. The bottom of the screen features a dark bar with playback controls (back, forward, play/pause) and a large button in the center that says 'Gewählte abspielen' (Play Selected).

W97 Pi-Hole

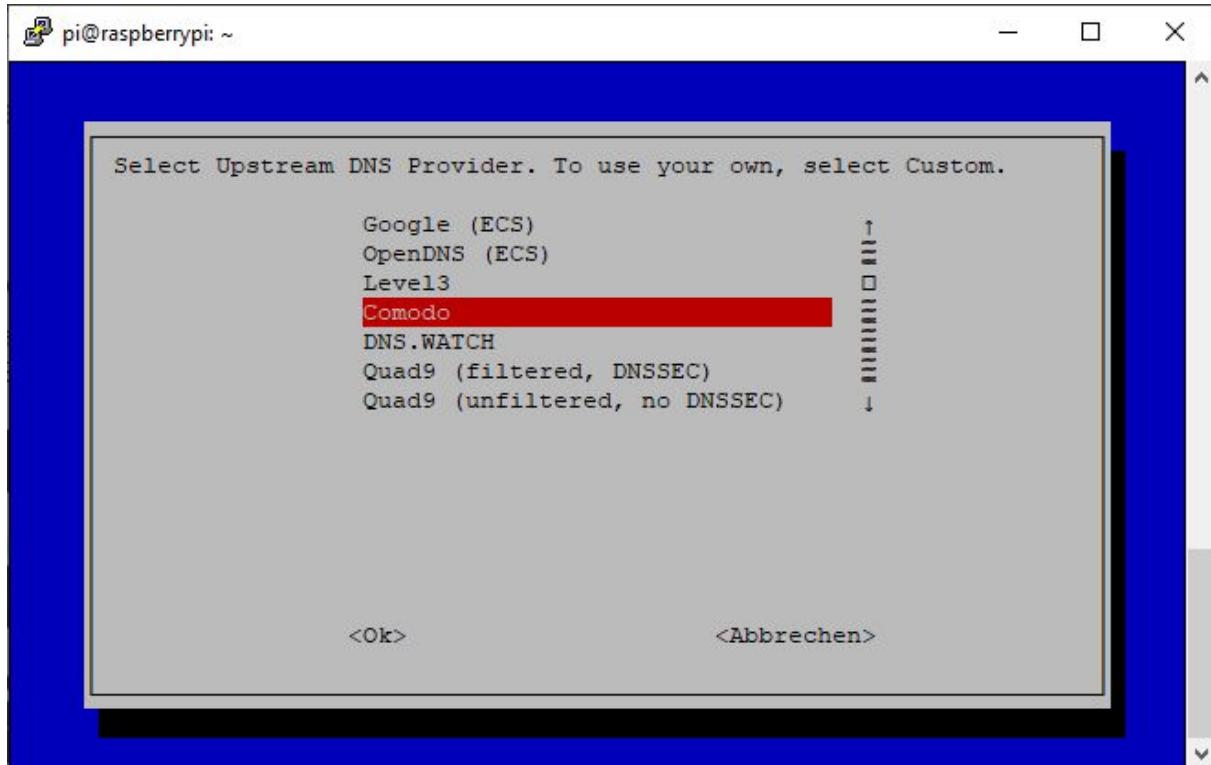
Die Installation des Pi Hole funktioniert ganz einfach. Dafür einfach folgenden Befehl ausführen.

```
pi@z54:~ $ sudo curl -sSL https://install.pi-hole.net/ | bash
```

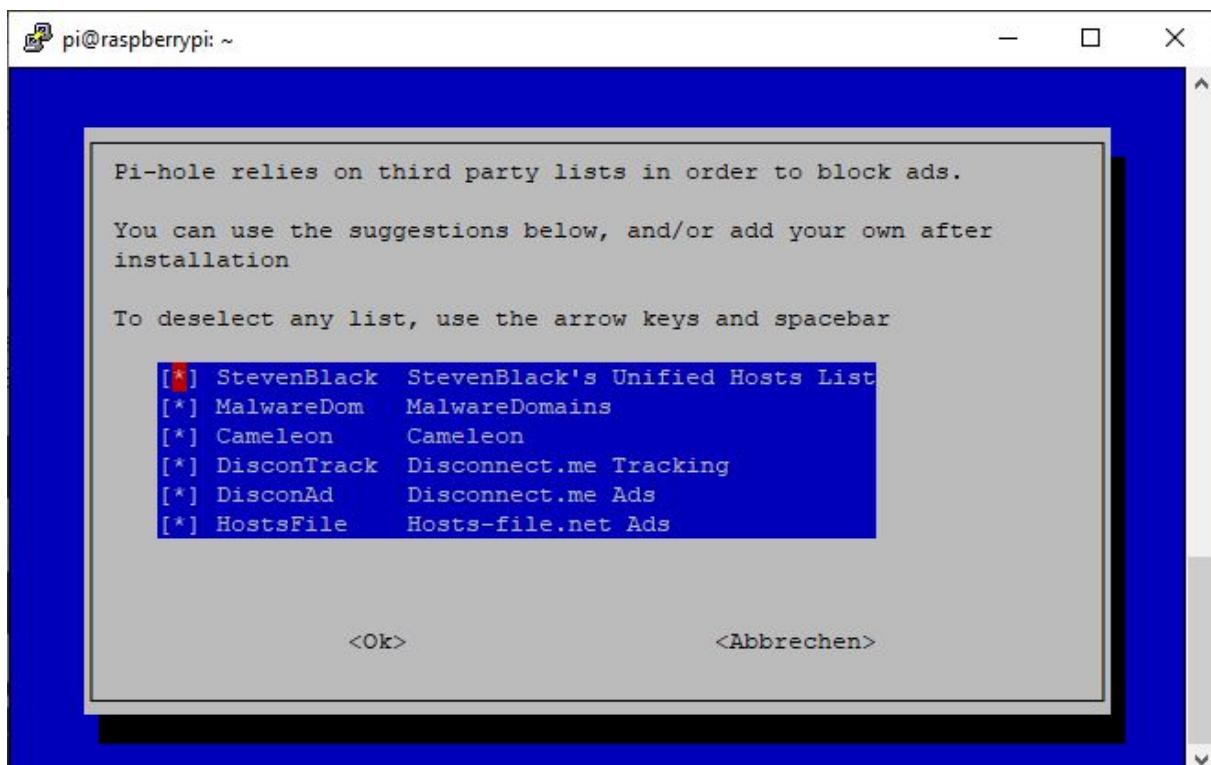
Wenn die entsprechende Datei heruntergeladen ist, startet das Setup automatisch. einfach mit "OK" bestätigen.



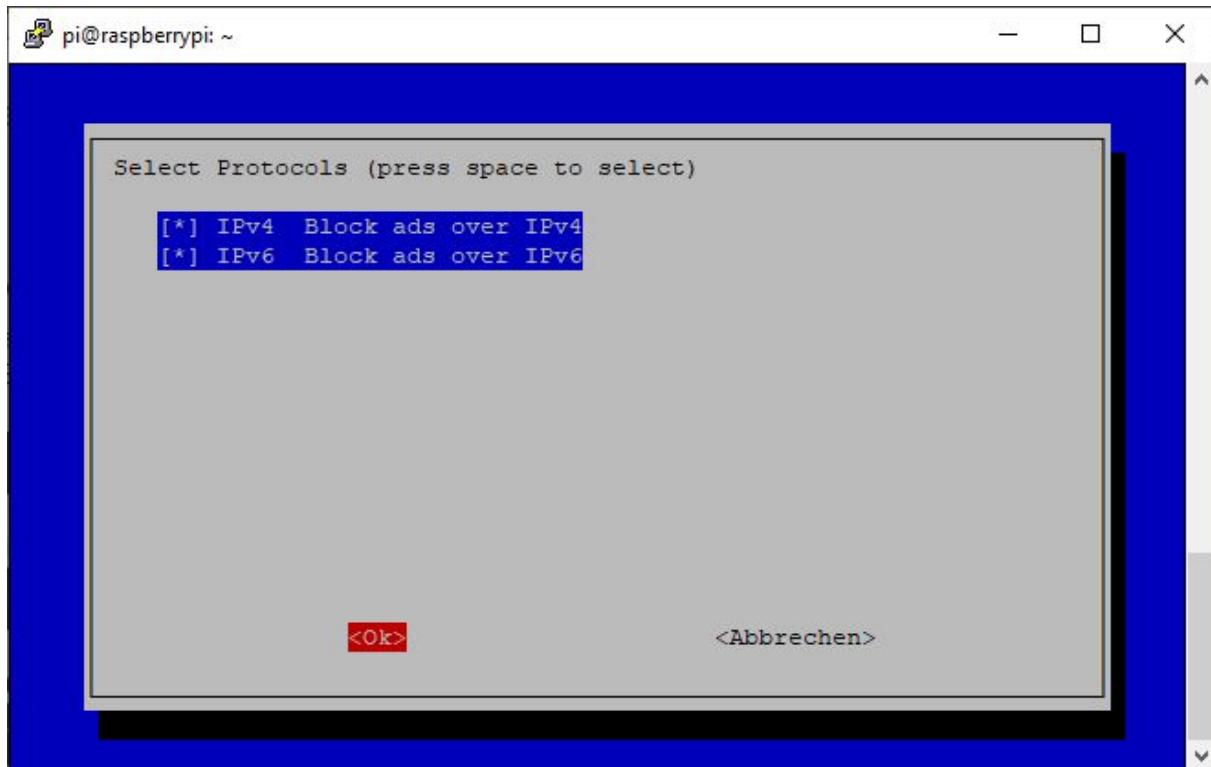
Nun muss man den entsprechenden DNS Provider. Ich habe mich für Comodo entschieden, da dieser alle Werbeanzeigen blockiert.



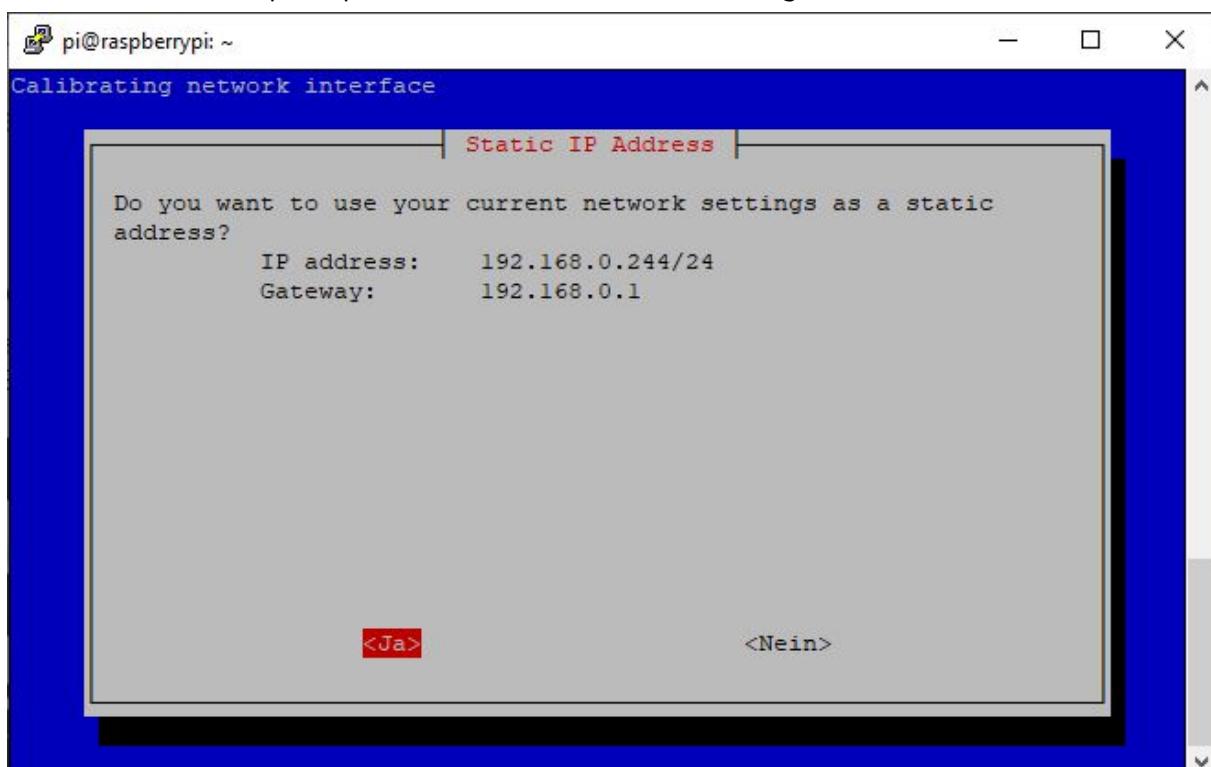
Da sich Pi-Hole auf Drittanbieter stützt, kann man die entsprechenden Anbieter aus- oder abwählen. Um den grössten Bereich abzudecken belasse ich dies beim Default.



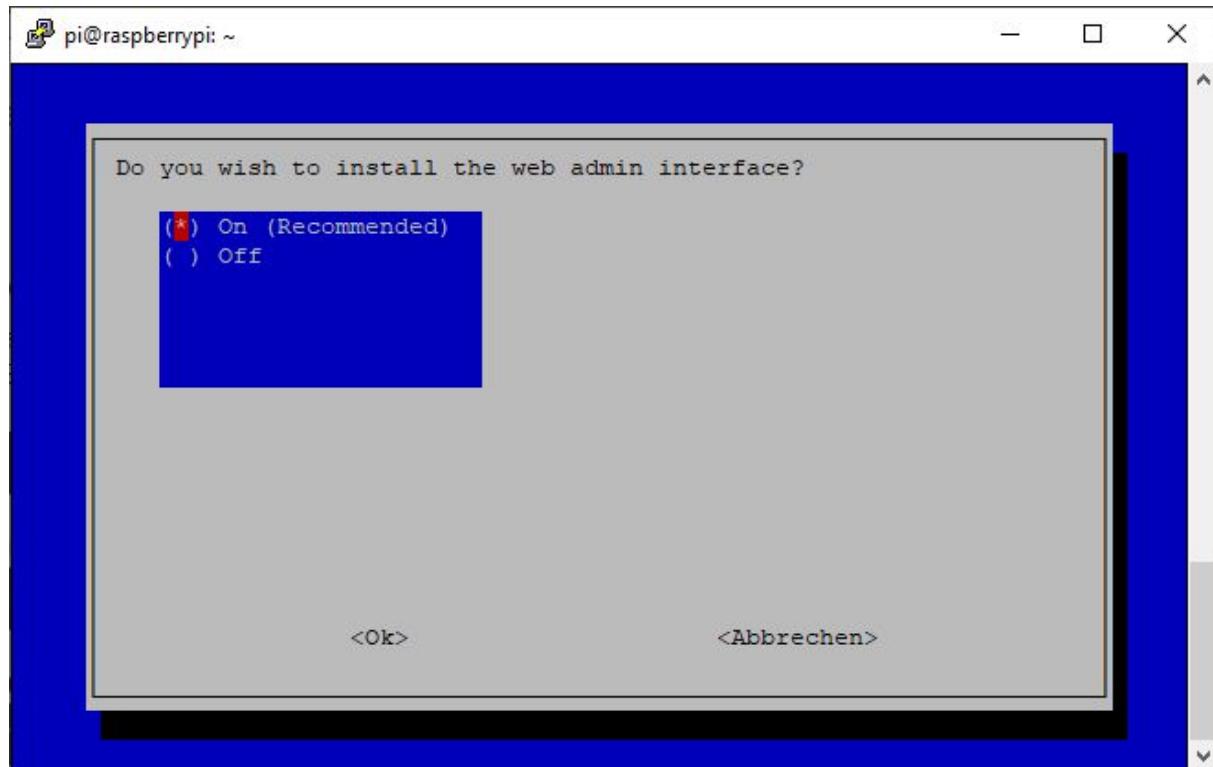
Dann muss man die entsprechenden Protokolle auswählen. Dies ist von netzwerk zu Netzwerk anderst.



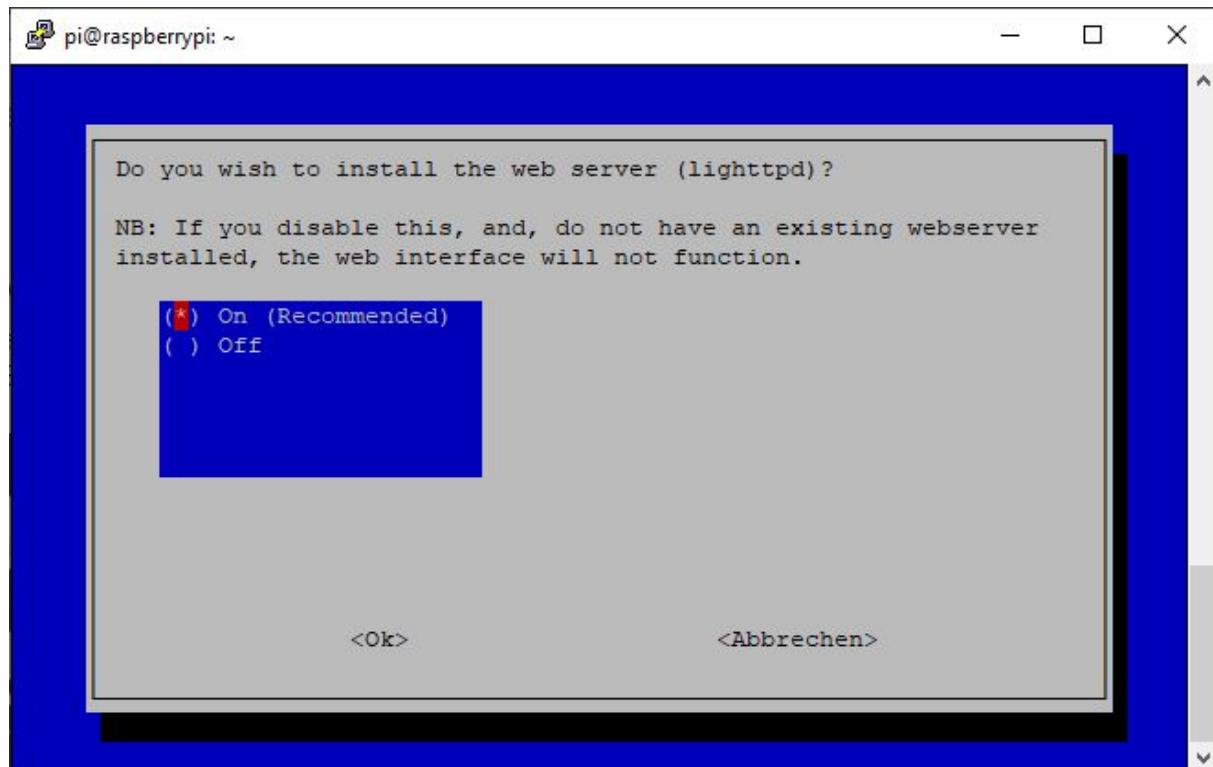
Dann muss man die entsprechenden netzwerkeinstellungen nun statisch setzen. Dies übernimmt das Setup Script für uns. einfach mit "Ja" bestätigen.



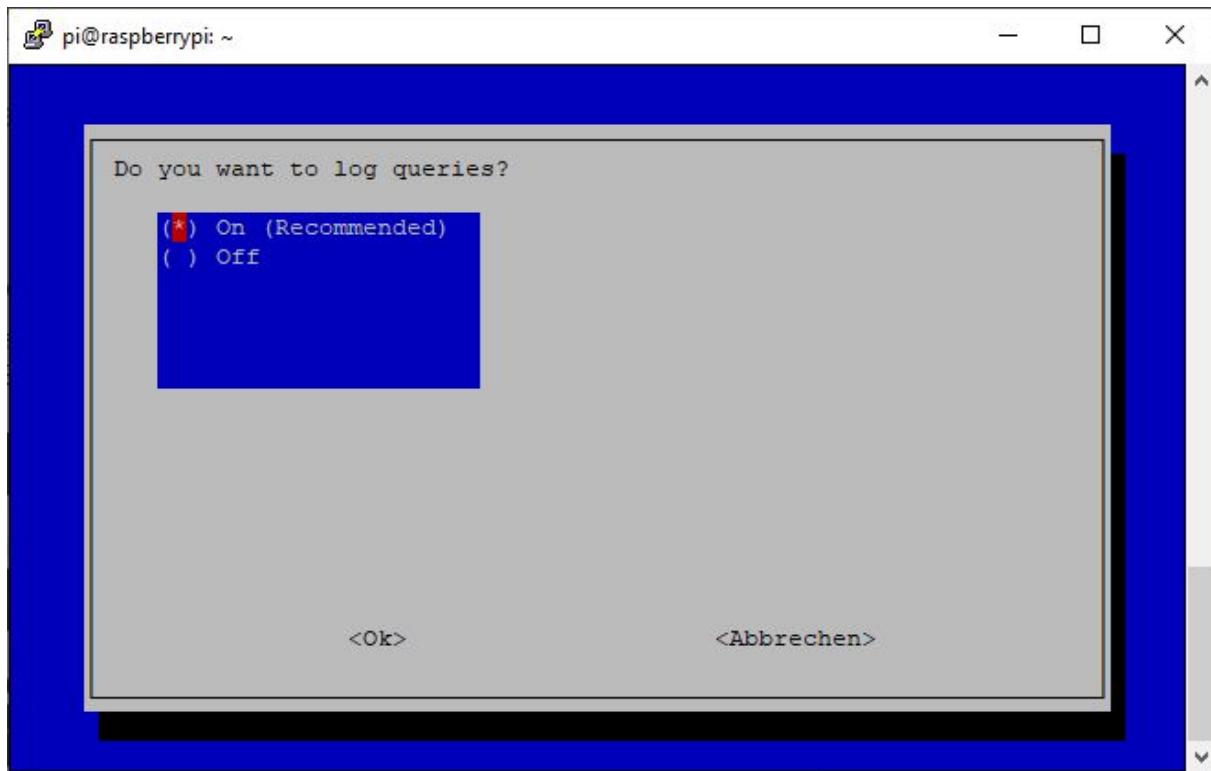
Nun kann man wenn gewünscht das Admin panel installieren. Da dieses noch tolle Funktionen anbietet werde ich dieses installieren.



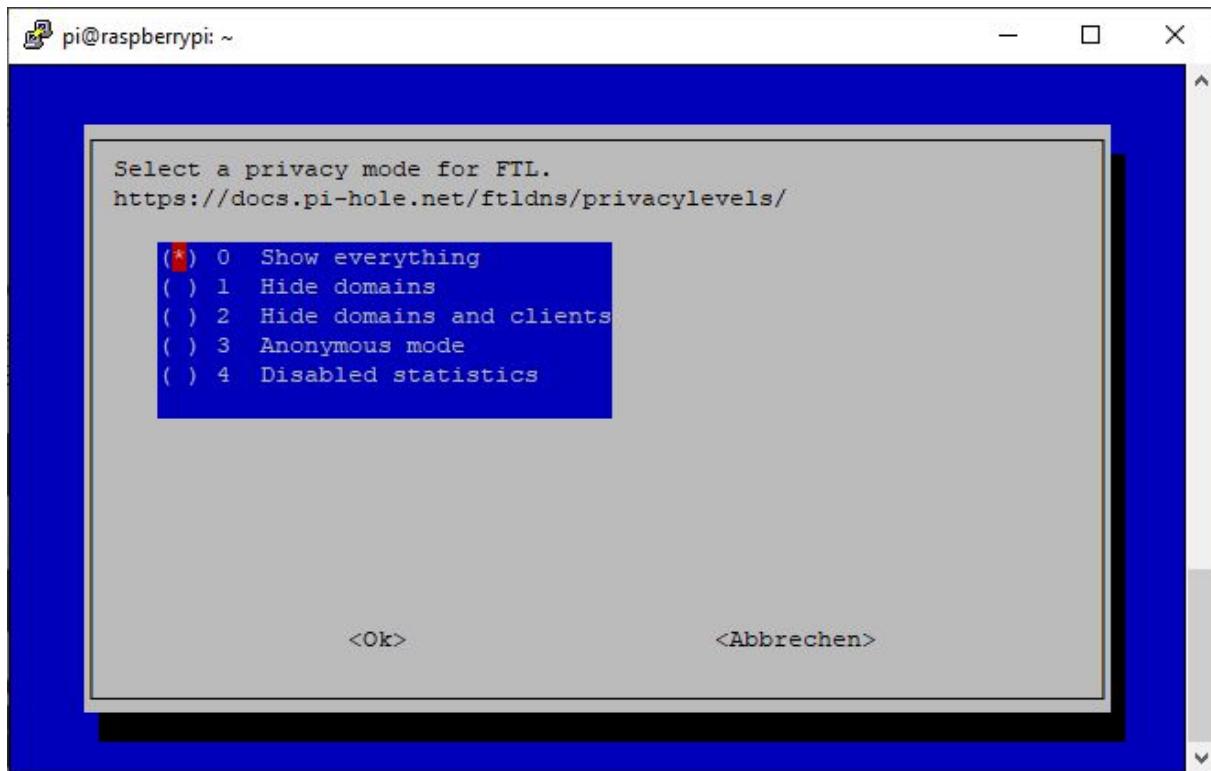
Nun muss man noch den Webserver installieren.



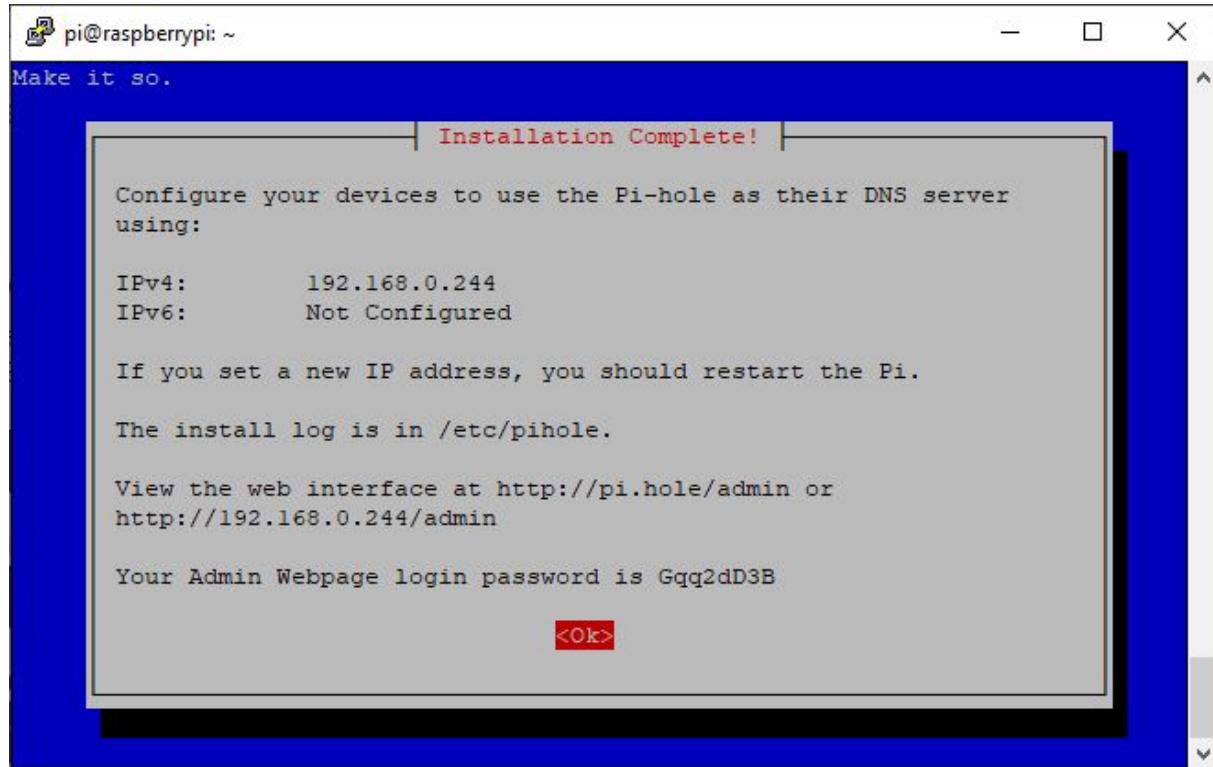
Zudem werden Anfragen gespeichert.



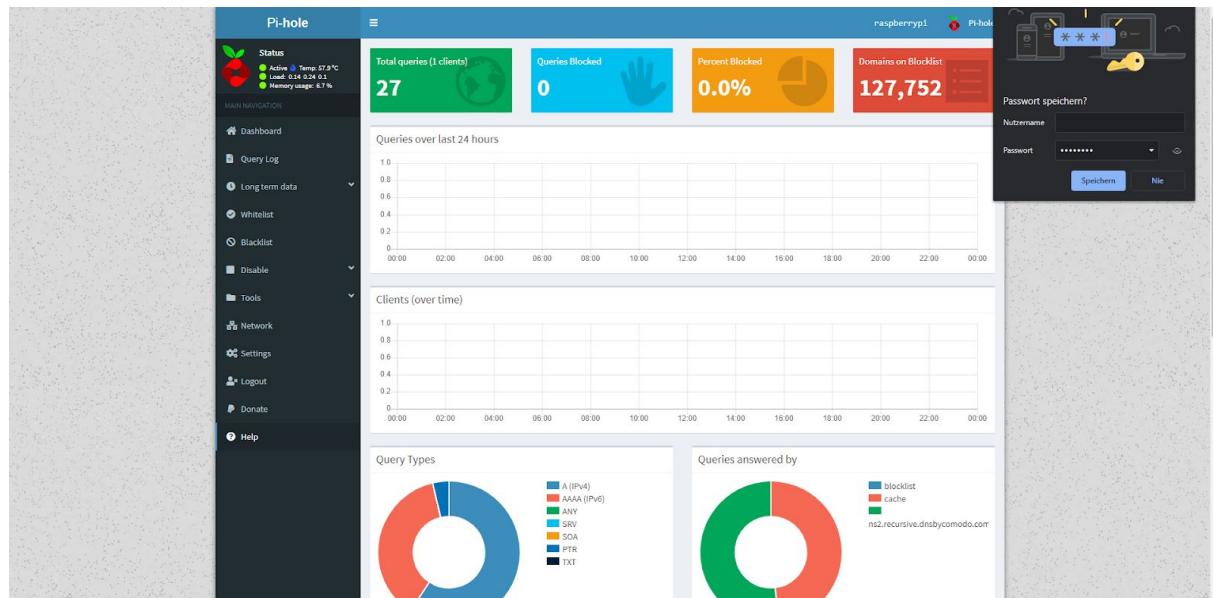
Nun werden wir noch die privatsphäre einstellungen noch bestimmen. Ich habe mich für 0 entschieden. Da ich diesen Pi-Hole nicht verwenden werde.



Nun sind wir am Ende des Setup angelangt. Hier einfach die entsprechenden parameter mit "OK" bestätigen. Wichtig ist, dass man das Passwort für das Admin Interface sich merkt.



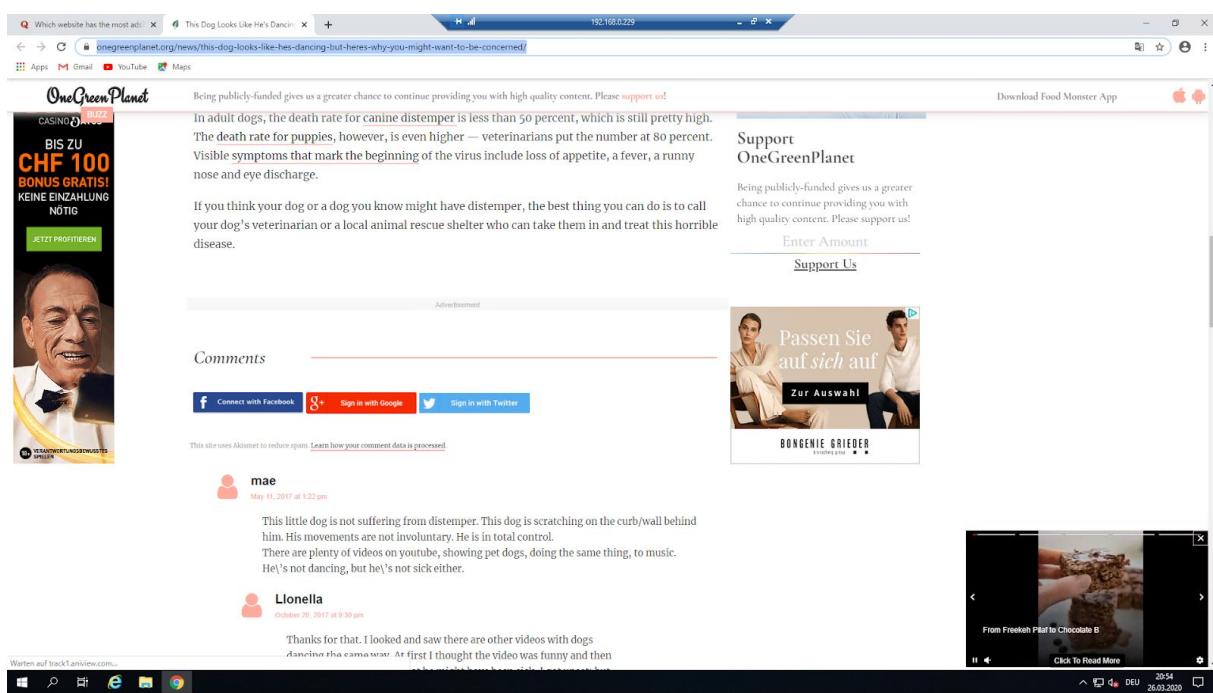
Nun kann man sich im Admin panel einloggen.



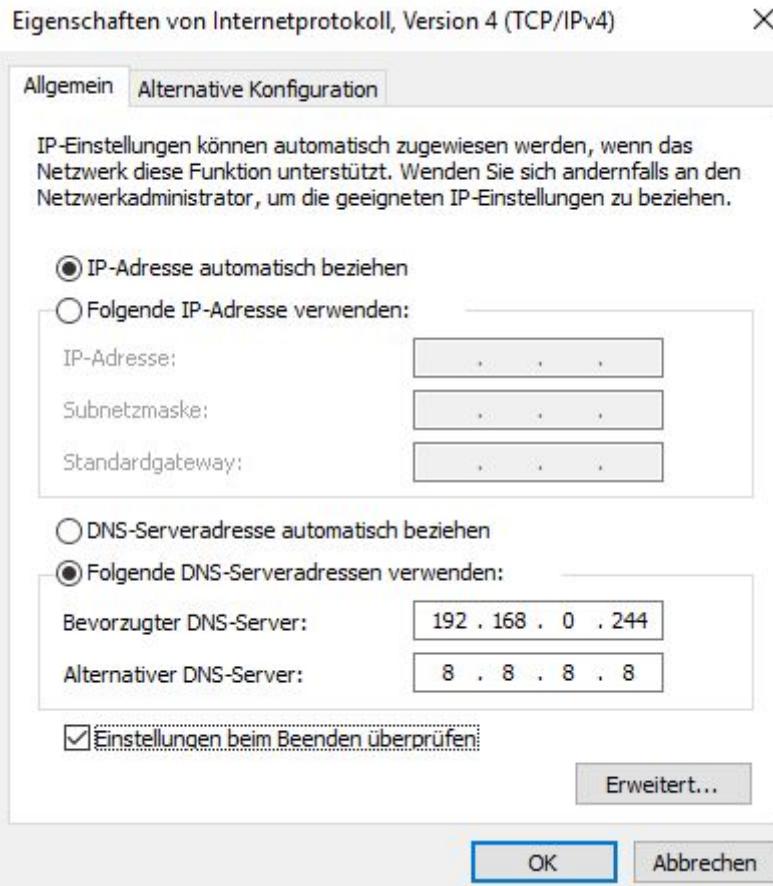
Nun werden wir die Funktionalität des Pi-hole zu testen. Dies ist eine Website deren Verkehr auf dem Gerät noch nicht über den Pi-hole läuft.

Luis Ricardo Lüscher

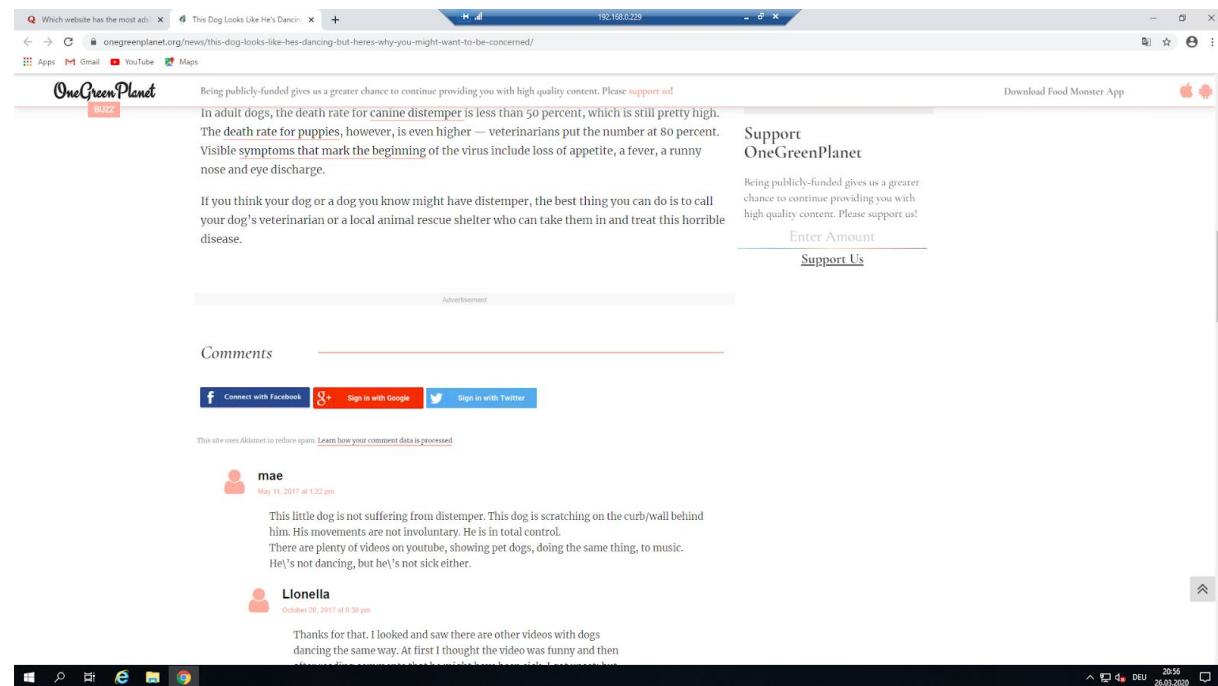
M126 - Peripheriegeräte im Netzwerkbetrieb einsetzen



Nun werden wir in den Adapter-Einstellungen des Gerätes das Pi-hole als bevorzugten DNS eingeben.



Wenn man die Seite nun aktualisiert, hat man keine Werbung mehr.



Im Admin Panel sieht man dann ebenfalls die blockierte Werbungen.



W98 Minecraft Server mit Cuberite

Nun werden wir mittels eines Github Repository einen im eigenen lokalen Minecraft Server erstellen. Man kann dann sogar einige Einstellungen auf einem zugehörigen Web Tool machen.

```
pi@z54:~ $ sudo apt-get update && sudo apt-get upgrade -y
```

Dann installieren wir einige Dinge für eine bessere Performance.

```
pi@z54:~ $ sudo apt-get install git cmake make gcc screen clang
```

Dann laden wir das Git Repository herunter auf welchem die Cuberite Datei sind.

```
pi@z54:~ $ git clone --recursive https://github.com/cuberite/cuberite.git
```

Danach wechselt man in den Ordner, initialisieren und updaten es.

```
pi@z54:~ $ cd cuberite
pi@z54:~ $ git submodule init
```

```
pi@z54:~ $ git submodule update
```

Jetzt können wir kompilieren. Dieser Schritt dauert allerdings fast zwei Stunden, daher ist Geduld gefragt. Achte darauf, dass du das Terminal nicht schließt, sonst muss erneut kompiliert werden.

```
pi@z54:~ $ cmake . -DCMAKE_BUILD_TYPE=RELEASE
```

Wenn es erfolgreich fertig kompiliert ist, verschieben wir den neu erstellten Ordner. Wir wechseln nun in den Ordner mit den kompilierten Daten.

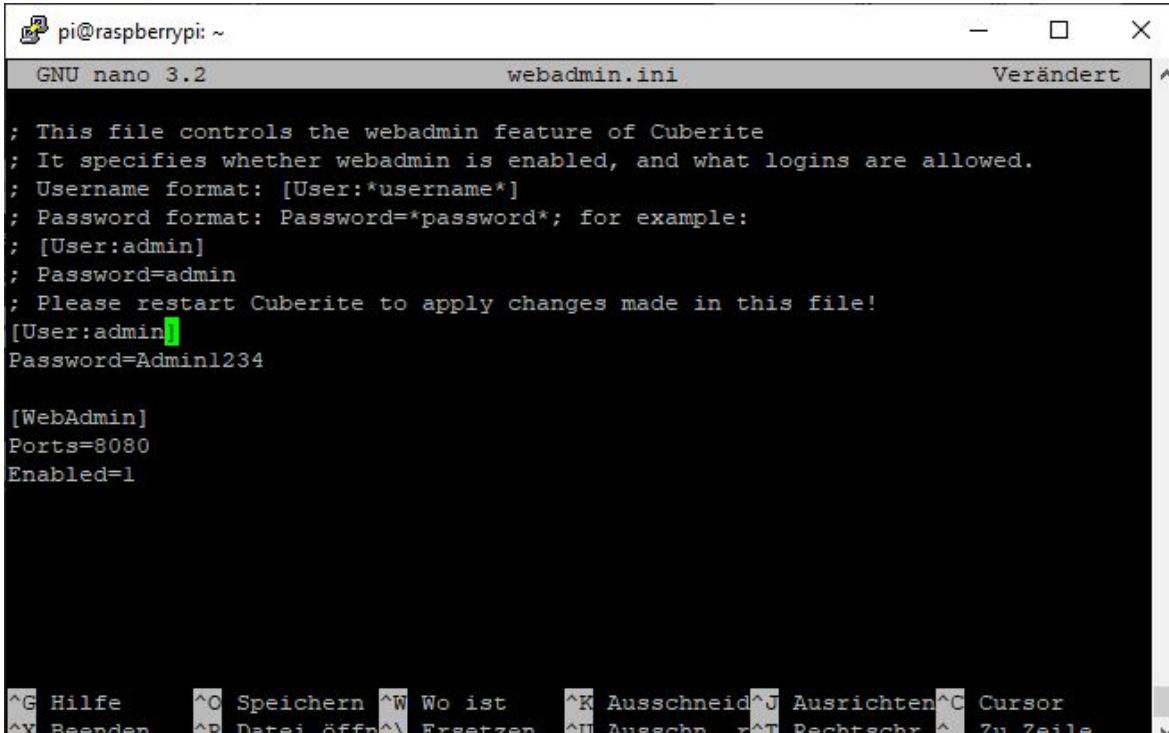
```
pi@z54:~ $ mv Server/ ~/Server/  
pi@z54:~ $ cd ~/Server/
```

Zu Beginn müssen wir für das Web Interface noch einen User anlegen.

```
pi@z54:~ $ sudo nano webadmin.ini
```

Nun fügen wir im File neue Anmelde Credentials ein.

```
[User:admin]  
Password=Admin1234
```



```
pi@raspberrypi: ~  
GNU nano 3.2          webadmin.ini          Verändert  
  
; This file controls the webadmin feature of Cuberite  
; It specifies whether webadmin is enabled, and what logins are allowed.  
; Username format: [User:*username*]  
; Password format: Password=*password*; for example:  
; [User:admin]  
; Password=admin  
; Please restart Cuberite to apply changes made in this file!  
[User:admin]  
Password=Admin1234  
  
[WebAdmin]  
Ports=8080  
Enabled=1  
  
^G Hilfe      ^O Speichern ^W Wo ist      ^K Ausschneid^J Ausrichten^C Cursor  
^X Beenden    ^R Datei öffn^E Ersetzen   ^U Ausschn. r^T Rechtschr.^_ Zu Zeile
```

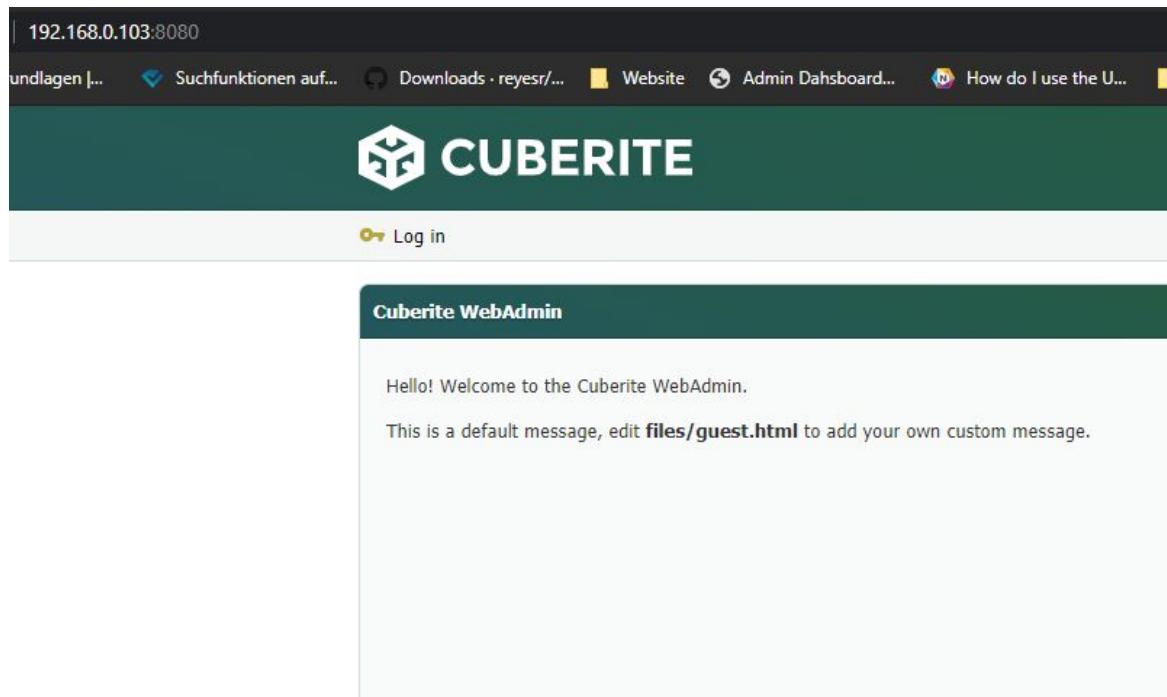
Danach kann man den Server mal starten.

```
pi@z54:~ $ ./Cuberite
```

Nun kann man im Browser folgendes eingeben.

```
http://IP_RASPI:8080
```

Sobald man diese Adresse öffnet kommt folgendes Fenster. Hier dann einfach auf Log In klicken.



Dann öffnet sich folgendes Fenster im Browser, hier dann einfach die vorhin definierten Anmeldedaten einfügen.

Anmelden

http://192.168.0.103:8080

Die Verbindung zu dieser Website ist nicht sicher

Nutzername

Passwort

Anmelden Abbrechen

Und so sieht dann am Ende das Web Interface dazu aus.

The screenshot shows the Cuberite web interface. At the top, there's a dark header bar with the Cuberite logo and the word "CUBERITE". Below it, a navigation bar has "Welcome back, admin" and "Log out" on the left, and "Players online: 0", "Memory: 143.38MB", and "Chunks: 199" on the right. The main area is divided into two sections: "Menu" on the left and "Current Game" on the right. The "Menu" section contains links for Home, Server Management (Core, Chat, Manage Server, Permissions, Player Ranks, Players, Plugins, Ranks, Server Settings, Time & Weather, Whitelist), and Help. The "Current Game" section shows "Plugins:" with "Core (version 16)" and "ChatLog (version 3)" listed, and an empty "Players:" list.

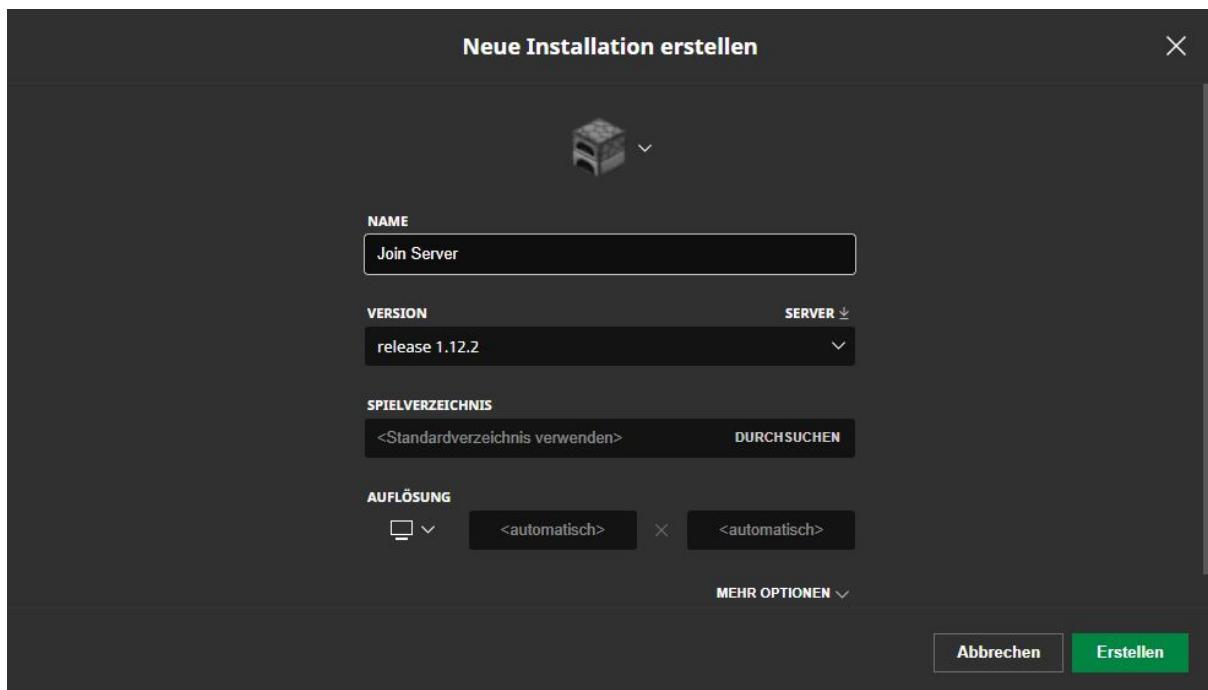
Etwas interessantes was ich noch herausgefunden habe. Es gibt einen expliziten MC-Server Port, der sogar offiziell ist laut Wikipedia.

The screenshot shows the "Core - Server Settings" page. At the top, there's a "Server Settings" section with tabs for General, Worlds, and World. Under "General Settings", there's a "Server" section with fields for Description (Cuberite - in C++!), Max Players (100), IPv4 Ports (25565), IPv6 Ports (empty), Default ViewDistance (10), and Hardcore (Off). Below this is an "Authentication" section. Above the "Server Settings" section, there's a table showing port mappings: 24842, 25565, and 25565. The first row (24842) is for StepMania. The second row (TCP) is for Minecraft, marked as "Ausgewiesener Server". The third row (MySQL) is for MySQL Standard MySQL Port. A legend indicates that yellow cells are "offiziell" (official) and red cells are "noffiziell" (unofficial).

Nun können wir mal Minecraft starten. Danach auf Direkt verbinden klicken und die IP-Adresse des RasPi eingeben. Es kann sein das folgende Fehlermeldung kommt.



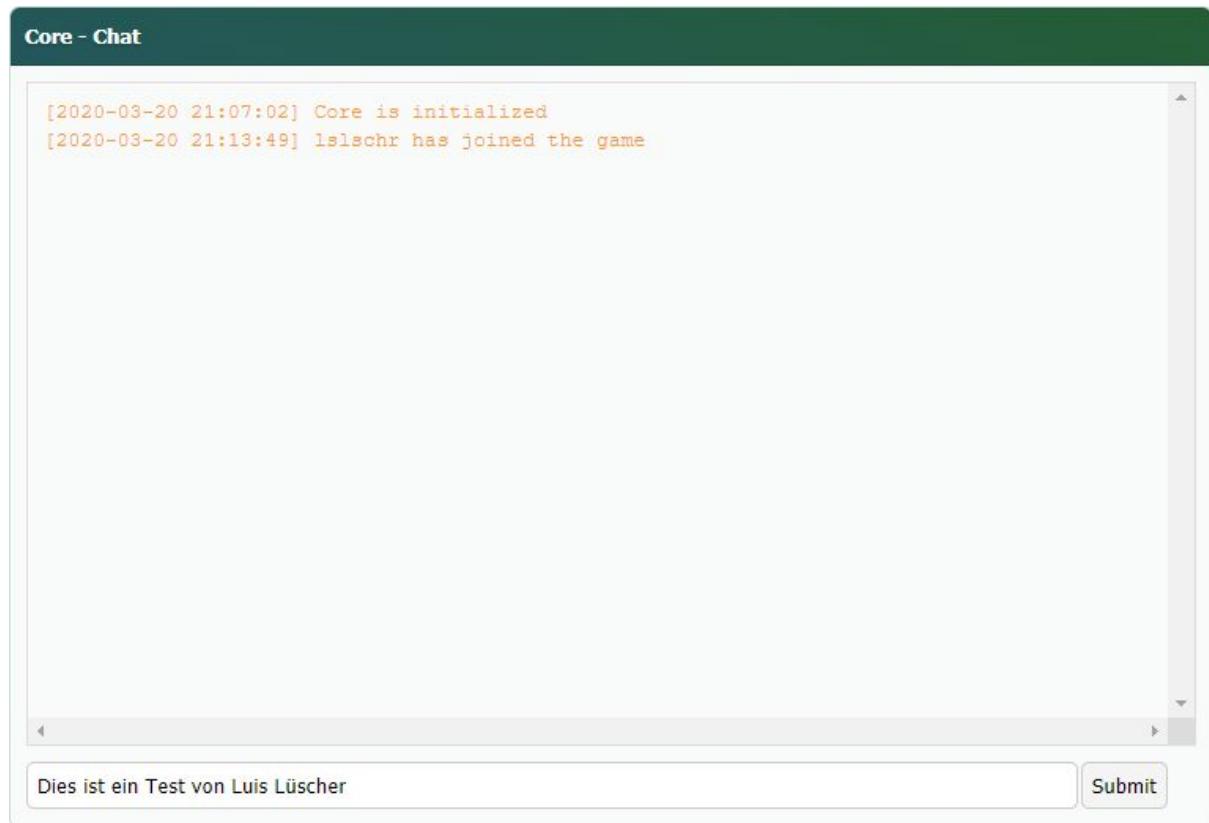
Nun muss man zurück ins Profil, ein neues Profil erstellen und dann eine der obenstehenden Version auswählen.



Danach sollte man ohne Probleme dem Server beitreten können.



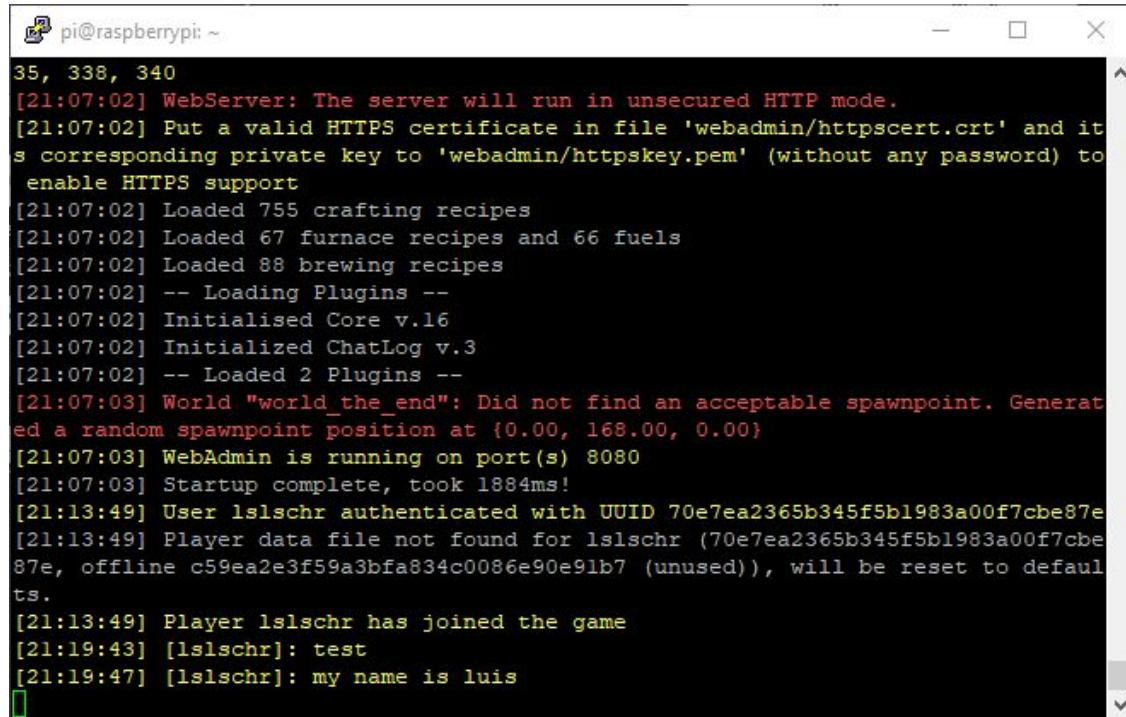
Im Chat auf dem Web Interface sieht man ebenfalls, dass der Spieler lslschr (Mein Gamertag) beigetreten ist. Nun versenden wir als Admin eine Nachricht in den Chat des Servers.



Ingame sehe ich diesen dann.



Auch im Terminal ist der Chatverlauf einsehbar. Der Server kann mittels CTRL + C beendet werden.



pi@raspberrypi: ~

```
35, 338, 340
[21:07:02] WebServer: The server will run in unsecured HTTP mode.
[21:07:02] Put a valid HTTPS certificate in file 'webadmin/httpscert.crt' and its corresponding private key to 'webadmin/httpskey.pem' (without any password) to enable HTTPS support
[21:07:02] Loaded 755 crafting recipes
[21:07:02] Loaded 67 furnace recipes and 66 fuels
[21:07:02] Loaded 88 brewing recipes
[21:07:02] -- Loading Plugins --
[21:07:02] Initialised Core v.16
[21:07:02] Initialized ChatLog v.3
[21:07:02] -- Loaded 2 Plugins --
[21:07:03] World "world_the_end": Did not find an acceptable spawnpoint. Generated a random spawnpoint position at {0.00, 168.00, 0.00}
[21:07:03] WebAdmin is running on port(s) 8080
[21:07:03] Startup complete, took 1884ms!
[21:13:49] User lschr authenticated with UUID 70e7ea2365b345f5b1983a00f7cbe87e
[21:13:49] Player data file not found for lschr (70e7ea2365b345f5b1983a00f7cbe87e, offline c59ea2e3f59a3bfa834c0086e90e91b7 (unused)), will be reset to defaults.
[21:13:49] Player lschr has joined the game
[21:19:43] [lschr]: test
[21:19:47] [lschr]: my name is luis
```

W100 Surveillance Station

Mit einem Raspberry Pi, einem Monitor, TV oder sonstigem Display lässt sich ohne grossen Aufwand ein Überwachungsmonitor der Kameras von einem Ubiquiti UniFi Video NVR System erstellen. (Sollte aber auch mit anderen Systemen funktionieren die RTSP streamen können.)

Nun laden wir die aktuelle Version herunter.

```
pi@z54:~ $ wget
https://github.com/Anonymousdog/displaycameras/archive/master.zip
```

Danach kann man das File extrahieren.

```
pi@z54:~ $ unzip master.zip
```

Nun wechseln wir ins das entpackte Verzeichnis.

```
pi@z54:~ $ cd displaycameras-master/
```

Nun machen wir das install.sh Script ausführbar.

```
pi@z54:~ $ chmod +x install.sh
```

Und danach kann man das Install.sh ausführen.

```
pi@z54:~ $ sudo ./install.sh
```

Nun kann man die verschiedenen Layouts einsehen.

```
pi@z54:~ $ ls -la example_layouts/
```

Die Konfigurationsdateien liegen unter /etc/displaycameras/

```
pi@z54:~ $ cd /etc/displaycameras/
```

Nun braucht man den sogenannten RTSP Link der jeweiligen Kamera. Dafür bin ich auf die jeweiligen Konfigurationsseiten auf den Kamerassen gegangen.

```
192.168.0.188/camera/config
```

Unter dem Punkt RTSP URL finde ich dann die jeweiligen Links meiner beiden Kamera's.

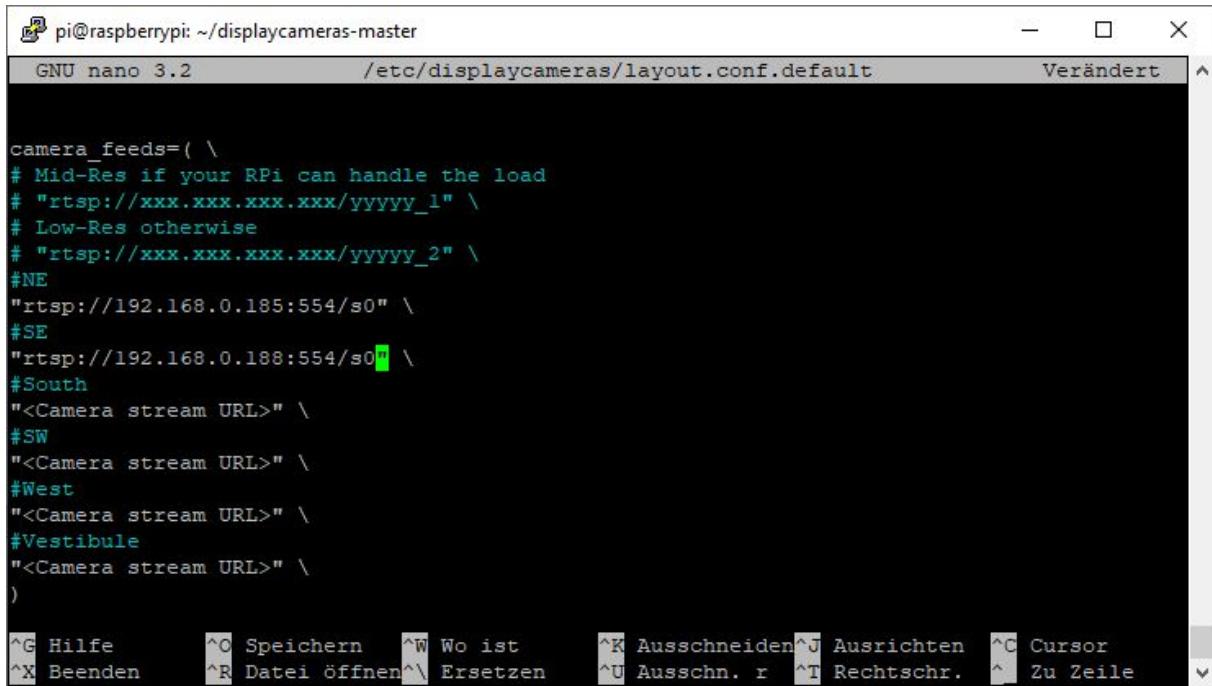
The screenshot shows the configuration page for a UniFi Video camera. At the top, there is a blue header bar with the UniFi logo and a back arrow icon. Below the header, the camera model is shown as "UVC G3 Dome". The "Mode" dropdown is set to "Standalone". Under "Display Options", the checkboxes for "Display Camera Name", "Display Timestamp", and "Display Logo" are checked, while "Enable Anonymous Snapshot" is unchecked. The "Microphone Volume" slider is at its maximum. To the right of the slider, the camera's status is listed: "Mode: Standalone", "Uptime: 28d 2h 16m". In the "VIDEO QUALITY" section, the "RTSP Stream" dropdown is set to "High (1920x1080)". To the right, the RTSP URL is "rtsp://192.168.0.188:554/s0" and the video parameters are "Width: 1920", "Height: 1080", "Bitrate: 3000 kbps", and "FPS: 15". There is also a checkbox for "Enable RTSP Authentication". At the bottom of the page are three buttons: "SAVE CHANGES" (in a blue bar), "Revert", and "SHOW VIDEO" (with a camera icon).

Luis Ricardo Lüscher
M126 - Peripheriegeräte im Netzwerkbetrieb einsetzen

The screenshot shows the 'Unifi Video' interface for a 'Standalone' camera. The camera model is UVC G3, operating in Standalone mode with an uptime of 28d 2h 18m. Configuration options include enabling external accessories, displaying camera name (checked), timestamp (checked), logo (checked), and anonymous snapshots (unchecked). A microphone volume slider is set to a medium level, and a red 'DISABLE MICROPHONE' button is visible. Under 'VIDEO QUALITY', the RTSP Stream is set to 'High (1920x1080)'. The RTSP URL is rtsp://192.168.0.185:554/s0, with width 1920, height 1080, bitrate 3000 kbps, and FPS at 15. An unchecked checkbox for 'Enable RTSP Authentication' is present. At the bottom are 'SAVE CHANGES', 'Revert', and 'SHOW VIDEO' buttons.

Danach öffnet man das Konfigurationsfile und fügt die Links unter dem Punkt "Camera Feeds" ein.

```
pi@z54:~ $ sudo nano /etc/displaycameras/layout.conf.default
```



```
pi@raspberrypi: ~/displaycameras-master
GNU nano 3.2          /etc/displaycameras/layout.conf.default      Verändert

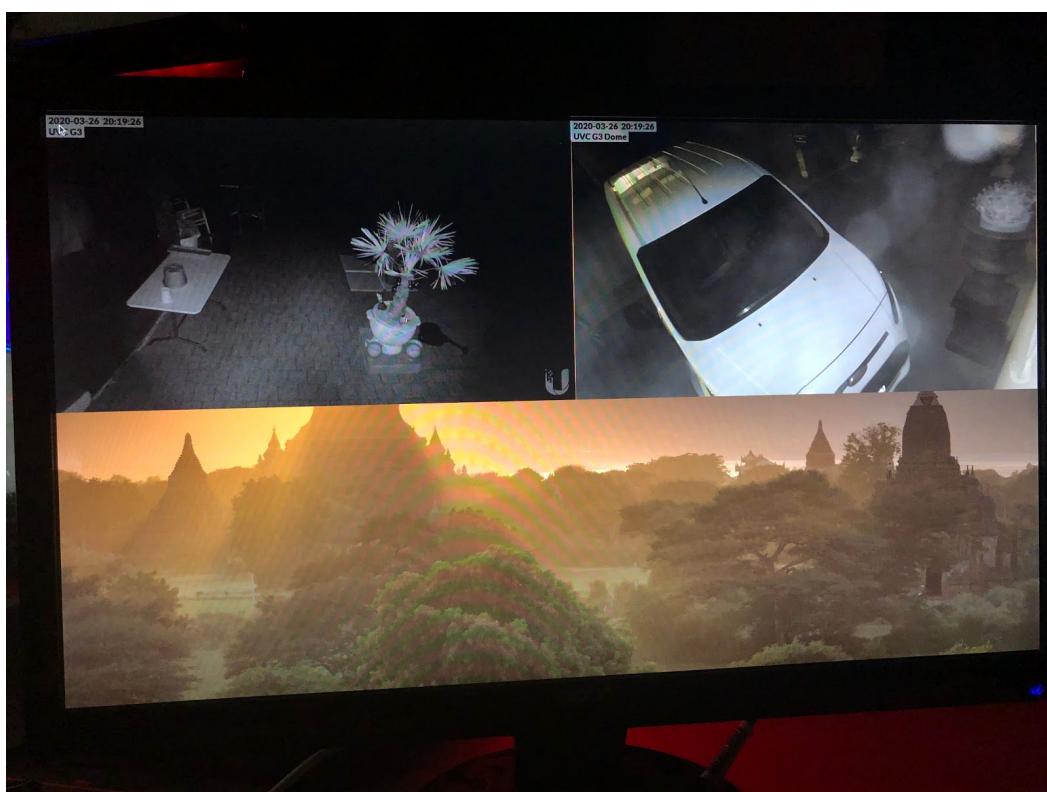
camera_feeds=( \
# Mid-Res if your RPi can handle the load
# "rtsp://xxx.xxx.xxx.xxx/yyyyy_1" \
# Low-Res otherwise
# "rtsp://xxx.xxx.xxx.xxx/yyyyy_2" \
#NE
"rtsp://192.168.0.185:554/s0" \
#SE
"rtsp://192.168.0.188:554/s0" \
#South
"<Camera stream URL>" \
#SW
"<Camera stream URL>" \
#West
"<Camera stream URL>" \
#Vestibule
"<Camera stream URL>" \
)

^G Hilfe      ^C Speichern    ^W Wo ist      ^K Ausschneiden^J Ausrichten   ^C Cursor
^X Beenden    ^R Datei öffnen^V Ersetzen    ^U Ausschn. r  ^T Rechtschr.  ^Z Zu Zeile
```

Nachdem man den Link hinzugefügt hat, kann man den Service neu starten.

```
pi@z54:~ $ sudo systemctl restart displaycameras.service
```

Ist der RasPi erfolgreich gestartet, kann man nun auf einem Bildschirm die entsprechenden Kameras sehen.



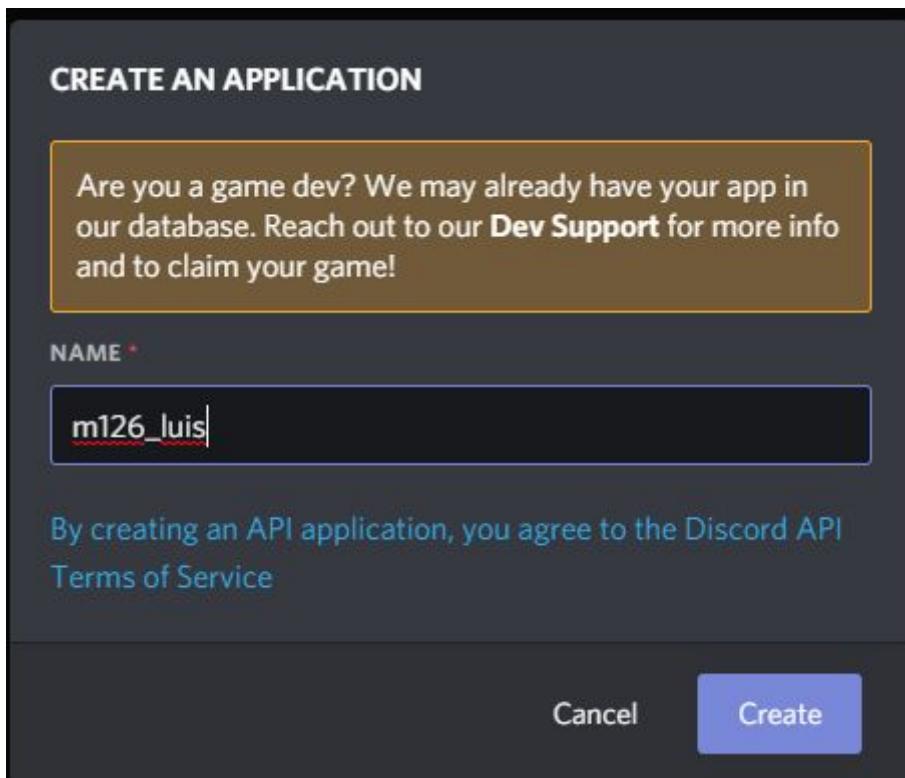
W101 Discord Python Bot

Zuerst braucht man eine Discord API. Dafür meldet man sich auf dieser [Website](#) mit seinem Discord Account an.

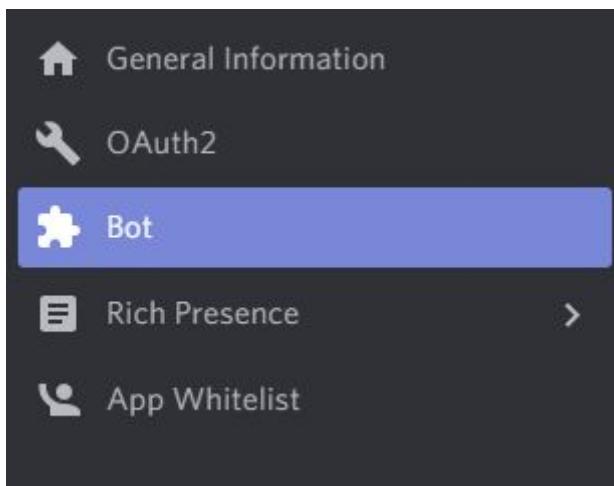
Danach klickt man auf "New Application".

New Application

Und dann gibt man seiner Applikation einen namen in diesem Fall "m126_luis".



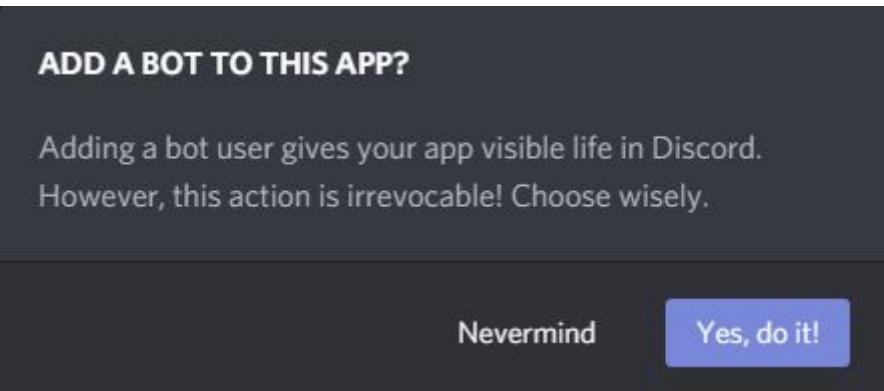
Dann geht man im Menü auf den Punkt "Bot".



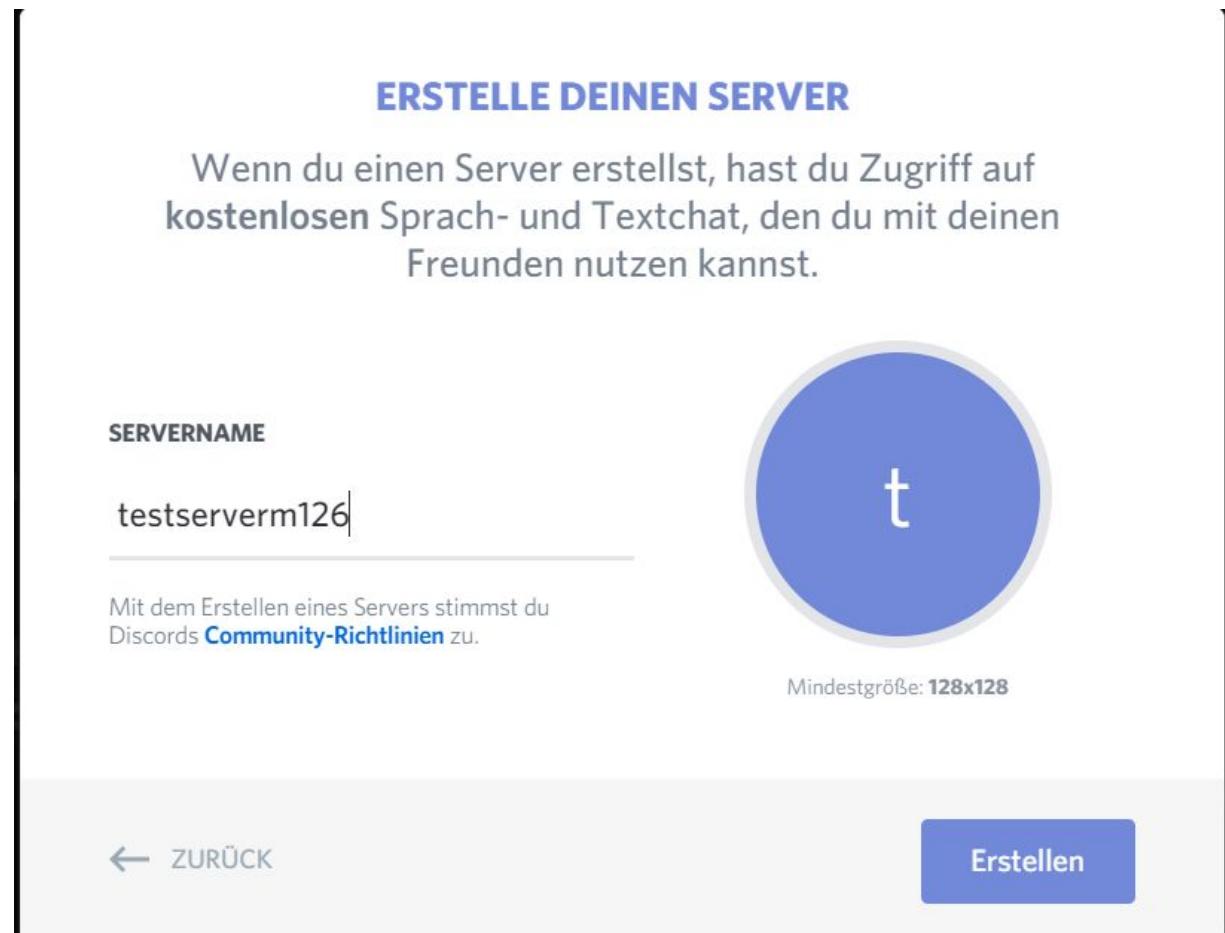
Und dann auf "Add Bot".

Add Bot

Es kommt dann ein Warnhinweis, diesen einfach mit "Yes, do it!" bestätigen.



Im Anschluss habe ich dann einen Server erstellt.



Und dann den Bot auf meinen Server eingeladen. Folgender Link:
https://discordapp.com/oauth2/authorize?client_id=your_client_id_goes_here&scope=bot&permissions=0



Danach installieren wir einige Pakete.

```
pi@z54:~ $ sudo apt install python3-pip
pi@z54:~ $ sudo apt install python3-cffi
pi@z54:~ $ sudo pip3 install discord.py[voice]
```

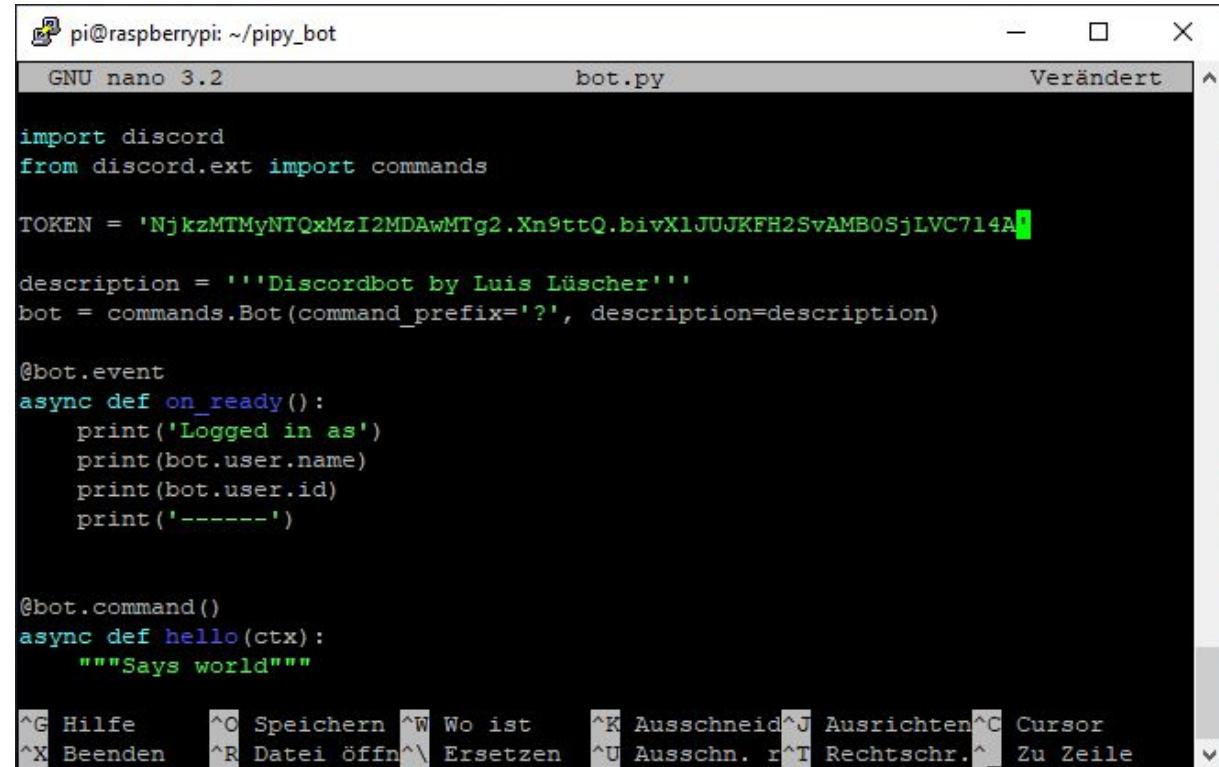
Zudem installieren wir die Python Discord Library

```
pi@z54:~ $ sudo python3 -m pip install -U discord.py[voice]
```

Danach erstellen wir ein Verzeichnis und ein File.

```
pi@z54:~ $ mkdir ~/pipy_bot
pi@z54:~ $ cd ~/pipy_bot
pi@z54:~ $ touch bot.py
pi@z54:~ $ nano bot.py
```

Dann habe ich einige Standard Codezeilen erstellt, die ein Discord Bot haben sollte.



The screenshot shows a terminal window titled "pi@raspberrypi: ~/pipy_bot". The file being edited is "bot.py". The code is a basic Discord bot setup using the discord.ext.commands library. It includes a token variable, a bot description, and an event listener for the "on_ready" event that prints the bot's name and ID. It also defines a command named "hello" that responds with "Says world". The terminal window has a status bar at the bottom with various keyboard shortcuts.

```
pi@raspberrypi:~/pipy_bot
GNU nano 3.2          bot.py          Verändert

import discord
from discord.ext import commands

TOKEN = 'NjKzMTMyNTQzMzI2MDAwMTg2.Xn9ttQ.bivX1JUJKFH2SvAMB0SjLVC714A'

description = '''Discordbot by Luis Lüscher'''
bot = commands.Bot(command_prefix='?', description=description)

@bot.event
async def on_ready():
    print('Logged in as')
    print(bot.user.name)
    print(bot.user.id)
    print('-----')

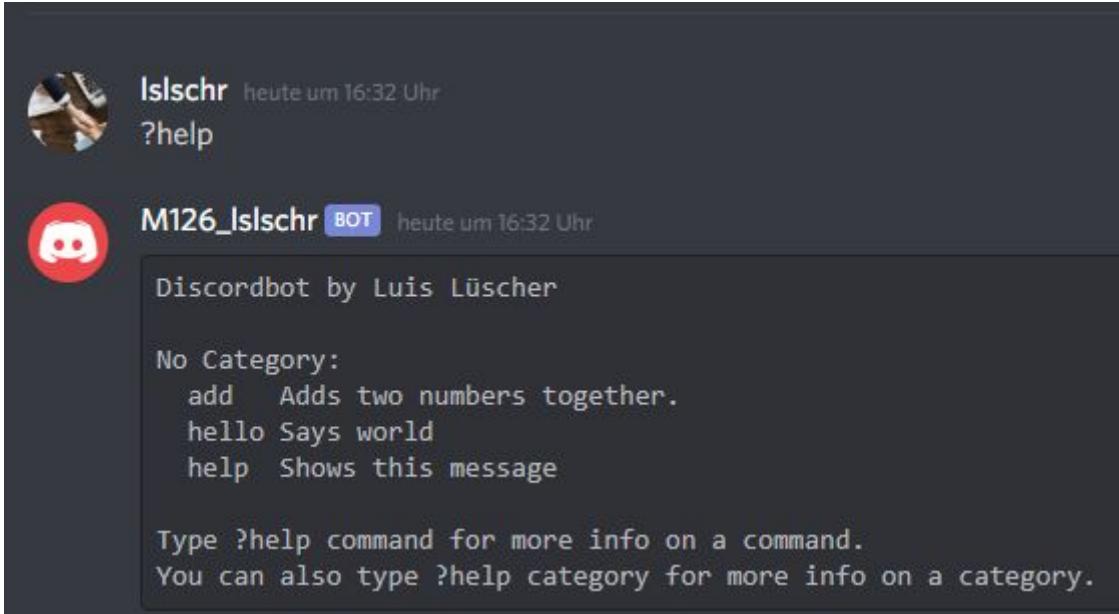
@bot.command()
async def hello(ctx):
    """Says world"""

^G Hilfe      ^O Speichern ^W Wo ist      ^K Ausschneid^J Ausrichten^C Cursor
^X Beenden    ^R Datei öffn^V Ersetzen   ^U Ausschn. r^T Rechtschr.^_ Zu Zeile
```

Und habe dann den Bot laufen lassen via:

```
pi@z54:~ $ python3 bot.py
```

Hier sind einige Beispiele meiner Default Commands dann auf dem Server.



lslschr heute um 16:32 Uhr
?help

M126_lschr BOT heute um 16:32 Uhr
Discordbot by Luis Lüscher

No Category:
add Adds two numbers together.
hello Says world
help Shows this message

Type ?help command for more info on a command.
You can also type ?help category for more info on a category.



lslschr heute um 16:32 Uhr
?add 187 420

M126_lschr BOT heute um 16:32 Uhr
607



lslschr heute um 16:33 Uhr
?hello

M126_lschr BOT heute um 16:33 Uhr
world

Mein programm sieht folgendermassen aus:

```
import discord
from discord.ext import commands
import asyncio

TOKEN = 'NjkzMMyNTQzMzI2MDAwMTg2.Xn9ttQ.bivXlJUJKFH2SvAMB0SjLVC714A'

description = '''Discordbot by Luis Lüscher'''
bot = commands.Bot(command_prefix='?', description=description)

@bot.event
async def on_ready():
```

```
print('Logged in as')
print(bot.user.name)
print(bot.user.id)
print('-----')

@bot.command()
async def hello(ctx):
    """Says world"""
    await ctx.send("world")

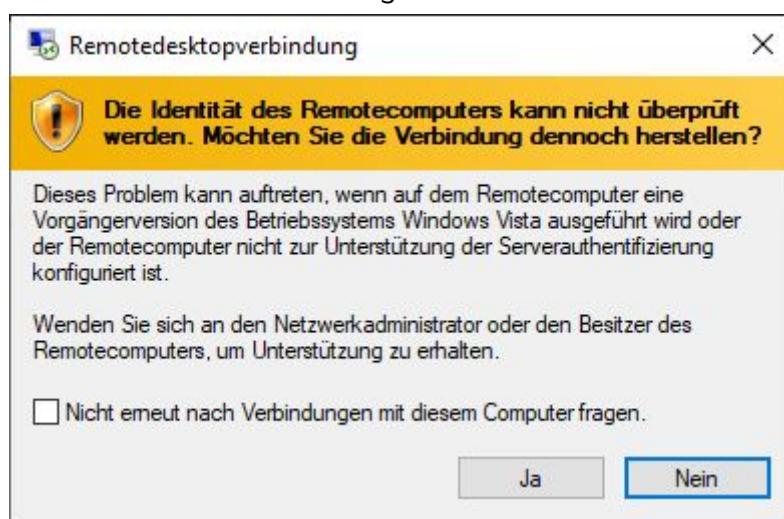
@bot.command()
async def add(ctx, left : int, right : int):
    """Adds two numbers together."""
    await ctx.send(left + right)
bot.run(TOKEN)
```

W103 RDP Raspi

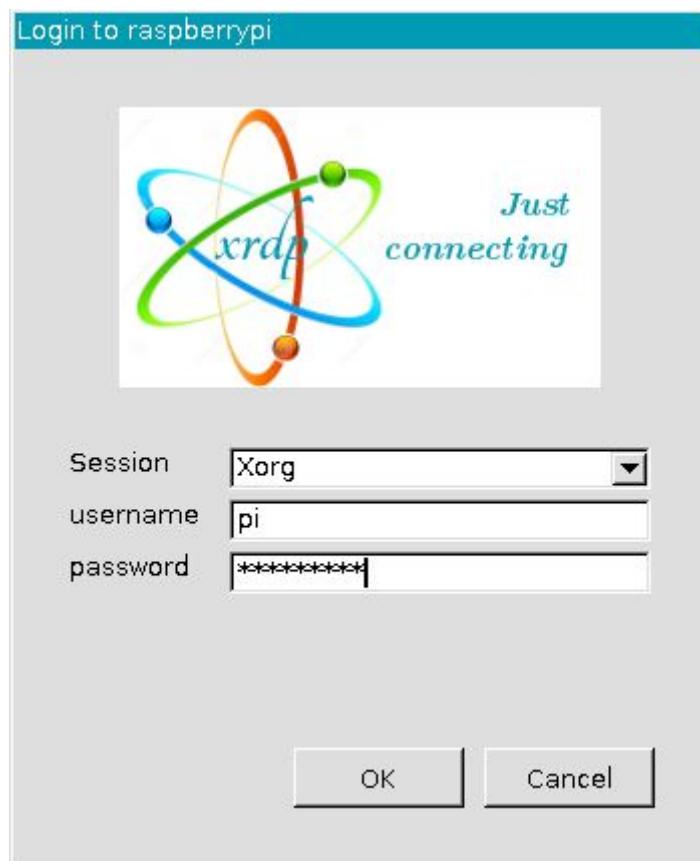
Damit man ein GUI für den RasPi hat kann man sich via VNC Viewer damit verbinden. Wenn man jedoch nicht eine weitere Datei installieren möchte, kann man auch einfach folgende Paket installieren und via RDP darauf zugreifen.

```
pi@z54:~ $ sudo apt-get install xrdp
```

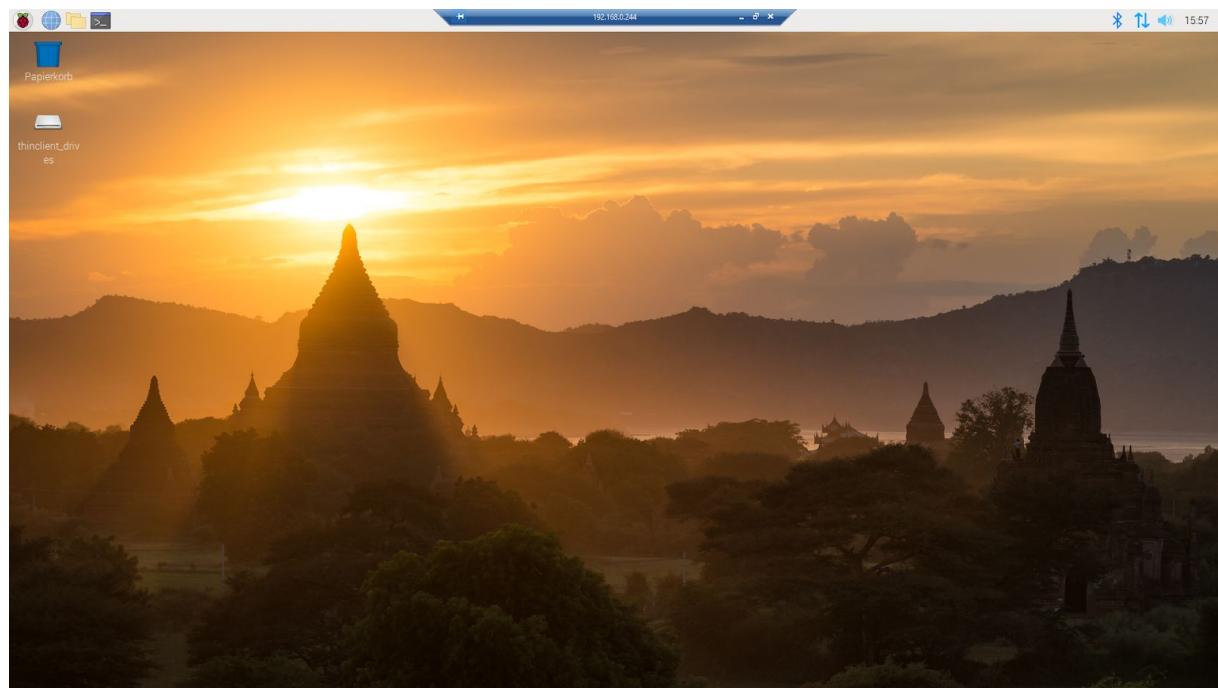
Danach kann man RDP öffnen und die IP des RasPi eingeben. Es kommt eine Warnmeldung diese einfach mit "Ja" bestätigen.



Nun kann man einfach die entsprechenden Credentials eingeben.



Und nun sieht man die Benutzeroberfläche, die der von VNC identisch ist.



Glossar

Proxy

Was ist ein Proxy Server?

Wenn Sie mit Ihrem Computer eine Internetverbindung aufbauen, zum Beispiel um eine Webseite aufzurufen, senden Sie als Client Ihre IP-Adresse, die zu Ihrem PC gehört, vergleichbar mit einem digitalen Fingerabdruck. Wenn Sie als Client eine Webseite abrufen, schicken Sie eine Anfrage an den Server, auf dem die Inhalte der Webseite liegen, und der Server sendet die Inhalte an Sie zurück. Ein Proxy-Server ist ein Server, der zwischen Ihren PC und den Ziel-Server geschaltet wird. Er fungiert sozusagen als Mittelsmann. Wenn Sie mit aktiviertem Proxy-Server eine Anfrage senden, kommt diese zunächst beim Proxy-Server an. Dieser sendet die Anfrage dann mit seiner eigenen IP-Adresse weiter. Die Inhalte vom Zielserver werden an den Proxy-Server zurückgeschickt und dann zu Ihnen weitergeleitet.

Wozu dient ein Proxy Server

Ein Proxy-Server kann verschiedenen Zwecken dienen:

- Offensichtlich ist, dass Sie mit einem Proxy-Server Ihre Identität verschleiern können. Es wird nicht Ihre eigene IP-Adresse gesendet, sondern die des Proxys.
- Proxy-Server können aber auch als Cache dienen. Webseiten oder Inhalte, die oft benötigt werden, können auf dem Proxy-Server zwischengespeichert werden. So muss nicht jede Anfrage, die beim Proxy ankommt, weiter zum Ziel-Server geleitet werden. Wenn der angefragte Inhalt im Proxy-Cache liegt, wird er direkt zu Ihnen zurückgesendet.
- Wenn Sie einen Proxy-Server verwenden, der im Ausland steht, können Sie dem Ziel-Server vorgaukeln, die Anfrage käme nicht aus der Schweiz. So können Sie zum Beispiel gesperrte YouTube Videos umgehen.

localhost

Grundsätzlich : der Rechner, vor dem Sie momentan sitzen.

Anwendungsprogramme, die netzwerkfähig sind, müssen irgendwie getestet werden. Da nun nicht für jeden Testlauf erst irgendwelche Teile des Programms auf einen anderen Computer übertragen werden sollen, können derartige Anwendungen auch auf dem eigenen Computer laufen. Sinnvoll ist so etwas beispielsweise für Webserver oder Spieleserver.

Dabei stellt der Computer sozusagen eine virtuelle Netzwerkkarte bereit. Diese sendet nun alle ausgehenden Daten an ihren eigenen Eingang zurück. Dieses Verfahren wird auch als

"Loopback" bezeichnet. Aus diesem Grunde kann man die Bezeichnung "localhost" und "lokaler Loopback" synonym verwenden.

Wenn Sie also auf Ihrem System einen Webserver installiert hätten, könnten Sie ihn über die Adresse <http://localhost> erreichen. Ebenso können Sie die für diesen Zweck reservierte IP-Adresse 127.0.0.1 verwenden.

Putty

PuTTY ist eine freie Software zum Herstellen von Verbindungen über Secure Shell, Telnet, Remote login oder serielle Schnittstellen. Dabei dient PuTTY als Client und stellt die Verbindung zu einem Server her. Beim Verbindungsaufbau wird die Identität des Benutzers mittels einer der bereitgestellten Methoden zur Authentifizierung überprüft. PuTTY ist für Windows und Linux verfügbar.

VNC

Virtual Network Computing, kurz VNC, ist eine Software, die den Bildschirminhalt eines entfernten Rechners (Server) auf einem lokalen Rechner (Client) anzeigt und im Gegenzug Tastatur- und Mausbewegungen des lokalen Rechners an den entfernten Rechner sendet. Damit kann man auf einem entfernten Rechner arbeiten, als säße man direkt davor. VNC implementiert das Remote Framebuffer Protocol und ist damit, im Gegensatz zu anderer Fernwartungs Software, plattformunabhängig nutzbar.

Stream-Multiplexing

Stream-Multiplexing (oft als "Stream-Muxing" abgekürzt) ermöglicht mehrere unabhängige logische Ströme, die sich alle ein gemeinsames zugrunde liegendes Transportmedium teilen.

Heartbeat-Mechanismus

Ein Heartbeat-Protokoll wird im Allgemeinen verwendet, um die Verfügbarkeit einer Ressource, wie z.B. einer schwebenden IP-Adresse, auszuhandeln und zu überwachen. Wenn ein Heartbeat auf einer Maschine startet, führt es normalerweise einen Wahlprozess mit anderen Maschinen im Heartbeat-Netzwerk durch, um festzustellen, welche Maschine, wenn überhaupt, die Ressource besitzt. Bei Heartbeat-Netzwerken mit mehr als zwei Maschinen ist es wichtig, die Partitionierung zu berücksichtigen, bei der zwei Hälften des Netzwerks zwar funktionieren, aber nicht in der Lage sind, miteinander zu kommunizieren. In einer solchen Situation ist es wichtig, dass die Ressource nur einer Maschine gehört und nicht einer Maschine in jeder Partition.

Da ein Heartbeat zur Anzeige des Zustands einer Maschine verwendet werden soll, ist es wichtig, dass das Heartbeat-Protokoll und der Transport, auf dem es läuft, so zuverlässig wie möglich ist. Die Verursachung eines Failover aufgrund eines Fehlalarms kann je nach

Ressource höchst unerwünscht sein. Es ist auch wichtig, schnell auf einen tatsächlichen Ausfall zu reagieren, daher ist es auch hier wichtig, dass der Herzschlag zuverlässig ist. Aus diesem Grund ist es oft wünschenswert, dass der Heartbeat über mehr als einen Transport läuft, z.B. über ein Ethernet-Segment mit UDP/IP und eine serielle Verbindung.

Polling-Anfragen

Polling ist in der Informatik der Ausdruck für eine zyklische Abfrage, um den Status von Hard- oder Software oder das Ereignis einer Wertänderung mit zyklischem Abfragen zu ermitteln. Hardware, deren Status benötigt wird, besteht häufig aus Ports elektrischer Schaltungen, Schnittstellen oder externen Geräten. Software-Zustände, die häufig abgefragt werden, sind Dateisperren auf anderen Rechnern oder Semaphore. Ein möglicher Zweck des Pollings ist das aktive Warten auf Zustandsänderungen, auch Spinning genannt. Eine andere Form ist die Abfrage jeweils einmal in einem Abtastzyklus, oder die Abfrage nach jeweils einer anderen Aktivität. Ein Beispiel für die Nutzung von Polling ist der Sendeaufruf.

Firewall

Auf meiner [Website](#) kann man Informationen zu Firewall einsehen.

RAID

Das Ziel eines RAID (Redundant Array of Independent Disks oder zu Deutsch: Redundante Anordnung unabhängiger Festplatten) ist Daten zu sichern, ohne dass diese verloren gehen.

Software vs. Hardware RAID

Software RAID

Bei einem Software-RAID übernimmt eine Software das Organisieren der Festplatten. Da keine eigene Hardware für diese Software bereitgestellt wird, muss die CPU einspringen. Eigentlich so gut wie jedes moderne OS wie Apple Mac OS X, Linux (Ubuntu, CentOS etc.) oder auch Microsoft Windows ab Windows NT sind in der Lage ein Software-RAID zu realisieren.

Vorteile: Kein separaten RAID-Controller wird benötigt, dadurch kann man Kosten sparen.

Nachteile: Belastung der CPU. Verlangsamte Lese- und Schreibgeschwindigkeit, da CPU Ussage und eventuell noch anderen laufenden Diensten auf dem System.

Hardware RAID

Beim Hardware-RAID übernimmt, der RAID-Controller, die Organisation der verschiedenen Platten. Der RAID-Controller und die Festplatten sind meistens in einem eigenen Gehäuse, dem sogenannten Disk Array untergebracht.

Vorteile: Entlastung der Host-CPU durch den RAID-Controller, hohe Transferrate durch das Anbinden der Laufwerke über mehrere Kanäle

Nachteile: Hohe Anschaffungskosten für den RAID-Controller (besonders in Ansicht für den Privatgebrauch)

Cache RAID Controller

Caches im RAID-Controller können die Performance beim Schreiben von Daten beträchtlich erhöhen. Typisch sind aktuell beispielsweise 256, 512 oder 1024 MB Cache. Bei einem Stromausfall geht der Inhalt dieses Caches verloren ausser man schützt den Inhalt durch eine BBU/BBM (Battery Backup Unit, Battery Backup Module)

Sicherungstechniken

Spiegelung

Bei der Spiegelung wird der Datensatz komplett auf zwei verschiedenen Festplatten abgespeichert. Beim Ausfall einer Platte geht nichts verloren, da die ganzen Daten auf der zweiten Disk vorhanden sind. Sprich die Platte 1 und Platte 2 haben den selben Datensatz gespeichert.

Streifen

Beim Streifen wird der Datensatz aufgeteilt. Beim Ausfall einer Festplatte sind die Daten verloren. Mit Streifen wird die Lese- und Schreibgeschwindigkeit erhöht, da von mehreren Festplatten gleichzeitig gelesen werden kann.

Parität

Parität ergänzt jeden Streifen mit der Möglichkeit verlorene Datensätze wiederherzustellen. Ein

Datensatz wird mit Streifen auf mehrere Platten verteilt. Auf einer zusätzlichen Platte wird für jeden Streifen ein paritätswert «P» errechnet. Somit kann man, wenn eine Platte ausfällt, die fehlenden Informationen errechnen. Ein kleines Beispiel: Wenn Platte 1 den Wert «5» und Platte 2 den Wert «8», dann speichert die Parität die Summe der beiden zahlen, in diesem Fall «13». Wenn nun eine Platte ausfällt, kann man durch die Summe, den verlorenen Wert ausrechnen.

RAID 0

Bei einem RAID 0 werden mind. zwei Festplatten benötigt. Dadurch, dass die Daten auf mehrere Festplatten verteilt sind, wären alle Daten gelöscht, wenn eine Platte ausfällt. Raid 0 ist ein Verfahren ohne irgendeinen Schutz vor Verlust, dadurch das aber von beiden Platten gleichzeitig gelesen werden kann, ist die Lese- und Schreibgeschwindigkeit sehr hoch. Als Sicherungstechnik wird hier die Streifen-Technik verwendet. Der Vorteil des Raid 0 liegt ganz klar bei der Geschwindigkeit, es ist extrem schnell. Der Nachteil liegt bei der Ausfallsicherheit, denn diese ist nicht gegeben.

RAID 1

Bei einem RAID 1 werden mindestens 2 Festplatten benötigt. Die Daten werden jeweils gespiegelt, sprich die Spiegel-Technik (Mirroring) wird hier verwendet. Der Vorteil des RAID 1 liegen bei der Ausfallsicherheit, wenn eine Disk ausfällt sind die Daten noch vorhanden. Der Nachteil liegt bei der Nutzungskapazität der Disk. Dadurch das alles redundant ist, hat man nur die Hälfte des Speichers.

RAID 5

Bei einem RAID 5 benötigt man mindestens 3 Disks. Die Daten werden auf alle Disks verteilt. Zusätzlich wird ein paritätswert errechnet und gespeichert. Wenn eine Festplatte nun ausfallen sollte, kann der RAID-Controller anhand dieser Parität die fehlenden Daten errechnen. Dadurch das ein Paritätswert berechnet werden muss, benötigt ein System mit RAID 5 mehr Rechenleistung, was in einer langsamen Schreibgeschwindigkeit resultiert. Der grosse Vorteil liegt in der Ausfallsicherheit. Beim RAID 5 wird die Streifen Technik verwendet mit Parität. Demnach ist RAID 5 ein Hybrid aus RAID 0 und Parität.

RAID 6

Bei einem RAID 6 werden mindestens 4 Festplatten benötigt. Anders als beim RAID 5 wird hier nicht nur eine sondern zwei Schemata verwendet. Das heisst, es können maximal 2 Festplatten ausfallen, ohne das Daten verloren gehen. Das Speichern von Daten ist aufwendiger, da durch die zwei Paritäten, jedes Mal zwei Paritätsaktualisierung berechnet und geschrieben werden muss. Die Lesegeschwindigkeit ist wie beim RAID 5 recht schnell, daher ist ebenfalls das RAID 6 für Archivierungssysteme geeignet. Die Nutzungskapazität errechnet sich durch die Gesamtkapazität minus 2 Laufwerke.

RAID 10

Das RAID 10 vereint das RAID 0 und 1 miteinander. Dafür werden mindestens 4 Festplatten benötigt. Die daten werden zuerst mit Streifen in 2 Teile aufgeteilt und anschliessend gespiegelt gespeichert. Der Vorteil liegt ganz klar in der Geschwindigkeit und in der Ausfallsicherheit, jedoch ist die Nutzungskapazität mit 50 % relativ gering.

GrovePi Erklärung

GrovePi+ ist eine Zusatzplatine mit 15 4-poligen Grove-Schnittstellen, die die Grove-Sensoren mit der Raspberry Pi verbindet. Es ist die neueste Version, die perfekt mit dem Raspberry Pi Modell B/B+/A+/2/3 kompatibel ist.

GrovePi+ ist ein einfache zu handhabendes und modulares System für Hardware-Hacking mit der Raspberry Pi, ohne Löten oder Breadboards: Schließen Sie Ihre Grove-Sensoren an und beginnen Sie direkt mit der Programmierung. Grove ist eine einfache zu bedienende Sammlung von mehr als 100 preiswerten Plug-and-Play-Modulen, die die physikalische Welt erfassen und steuern. Durch den Anschluss von Grove-Sensoren an Raspberry Pi wird Ihr Pi in der physischen Welt gestärkt. Mit Hunderten von Sensoren, die aus den Grove-Familien ausgewählt werden können, sind die Möglichkeiten für die Interaktion endlos.

Linux Grundbefehle

Einsehbar auf it.luis-luescher.com

Fazit

Das ganze Modul M126 war für mich und ich denke ebenfalls für meine ganzen Mitschüler eine komplett neue Erfahrung. So war alles bis zum 13. März alles normal, wir kamen in die Schule und erlernte im Teil 1 des Modules einige Dinge in Bezug auf IoT (Internet of Things) sowie einiges an Grundwissen in Bezug auf den RasPi. Unser neu erlangtes Wissen wurde dann durch die LB1 geprüft, ich war eigentlich ganz zufrieden mit meinem Resultat. Danach ging es los mit dem Teil 2 des Moduls. So begannen wir zuerst mit dem Demoprojekt. Hier mussten wir mittels einer Website eine LED-Lampe auf dem GrovePi ein und ausschalten. Danach habe ich weitere Projekte angefangen und diese immer wieder an unseren Dozenten präsentiert. Der auch sehr zufrieden war mit meiner Arbeit. Im ganzen fand ich dieses Modul sehr spannend und hoffe, dass das nächste ebenfalls so wird.

Quellenangaben

Zusammenarbeit

Ich habe mit einigen Mitschülern zusammengearbeitet und ebenfalls mein Wissen geteilt. Ich konnte über verschiedene Wege wie zB. Microsoft Teams, Discord oder auch WhatsApp mich mit meinen Mitschülern austauschen. Besonders möchte ich an dieser Stelle mich bei Michalis Chatzimichalis bedanken, der mich bei einigen Werkstattaufträgen begleitet und unterstützt hat. Ich bin sehr stolz auf mich, dass ich andere Mitschüler beim bewältigen der Werkstattaufträge unterstützen konnte. Die Zusammenarbeit innerhalb der Klasse verlief

sehr gut! Zudem konnte ich auch klassenübergreifend zB. Simon Steiner, Fabio Lichtler oder auch Ryan Simmonds helfen und beraten.

Inhaltliche Quellenangaben

- <https://ngrok.com/docs>
- https://xavierberger.github.io/RPi-Monitor-docs/11_installation.html
- <https://www.pool.ntp.org/de/use.html>
- <http://www.webmin.com/>
- <https://www.dexterindustries.com/grovepi/>
- <https://raspap.com/>
- <https://www.pivpn.io/>
- <https://www.plex.tv/de/>
- <https://pi-hole.net/>
- <https://cuberite.org/>
- <https://docs.python.org/3/>
- <https://stackoverflow.com/>
- Arbeitsbuch Luis Lüscher CIT-OIW (SIX GROUP)