

Doorpi installieren

am besten erst den ganzen Abschnitt lesen, besonders das mit dem patch!

<https://www.doorpi.org/forum/lexicon/entry/1-installation-doorpi-mittels-pypi-auf-einem-raspberry-pi-raspbian/>

Deinstallation von "unnötigen" #apt-get Paketen (kommen später über #pip / #easy_install wieder, wenn diese benötigt sind)

```
sudo apt-get update
```

```
sudo apt-get upgrade
```

Shell-Script

```
sudo apt-get remove python-pip python-rpi.gpio
```

```
0 aktualisiert, 0 neu installiert, 1 zu entfernen und 0 nicht akt
Nach dieser Operation werden 103 kB Plattenplatz freigegeben.
Möchten Sie fortfahren? [J/n] j
(Lese Datenbank ... 36345 Dateien und Verzeichnisse sind derzeit
Entfernen von python-rpi.gpio (0.6.3~jessie-1) ...
pi@Door:~ $
```

Aktuelle notwendig ist die Installation von #python-dev per apt-get (siehe auch Bug bei der Installation von DoorPi)

Shell-Script

```
sudo apt-get install python-dev
```

```
Es müssen 18,3 MB an Archiven heruntergeladen werden.
Nach dieser Operation werden 26,3 MB Plattenplatz zusätzlich benutzt.
Möchten Sie fortfahren? [J/n] j
```

```
Trigger für man-db (2.7.0.2-5) werden verarbeitet ...
libexpat1-dev:armhf (2.1.0-6+deb8u3) wird eingerichtet ...
libpython2.7-dev:armhf (2.7.9-2+deb8u1) wird eingerichtet ...
libpython-dev:armhf (2.7.9-1) wird eingerichtet ...
python2.7-dev (2.7.9-2+deb8u1) wird eingerichtet ...
python-dev (2.7.9-1) wird eingerichtet ...
pi@Door:~ $
```

Betriebssystem auf dem RaspberryPi aktualisieren

Shell-Script

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

```
Paketlisten werden gelesen... Fertig
Paketlisten werden gelesen... Fertig
Abhängigkeitsbaum wird aufgebaut.
Statusinformationen werden eingelesen... Fertig
Paketaktualisierung (Upgrade) wird berechnet... Fertig
0 aktualisiert, 0 neu installiert, 0 zu entfernen und 0 nicht aktualisiert.
Paketlisten werden gelesen... Fertig
Abhängigkeitsbaum wird aufgebaut.
Statusinformationen werden eingelesen... Fertig
Paketaktualisierung (Upgrade) wird berechnet... Fertig
0 aktualisiert, 0 neu installiert, 0 zu entfernen und 0 nicht aktualisiert.
pi@Door:~ $
```

Installation von pip über easy_install (Thema) sowie die Pakete #linphone4raspberrypi und #python-daemon, die in der requirements.txt sonst regelmäßig Probleme machen.

Shell-Script

```
sudo easy_install pip || (wget https://bootstrap.pypa.io/easy_install.py -O - | sudo python)
&& sudo easy_install pip
```

```
sudo pip install linphone4raspberrypi python-daemon
```

```
creating /usr/local/lib/python2.7/dist-packages/pip-9.0.1-py2.7.egg
Extracting pip-9.0.1-py2.7.egg to /usr/local/lib/python2.7/dist-packages
Adding pip 9.0.1 to easy-install.pth file
Installing pip script to /usr/local/bin
Installing pip2.7 script to /usr/local/bin
Installing pip2 script to /usr/local/bin

Installed /usr/local/lib/python2.7/dist-packages/pip-9.0.1-py2.7.egg
Processing dependencies for pip
Finished processing dependencies for pip
pi@Door:~ $
```

```
100% |#####| 342KB 102KB/s
Collecting lockfile>=0.10 (from python-daemon)
  Downloading lockfile-0.12.2-py2.py3-none-any.whl
Installing collected packages: linphone4raspberrypi, docutils, lockfile,
python-daemon
Successfully installed docutils-0.13.1 linphone4raspberrypi-3.9.0 lockf
python-daemon-2.1.2
pi@Door:~ $
```

Download, Installation und erster Start von DoorPi

Shell-Script

```
sudo pip install doorpi && sudo doorpi_cli --trace
```

Nun ist DoorPi installiert und das erste Mal gestartet. Die Weboberfläche (siehe FAQ) ist erreichbar und die passende URL wird von DoorPi ausgegeben:

Eventuell kann auch ein #Port angegeben sein, wenn DoorPi nicht an Port 80 gebunden

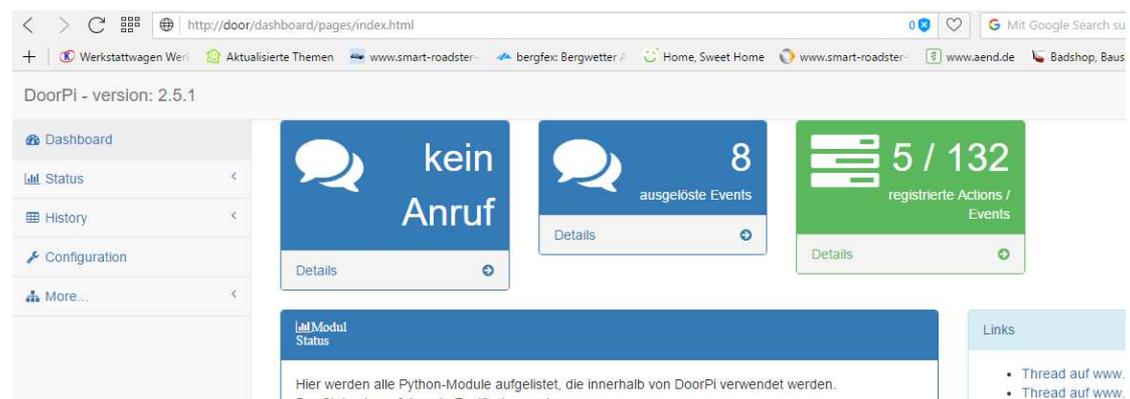
werden konnte (Default-Ausweichport wäre 8080).

Im Windows Explorer

door:8080 oder eben door:80 eingeben, door in meinem Fall Name des Raspis, in rasp-config definiert

Benutzer: door

Passwort pi



Mit der Tastenkombination [Strg] + [c] beende ich DoorPi und prüfe die entstandenen Dateien:

Datei `/usr/local/bin/doorpi_cli`

`cd /usr/local/bin/`

`ls -l`

`doorpi_cli`

vorhanden!

Datei `/etc/init.d/doorpi` (falls nicht vorhanden -> Neuerstellung daemon-file)

`cd /etc/init.d/`

`ls -l`

bei mir nicht vorhanden daher Neuerstellung (Link im Orginal Artikel)

und nun der Knackpunkt, nach 3 Fehlversuchen mit dem daemon file hab ich das hier gefunden:

<https://www.doorpi.org/forum/thread/12-doorpi-als-daemon-dienst-service-autostart-unter-jessie-einrichten/?pageNo=2>

wenn das Thema noch aktuell ist, hatte den auch diesen Fehler.

auch da war ein kleiner Fehler, watchdog ließ sich wegen fehlender Pakete erst nicht installieren

dann das hier vorab

```
sudo apt-get update && sudo apt-get -y install perl
```

dann ging es weiter mit

Lösung:

```
sudo apt-get install python-daemon python-watchdog
```

Überprüft das einfach mal.

Neuerstellung daemon-file

12. März 2016

Ausgangslage:

Bei der Installation wurde kein daemon-File erzeugt und / oder soll nachträglich erzeugt werden.

Achtung: Die Datei /etc/init.d/doorpi wird ohne Rückfrage überschrieben!

<https://www.doorpi.org/forum/thread/12-doorpi-als-daemon-dienst-service-autostart-unter-jessie-einrichten/?postID=33#post33>

```
cd /tmp
```

```
sudo wget
```

```
https://raw.githubusercontent.com/motom001/DoorPi/master/doorpi/docs/service/create\_daemon\_file.py -O - | sudo python
```

```
HTTP-Anforderung gesendet, warte auf Antwort... 200 OK
Länge: 923 [text/plain]
In »»STDOUTOUT«« speichern.

-                               100%[=====>]                923  --.-KB/s   in 0s

2017-05-19 16:39:38 (11,5 MB/s) - auf die Standardausgabe geschrieben [923/923]

start to create daemon file now...
start download and parse new daemon file:
URL:  https://raw.githubusercontent.com/motom001/DoorPi/master/doorpi/docs/service/doorpi.t
FILE:  /etc/init.d/doorpi
download successfully - change chmod to 0755 now
finished
pi@Door:/tmp $
```

```
sudo systemctl daemon-reload
```

sudo reboot

nach reboot kontrollieren

sudo systemctl daemon-reload

sudo /etc/init.d/doorpi start

[ok] Starting doorpi (via systemctl): doorpi.service.

sudo /etc/init.d/doorpi status

● doorpi.service - LSB: DoorPi

Loaded: loaded (/etc/init.d/doorpi)

Active: active (running) since Do 2017-05-04 22:50:19 CEST; 20s ago

Process: 4549 ExecStart=/etc/init.d/doorpi start (code=exited, status=0/SUCCESS)

CGroup: /system.slice/doorpi.service

└─4557 /usr/bin/python /usr/local/bin/doorpi_cli start --configfile /usr/local/etc...

<https://www.doorpi.org/forum/thread/12-doorpi-als-daemon-dienst-service-autostart-unter-jessie-einrichten/?postID=18#post18>

Danach wird dem System dieser Daemon in den "Autostart" eingetragen:

Quellcode

sudo systemctl daemon-reload

sudo update-rc.d doorpi defaults

```
pi@Door:~ $ sudo systemctl daemon-reload
pi@Door:~ $ sudo update-rc.d doorpi defaults
pi@Door:~ $
```

Fertig - mit dem nächsten Systemstart wird DoorPi automatisch gestartet und bei einem System-Shutdown wird DoorPi ordentlich runtergefahren.

Test:

sudo reboot

Test im Windows direkt nach Neustart



nicht erforderlich

Achtung Python 3 macht Probleme mit Python 2 von Doorpi nicht installieren

Installing PiFace Digital modules

With the latest version of Raspbian installed you can easily install PiFace Digital software with apt-get.

To install the PiFace Digital software run the command:

```
sudo apt-get install python3-pifacedigital-emulator
```

Enter **Y** when prompted.

orgelt recht lange rum

```
python3-pyside (1.2.2-1) wird eingerichtet ...
python3-pifacedigital-emulator (2.1.0-1) wird eingerichtet ...
Creating desktop icon.
vlc-plugin-notify (2.2.5-1~deb8ul) wird eingerichtet ...
vlc-plugin-samba (2.2.5-1~deb8ul) wird eingerichtet ...
Trigger für libc-bin (2.19-18+deb8u9) werden verarbeitet ...
pi@door:~ $
```

If you want to use Scratch with your PiFace Digital, you will need to install the PiFace Scratch

handler:

```
sudo apt-get install python3-pifacedigital-scratch-handler
```

Now reboot your Raspberry Pi

```
sudo reboot
```

Funktionstest des piface

To test all your PiFace software has installed and your RaspberryPi is set up correctly, type one of the following commands into a Terminal:

```
python3 /usr/share/doc/python3-pifacedigitalio/examples/blink.py
```

und es blinkt ;-)