Zentrum für sichere Informationstechnologie – Austria
Secure Information Technology Center – Austria

A-SIT

Seidlgasse 22 / 9, 1030 Wien
Tel.: (+43 1) 503 19 63–0
Fax: (+43 1) 503 19 63–66

Inffeldgasse 16a, 8010 Graz
Tel.: (+43 316) 873-5514
Fax: (+43 316) 873-5520

http://www.a-sit.at
E-Mail: office@a-sit.at

DVR: 1035461
ZVR: 948166612
UID: ATU60778947

INSTALLATIONSANLEITUNG ZUR IOT-ISOLATION MITTELS MINI-ROUTER

Version 1.0 vom 23.05.2018
Autor – peter.aufner@iaik.tugraz.at

Abstract/Zusammenfassung: Dieses Dokument gibt eine Anleitung zur Einrichtung eines Raspberry Pi um beliebige Geräte, z.B. IoT Geräte, vom restlichem Heimnetzwerk abzuschotten.
Die Dokumentation richtet sich an Enthusiasten mit etwas Erfahrung im Umgang mit Linux und Verständnis für die Funktionalität ihres Heimnetzwerks.

Inhaltsverzeichnis

1. Voraussetzungen

In diesem Kapitel folgt eine kurze Aufstellung der benötigten Hardware und Software, sowie grundlegendes Wissen für das Projekt:

1.1. Hardware

Zum Nachvollziehen dieser Anleitung werden folgende Geräte benötigt:

- PC/Laptop mit MicroSD Karten Leser
  - Viele Computer haben nur einen SD Karten Leser eingebaut. MicroSD Karten können in diesem Fall mittels Adapter eingelegt werden. Diese liegen oftmals den MicroSD Karten bei.
  - Alternativ gibt es auch USB-MicroSD Karten Leser.
- Router mit mindestens einem freien Ethernet Port

    - Der Router muss es erlauben ein Gerät in die so genannte ‚DMZ'(DeMilitarized Zone) zu verschieben. Ob der vorhandene Router das unterstützt ist der Gebrauchsanweisung des Gerätes zu entnehmen.
    - Üblicherweise ist diese Funktionalität vorhanden.
- Raspberry Pi (mit W-LAN) inklusive MicroSD Karte und Netzteil
    - Für diese Anleitung wurde ein Raspberry Pi 3 verwendet. Es ist auch möglich einen W-LAN USB-Stick zu nutzen, jedoch muss darauf geachtet werden, dass dieser für die Verwendung als W-LAN Access Point geeignet ist.
- Ethernetkabel zum Anschluss es Raspberry Pi

1.2. Software

Folgende Software sollte vor Beginn der Anleitung bereits heruntergeladen bzw. installiert sein:

- Ein SSH Client:
    - Unter Linux im Allgemeinen bereits installiert, sonst nach ‚ssh' im Paketmanager suchen
    - Unter Macos bereits installiert
    - Unter Windows empfiehlt sich Putty [1].
- Ein Webbrowser
- Software zum Erstellen der MicroSD Karte für den Raspberry Pi
    - Hierfür bietet sich Etcher [2] für alle gängigen Betriebssysteme an.
- Das aktuelle Image von Raspbian [3] für den Raspberry Pi.

1.3. Hintergrundwissen

Um dieser Anleitung folgen zu können, ist es notwendig, mit zwei Konzepten vertraut zu sein:

- Verbindungen via SSH
- Textverarbeitung auf der Kommandzeile z.B. mit nano

1.3.1. Konvention

Befehle, welche mittels SSH an den Raspberry Pi übermittelt werden sollen, sind

```
in dieser Schrift geschrieben.
```

1.3.2. Verbindungen mit SSH

Unter Linux und Macos sind üblicherweise SSH Clients installiert. Um dieser Anleitung folgen zu können, genügt es, ein Terminal zu öffnen und darin den Befehl:

```
ssh <user>@<IP-Adresse>
```

einzugeben. In der Praxis sieht das in etwa so aus:

```
ssh pi@192.168.1.100
```

Unter Windows ist die Verwendung von Putty empfohlen. Hier genügt es im Hauptfenster unter ‚Hostname' die gewünschte IP-Adresse einzutragen und anschließend auf ‚Open' zu drücken. Für eine umfangreichere Einführung in Putty, wird [4] (Englisch) empfohlen.

Will man eine SSH Verbindung, egal ob unter Linux/Macos oder Putty beenden, genügt die Eingabe von:

```
exit
```

1.3.3. Textverarbeitung auf der Kommandozeile mit nano

Auf der Linux Kommandozeile stehen viele Textverarbeitungswerkzeuge zur Auswahl. Für diese Anleitung wird ‚nano' verwendet. Dieser Editor ist relativ simpel gestaltet und genügt vollkommen für die Änderungen, die vorgenommen werden.

Um eine Datei zu öffnen, gibt es den folgenden Befehl:

```
nano <Dateiname>
```

Um zum Beispiel ‚testdatei' zu öffnen:

```
nano testdatei
```

Nano erlaubt die Steuerung mittels Pfeiltasten. ‚Pfeil hinunter' wechselt in die nächste Zeile, ‚Pfeil hinauf' in die davor, ‚Pfeil nach links' geht nach links, ‚Pfeil nach rechts' geht nach rechts. Ebenso kann Pos1 verwendet werden um zum Zeilenanfang zu gelangen. Ende um zu Ende der Zeile zu kommen.

In der Anleitung wird häufig davon geschrieben werden eine Datei zu ‚speichern und schließen'. Wenn diese Aussage fällt, ist damit gemeint:
Zuerst Strg gedrückt halten und dabei dann ‚O' drücken, anschließend Strg loslassen. Das öffnet den ‚Speichern Dialog' von nano. Hier mit Druck auf die ‚Enter' Taste das Speichern der Datei bestätigen.

Anschließend wieder Strg gedrückt halten und dabei ‚X' drücken, anschließend Strg loslassen. Damit wird nano beendet und man ist auf der Kommandozeile zurück.

Eine umfangreichere Erklärung zur Verwendung von nano findet sich beispielsweise unter [5] (Englisch).

1.3.4. Sudo

Raspbian, das Betriebssystem des Raspberry Pi basiert auf Linux. Dieses Betriebssystem steht für höchste Sicherheit und gibt einer Anwenderin/einem Anwender daher von vorne herein nur eingeschränkte Rechte. Der Befehl ‚sudo' erlaubt es einen einzelnen Befehl mit den höchsten Rechten am System auszuführen. Zum Beispiel:

```
sudo date
```

Im Lauf dieser Anleitung werden nahezu alle Befehle mit sudo beginnen. Bei jedem Aufruf von sudo mit egal welchem Befehl wird erneut das Passwort, mit dem man sich davor eingeloggt hat verlangt. Da das sehr mühsam werden kann, empfiehlt es sich durch die Eingabe:

```
sudo su
```

für den Rest der Sitzung erhöhte Rechte zu erlangen. Das vorangestellte ‚sudo' vor allen Befehlen kann dann entfallen.

2. Beispielinstallation

Die gesamte Anleitung geht davon aus, dass der Raspberry Pi ohne eigenem Monitor verwendet wird. Darum wird die Konfiguration des Raspberry Pi vollständig über SSH vorgenommen.
Der PC bzw. das Laptop wird im Folgenden als ‚Computer' bezeichnet.

2.1. Vorbereitung der MicroSD Karte

Um die MicroSD Karte für den Raspberry Pi vorzubereiten, legt man diese in den MicroSD Karten Leser des Computers. Anschließend startet man Etcher.

ACHTUNG: Es empfiehlt sich vor diesem Schritt alle externen Festplatten und eingelegte Speicherkarten vom Computer zu entfernen. **Wird im Folgenden das falsche Speichermedium ausgewählt, führt das zu permanentem Datenverlust!**

Das Programm zeigt 3 Schritte an:

1. Auswahl des Images: Hier ist das zuvor heruntergeladene Image von Raspbian auszuwählen.
2. Auswahl des zu beschreibenden Geräts. Etcher zeigt hier normalerweise nur externe Speichermedien an. Hier ist die MicroSD Karte auszuwählen.
3. ‚Flash!' mit Druck auf diesen Button wird die Korrektheit der Auswahl bestätigt und die MicroSD Karte beschrieben.

Sobald Etcher den Abschluss des Vorganges meldet, ist Raspbian auf der MicroSD Karte bereit.

2.2. SSH Zugriff am Raspberry Pi einrichten

Bevor die MicroSD Karte in den Raspberry Pi eingelegt wird, ist es notwendig den SSH Zugriff zu ermöglichen. Sollte die MicroSD Karte am Computer nicht als Laufwerk angezeigt werden, ist es notwendig sie einmal aus dem Computer zu entfernen und erneut anzuschließen.

Unter Windows wird zu diesem Zeitpunkt ausschließlich die ‚boot' Partition der MicroSD Karte angezeigt. Unter Linux werden möglicherweise sowohl die ‚boot' als auch die ‚root' Partition eingebunden.

Um SSH auf dem Raspberry Pi zu aktivieren, ist es lediglich notwendig, eine Datei mit dem Namen ‚ssh', ohne Dateiendung anzulegen. Der Inhalt dieser Datei ist egal, sie kann ruhig leer sein. Unter Windows ist zu beachten, dass standardmäßig eine Dateiendung angehängt wird. Diese muss inklusive dem Punkt entfernt werden, sodass die Datei wirklich nur ‚ssh' heißt. [4]

2.3. Erste Inbetriebnahme des Raspberry Pi

Nachdem die MicroSD Karte vorbereitet ist, kann sie in den Raspberry Pi eingelegt werden. Anschließend empfiehlt es sich den Raspberry Pi per Ethernetkabel mit dem Router zu verbinden und anschließend durch Einstecken der Stromversorgung einzuschalten. Sobald er mit dem Strom verbunden ist, schaltet sich er Raspberry Pi von selbst ein. Zur Kontrolle des Bootvorgangs, kann er via HDMI mit einem Monitor verbunden werden.

Der Login am Raspberry Pi ist mit den Benutzerdaten:

User: pi
Passwort: raspberry

möglich. Diese gelten lokal, wenn man sich direkt am Raspberry Pi anmelden will, und via SSH.

Bevor die Installation fortgesetzt werden kann, muss die IP Adresse des Raspberry Pi ermittelt werden. Das funktioniert am einfachsten über das Web Interface des Routers. Die Details unterscheiden sich hier zwischen den unterschiedlichen Modellen. Im Allgemeinen gibt es jedenfalls eine Möglichkeit, die ‚DHCP Clients' o.ä. im Web-Interface des Routers anzuzeigen, wodurch es möglich ist, die IP Adresse des Raspberry Pi zu ermitteln.

Alternativ kann man ein Keyboard und Monitor an den Raspberry Pi anschließen und sich lokal einloggen.
Dann ist es möglich mittels des Befehls:

```
ifconfig
```

auf der Kommandozeile die IP Adresse des Raspberry Pi zu ermitteln.

Ist die IP Adresse bekannt, kann der Login am Raspberry Pi via SSH erfolgen. Anschließend kann mit der Einrichtung fortgefahren werden.

2.4. Allgemeine Einstellungen

Bevor die Konfiguration zum W-LAN Router beginnt, lohnt es sich einige allgemeine Einstellungen vorzunehmen.

Raspbian bringt ein nützliches Konfigurationswerkzeug, genannt ‚raspi-config', mit. Dieses lässt sich mit dem Befehl:

```
sudo raspi-config
```

ausführen.

Nach dem Start wird ein Menü angezeigt, welches mit den Pfeiltasten am Keyboard, sowie Enter (um einen Eintrag auszuwählen) und Esc (um aus einem Untermenü zurück zu kehren) zu bedienen ist.
Zuerst empfiehlt sich den Eintrag '4 Localisation Options' auszuwählen und darin 'Change Timezone' um die Zeitzone anzupassen.
Außerdem ist es besonders wichtig unter 'Change Wi-fi Country' das Land korrekt einzustellen, da sonst illegale W-LAN Frequenzen genutzt werden könnten.

Anschließend kann optional unter '7 Advanced Options' die Auswahl 'A1 Expand Filesystem' genutzt werden. Damit erweitert der Raspberry Pi beim nächsten Neustart seine Partition automatisch so, dass er die gesamte MicroSD nutzt.

Nach dem Verlassen von raspi-config mittels mehrfachen Druck auf die ‚Esc'-Taste, ist es Zeit das System auf den neuesten Stand zu bringen. Dazu genügen folgende Befehle:

```
sudo apt-get update
sudo apt-get -y dist-upgrade
```

Es empfiehlt sich diese Befehle auch später manuell oder automatisch regelmäßig auszuführen, da so wichtige Sicherheitsupdates in das System eingespielt werden.

Außerdem ist jetzt ein guter Zeitpunkt sein persönliches Passwort für den Zugriff auf den Raspberry Pi zu vergeben. Das tut man mit dem Befehl:

```
passwd
```

Man wird anschließend nach dem aktuellen Passwort (raspberry) gefragt und muss im Anschluss zwei Mal sein neues Passwort eingeben. Es ist hier besonders wichtig ein sicheres Passwort zu wählen, welches Buchstaben, Ziffern und Sonderzeichen enthält, da es den einzigen Schutz des Raspberry Pi vor unbefugtem Zugriff darstellt.

Bevor es mit der Konfiguration des W-LAN Access Points losgeht, sollte man den Raspberry Pi mit

```
sudo reboot
```

neustarten, damit die aktualisierte Software aktiviert und die Partition vergrößert wird.

2.5. Konfiguration des W-LAN Access Points

Nachdem der Raspberry Pi neugestartet hat und man sich wieder mittels SSH verbunden hat, wird es Zeit, die nötigen Pakete für die Verwendung als W-LAN Access Point zu installieren. [5] Dies funktioniert mit dem Befehl:

```
sudo apt-get install dnsmasq hostapd
```

Damit werden zwei Programme installiert:

1. Dnsmasq: Stellt einen DHCP Server zu Verfügung, damit die Geräte mit IP-Adressen versorgt werden. Außerdem leitet er zentral DNS Anfragen nach außen weiter.
2. Hostapd: Erlaubt es dem Raspberry Pi als Access Point aufzutreten.

Wir gehen im Folgenden davon aus, dass der W-LAN Adapter, den wir als Access Point nutzen wollen, den Namen ‚wlan0' bekommen hat. Verwendet man den eingebauten Adapter des Raspberry Pi, sollte dies immer der Fall sein. Möchte man einen anderen Adapter verwenden, kann der Name mit dem Befehl:

```
sudo ifconfig
```

abgefragt werden.

2.5.1. Vorbereitung des W-LAN Adapters

Als Erstes muss der W-LAN Adapter von der Liste der DHCP Clients ausgenommen werden. Das funktioniert mit dem Befehl:

```
sudo echo 'denyinterfaces wlan0' >> /etc/dhcpcd.conf
```

Anschließend benötigt der W-LAN Adapter eine statische IP Adresse. Diese muss außerhalb des Blocks sein, der im bestehenden Heimnetz verwendet wird. (Den Block des bestehenden Heimnetzes kann man im Web-Interface des Routers erfahren.) Für die folgende Konfiguration wird der IP-Block: 192.168.20.0 mit der Subnetzmaske: 255.255.255.0 verwendet. Der Raspberry Pi bekommt die IP-Adresse 192.168.20.1. Diese Konfiguration kann entsprechend des bestehenden Heimnetzes und eigener Wünsche angepasst werden.

Um die statische IP Adresse zu setzen, sind folgende Befehle nötig:

```
sudo echo 'allow-hotplug wlan0' >> /etc/network/interfaces

sudo echo 'iface wlan0 inet static' >> /etc/network/interfaces

sudo echo '    address 192.168.20.1' >> /etc/network/interfaces

sudo echo '    netmask 255.255.255.0' >> /etc/network/interfaces

sudo echo '    network 192.168.20.0' >> /etc/network/interfaces

sudo echo '    broadcast 192.168.20.255' >> /etc/network/interfaces
```

2.5.2. Konfiguration von hostapd

Nachdem der W-LAN Adapter vorbereitet ist, wird es Zeit den Access Point zu konfigurieren.

Zuerst wird die Konfigurationsdatei mit dem Befehl:

```
sudo nano /etc/hostapd/hostapd.conf
```

geöffnet. Anschließend muss sichergestellt werden, dass folgende Zeilen in der Datei enthalten sind. (Es ist möglich, dass einige der Zeilen schon richtig konfiguriert zu finden sind, oder mit abweichender Konfiguration. Diese kann einfach angepasst werden.)

Zeilen, die im folgenden Block mit einer Raute (#) beginnen, dienen als Hinweis für die Konfiguration und müssen nicht abgeschrieben werden.

```
interface=wlan0

driver=nl80211

# Der Name des W-LANs für die IoT Geräte

ssid=IoT-AP

hw_mode=g

channel=6

ieee80211n=1

wmm_enabled=1

ht_capab=[HT40][SHORT-GI-20][DSSS_CCK-40]

macaddr_acl=0

auth_algs=1

ignore_broadcast_ssid=0

wpa=2

wpa_key_mgmt=WPA-PSK

# Passwort für das W-LAN. Dieses wird benötigt um später die IoT Geräte
zu verbinden.

wpa_passphrase=himbeerkuchen

rsn_pairwise=CCMP
```

Wenn die Änderungen gemacht sind, kann die Datei gespeichert und geschlossen werden.

Bevor die Konfiguration von hostapd abgeschlossen ist, muss noch bekannt gegeben werden, wo die Konfigurationsdatei liegt. Dazu wird wieder nano verwendet:

```
sudo nano /etc/default/hostapd
```

in dieser Datei gibt es eine Zeile:

```
#DAEMON_CONF=""
```

diese muss mit:

```
DAEMON_CONF="/etc/hostapd/hostapd.conf"
```

ersetzt werden. Wichtig ist hierbei, dass die Raute am Anfang der Zeile entfernt wird. Anschließend wird die Datei gespeichert und geschlossen.

2.5.3. Konfiguration von DNSMasq

Nach dem vorherigen Schritt ist der Raspberry Pi bereit für Verbindungen von anderen Geräten. Jedoch würden diese noch keine IP Adresse zugewiesen bekommen und auch nicht auf das Internet zugreifen können. Darum folgt die Konfiguration von DNSMasq.

Zuerst wird die originale Konfigurationsdatei mit dem Befehl:

```
sudo mv /etc/dnsmasq.conf /etc/dnsmasq.conf.orig
```

gesichert.

Anschließend wird mit dem Befehl:

```
sudo nano /etc/dnsmasq.conf
```

eine neue Datei angelegt.

In diese ist der folgende Inhalt zu schreiben: (Zeilen, die mit Raute beginnen, sind wieder Kommentare für das bessere Verständnis und müssen nicht abgeschrieben werden)

```
# Hier muss der Name des vorher festgelegten W-LAN Adapters eingetragen
werden
interface=wlan0

# Hier die vorher festgelegte IP Adresse
listen-address=192.168.20.1

bind-interfaces

# Anschließend werden DNS Server zum Weiterleiten von Anfragen
eingetragen. Hier können beliebige DNS Server genutzt werden, z.B. die
von Google

server=8.8.8.8
server=1.1.1.1

domain-needed

bogus-priv

# Hier wird festgelegt welche IP Adressen an verbundene Geräte vergeben
werden sollen. Die folgende Zeile sagt aus: Verwende IP Adresse von
192.168.20.2 bis 192.168.20.253 und sichere eine bestimmte IP-Adresse
einem Gerät für 12 Stunden zu.

dhcp-range=192.168.20.2,192.168.20.253,12h
```

2.5.4. Zugriff auf das Internet und Schutz vor ungewolltem Zugriff aus dem Internet

Aktuell ist der Raspberry Pi in der Lage ein W-LAN bereitzustellen, Geräten, die sich damit verbinden eine IP-Adresse zuzuweisen und sie mit den notwendigen Informationen zu versorgen, damit sie mit dem Internet kommunizieren können. Allerdings verhindert er die tatsächliche Kommunikation mit dem Internet.

Um das zu beheben, sind zwei Schritte notwendig:

Zuerst muss IP-Forwarding erlaubt werden. Dazu ist die Datei /etc/sysctl.conf zu öffnen.

```
sudo nano /etc/sysctl.conf
```

In dieser befindet sich eine Zeile:

```
#net.ipv4.ip_forward=1
```

von dieser ist die Raute am Anfang zu entfernen. Anschließend wird die Datei wieder gespeichert und geschlossen.

Im zweiten Schritt muss die Firewall konfiguriert werden. Wer schon mit Linux zu tun hatte, wird mit iptables vertraut sein. Die Konfiguration ist eher schwer verständlich, darum wird hier ‚ufw' verwendet. Ufw stellt eine verständlichere Schnittstelle zu iptables dar. Installiert wird ufw mit dem Befehl:

```
sudo apt-get install ufw
```

Als erstes wird NAT konfiguriert. [5] Dazu muss die Datei /etc/default/ufw geöffnet werden.

```
sudo nano /etc/default/ufw
```

und die Zeile, die mit DEFAULT_FORWARD_POLICY beginnt, so geändert werden, dass sie wie folgt aussieht:

```
DEFAULT_FORWARD_POLICY="ACCEPT"
```

Anschließend die Datei speichern und schließen.

Als nächstes wird die Datei /etc/ufw/before.rules geöffnet:

```
sudo nano /etc/ufw/before.rules
```

An das Ende dieser Datei ist der folgende Textblock einzufügen:

```
# NAT table rules
*nat
:POSTROUTING ACCEPT [0:0]

# Weiterleitung des Datenverkehrs, muss dem externen Ethernet Adapter
entsprechen. Dieser sollte standardmäßig eth0 sein. Die Zahl IP nach dem
‚-s' entspricht dem vorher gewählten Bereich für das IoT W-LAN.
-A POSTROUTING -s 192.168.20.0/24 -o eth0 -j MASQUERADE

COMMIT
```

Anschließend wieder speichern und schließen.

Jetzt müssen noch einige Freigaben für das IoT Netzwerk gemacht werden:

Um weiterhin Zugriff via SSH auf den Raspberry Pi zu haben, ist die folgende Zeile einzugeben:

```
sudo ufw allow from 192.168.20.0/24 to any port 22
```

Damit DNS Anfragen von den verbundenen Geräten bedient werden können:

```
sudo ufw allow from 192.168.20.0/24 to any port 53
```

Damit die verbundenen Geräte IP Adressen bekommen:

```
sudo ufw allow in on wlan0 from any port 68 to any port 67 proto udp
```

2.5.5. Abschluss der Konfiguration und Verbindung mit Geräten

Jetzt ist die Konfiguration des Raspberry Pi abgeschlossen. Um die konfigurierten Dienste zu aktivieren, sind folgende Eingaben notwendig:

```
sudo ufw enable

sudo systemctl enable hostapd

sudo systemctl enable dnsmasq
```

Abschließend muss der Raspberry Pi mit der Eingabe:

```
sudo reboot
```

neugestartet werden.

Damit wird die bestehenden SSH Verbindung abgebrochen und der Raspberry Pi startet neu. Der Raspberry Pi ist nach dem Neustart vom Heimnetzwerk nicht mehr via SSH erreichbar.

Prüft man nach erfolgtem Neustart des Raspberry Pi auf seinem Computer, welche W-LANs zu sehen sind, sollte das IoT W-LAN vom Raspberry Pi angezeigt werden. Man kann sich jetzt temporär mit dem IoT W-LAN verbinden um wieder SSH Zugriff zu erlangen, oder die Verbindung der IoT Geräte mit dem abgetrennten Netzwerk zu erleichtern.

Diese Anleitung kann auf die Details zum Verbinden der einzelnen IoT Geräte nicht eingehen, da diese zu vielfältig sind. Wenn eine App vorhanden ist und das Smartphone genutzt wird, empfiehlt es sich auf jeden Fall das Smartphone temporär mit dem IoT W-LAN zu verbinden um die Anbindung der IoT Geräte zu erleichtern.

Nach Abschluss der Einrichtung der IoT Geräte im abgeschotteten W-LAN, sollte dieses von allen anderen Geräten (Smartphone, Computer) wieder ‚vergessen' werden um irrtümliche Verbindungen damit zu vermeiden.

2.5.6. Verschieben des Raspberry Pi in die DMZ

Nachdem die Konfiguration abgeschlossen ist und die IoT Geräte mit dem Raspberry Pi verbunden sind, bleibt noch sie vom restlichen Heimnetz abzuschotten.

Dies geschieht über das Web-Interface des bestehenden Routers. Die Details dazu sind dessen Anleitung zu entnehmen. Zuerst muss dem Raspberry Pi eine statische IP Adresse im Heimnetz zugewiesen werden. Anschließend wird der Raspberry Pi in die DMZ verschoben. Generell ist nach dem Begriff ‚DMZ' oder ‚DeMilitarized Zone' zu suchen und wie man ein Gerät im Heimnetz in diese verschiebt.

2.6. Optional

2.6.1. Vergabe statischer IP Adressen

Eventuell möchte man zur einfacheren Verwaltung sicherstellen, dass die IoT Geräte im abgeschotteten Netz immer dieselbe IP Adresse zugewiesen bekommen. Hierfür genügen zwei Schritte:

Mit der Eingabe von

```
cat /var/lib/misc/dnsmasq.leases
```

wird eine Liste der aktuell zugewiesenen IP Adressen ausgegeben.

Diese enthält Zeilen, die wie folgt aussehen:

```
1526420798 12:34:45:67:ab:cd 192.168.20.13 testgerät *
```

Wichtig sind hier die zweite Spalte, welche die MAC Adresse und die dritte Spalte, welche die IP Adresse angeben.

Möchte man, dass einem Gerät immer dieselbe IP Adresse zugewiesen wird, öffnet man /etc/ethers mit:

```
sudo nano /etc/ethers
```

und befüllt diese mit Zeilen der Form <MAC> <IP>.

Der Inhalt der Datei würde dann zum Beispiel wie folgt aussehen:

```
12:34:45:67:ab:cd 192.168.20.13
12:34:45:67:ab:ef 192.168.20.20
```

Anschließend speichert und schließt man die Datei wieder.

Referenzen

[1] Putty, "Putty," [Online]. Available: https://putty.org/. [Accessed 14 05 2018].

[2] Etcher, "Etcher," [Online]. Available: https://etcher.io/. [Accessed 14 05 2018].

[3] R. P. Foundation, "Raspbian," [Online]. Available: https://www.raspberrypi.org/downloads/raspbian/. [Accessed 14 05 2018].

[4] ssh.com, "How to Use PuTTY on Windows," [Online]. Available: https://www.ssh.com/ssh/putty/windows/. [Accessed 15 05 2018].

[5] 'YatriTrivedi', "The Beginner's Guide to Nano, the Linux Command-Line Text Editor," [Online]. Available: https://www.howtogeek.com/howto/42980/the-beginners-guide-to-nano-the-linux-command-line-text-editor/. [Accessed 15 05 2018].

[6] R. P. Foundation, "SSH (Secure Shell)," [Online]. Available: https://www.raspberrypi.org/documentation/remote-access/ssh/ . [Accessed 14 05 2018].

[7] P. Martin, "Using your new Raspberry Pi 3 as a WiFi access point with hostapd," [Online]. Available: https://frillip.com/using-your-raspberry-pi-3-as-a-wifi-access-point-with-hostapd/ . [Accessed 15 05 2018].

[8] 'Kimus', "NAT and FORWARD with Ubuntu's ufw firewall," [Online]. Available: https://gist.github.com/kimus/9315140. [Accessed 15 05 2018].