# Inhalt

1 Inhalt 1

2 Allgemein 1

3 Lizenz 1

4 Installation 2

4.1 Installation und Einrichting Raspberry PI 2

4.1.1 Betriebssystem installieren 2

4.1.2 Betriebssystem einrichten 2

4.1.3 Anwendungsprogramme einrichten 3

4.2 Installation und Einrichtung WebExpress 5

4.2.1 WebExpress installieren 5

4.2.2 WebExpress einrichten 6

4.2.3 Weitere WebExpress-Anwendungen installieren 6

5 OpenSSL CA erstellen und Zertifikate ausstellen 6

5.1 Certificate Authority (CA) erstellen 6

5.1.1 Privater Schlüssel 6

5.1.2 Root-Zertifikat erstellen 7

5.2 Zertifikat ausstellen 7

5.2.1 Privater Schlüssel 7

5.2.2 Zertifikatsanfrage erstellen 7

5.2.3 Zertifikat ausstellen 8

5.3 Zertifikate installieren 8

5.3.1 Zertifikate in WebExpress installieren 8

5.3.2 Zertifikate im Client installieren 8

5.3.3 WebExpress-Zertifikat vertrauen 9

6 Installation und Einrichtung eines Reverse Proxys 9

6.1 Apache Webserver installieren 10

6.2 Apache Module installieren 10

6.3 Apache konfigurieren 10

7 Update 11

8 Einkaufsliste 11

9 Quellen 11

# Allgemein

WebExpress ist ein leichtgewichtiger Webserver, welcher für den Einsatz in leistungsarmen Umgebungen optimiert wurde. Schon auf kleinen Systemen, wie zum Beispiel der Rasspberry PI, können Webanwendungen effizient betrieben werden. Dies wird durch einen kleinen Fußabdruck mit geringer Ressourcenbelastung erreicht. Weiterhin verfügt WebExpress über ein leistungsstarkes und optimiertes Plugin-System, mit einer umfassenden API und Anwendungsvorlagen. Damit lassen sich Webanwendungen einfach und schnell in einer .Net-Sprache (z.B. C#) entwickeln.

# Lizenz

Die Software steht als Open-Source (MIT) zur freien Verfügung. Die Softwarequellen können über <https://github.com/ReneSchwarzer/WebExpress> bezogen werden.

WebExpress baut auf Komponenten auf, welche als Open-Source zur Verfügung stehen:

* <https://github.com/dotnet/core> (MIT)
* <https://getbootstrap.com/> (MIT)
* <https://www.chartjs.org> (MIT)
* <https://jquery.com/> (MIT)
* <https://summernote.org/> (MIT)
* <https://popper.js.org/> (MIT)
* <https://github.com/kurtobando/simple-tags> (MIT)
* <https://github.com/uxsolutions/bootstrap-datepicker> (Apache 2.0)

# Installation

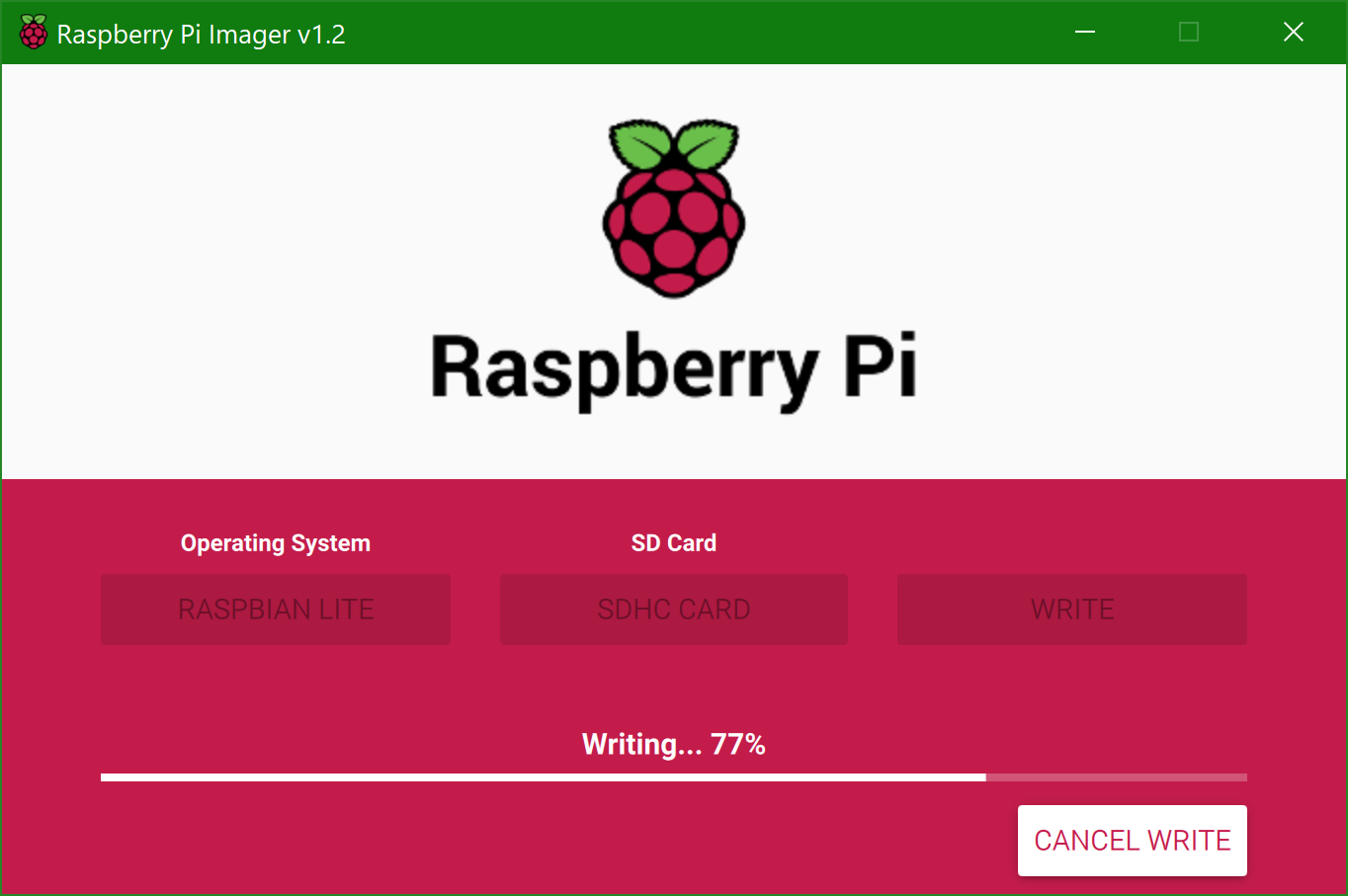
Die Installation wird anhand des Raspberry PI beschrieben. Die allgemeine Vorgehensweise kann jedoch auch auf andere Betriebssysteme angewendet werden.

## Installation und Einrichting Raspberry PI

Dank der schlanken Architektur von WebExpress läuft dieser und sicher auch auf Kleinstrechner, wie den Raspberry Pi performant. Um WebExpress nutzen zu können, sind einige Voraussetzungen auf den Raspberry Pi zu erfüllen:

### Betriebssystem installieren

Der Erste Schritt besteht darin, dass Betriebssystem auf eine SD-Karte zu schreiben. Hierzu gibt es unter <https://downloads.raspberrypi.org/imager/imager.exe> ein kostenloses Programm (Windows), mit deren Hilfe das Image auf die SD-Karte kopiert wird.



### Betriebssystem einrichten

Im zweiten Schritt wird die SD-Karte in dem Raspberry Pi eingesetzt und der Raspberry Pi gestartet. Da derzeit SSH noch nicht aktiv ist, muss eine Tastatur und ein Bildschirm angeschlossen werden. Wenn der Raspberry Pi gebootet wurde, kann das Anmelden mit

User: pi

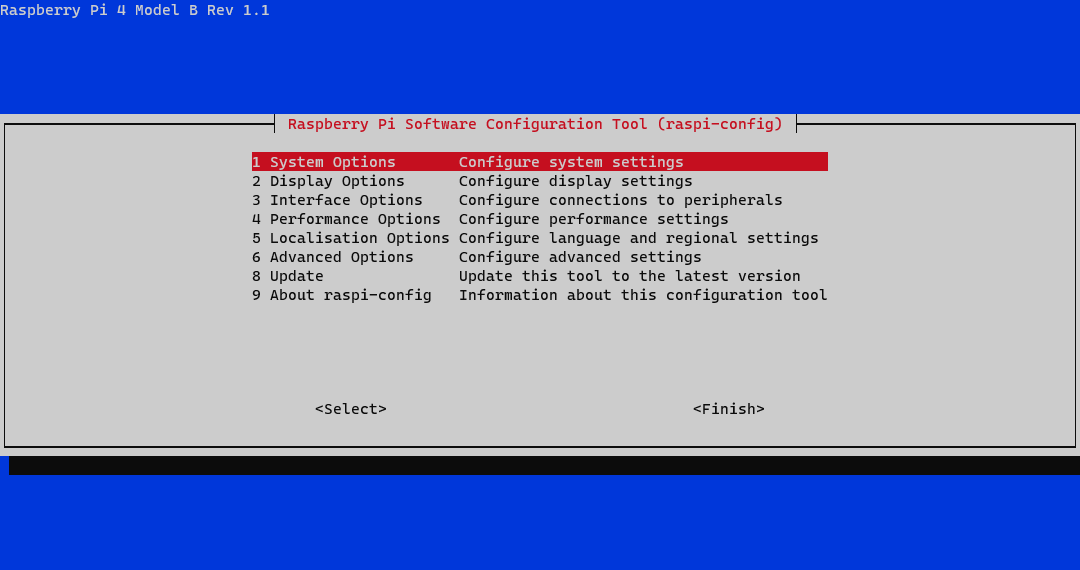
Passwort: rasperry

erfolgen.

Nach der erfolgreichen Anmeldung wird das Dienstprogramm raspi-config aufgerufen, mit deren Hilfe die Grundkonfiguration des Raspberrys erfolgt.

pi@raspberrypi:~ $ sudo raspi-config

Es wird empfohlen das Password zu ändern, sowie bei Bedatf WLAN-einzurichten, die Zeitzone und den Hostnamen (z.B. auf wx) zu ändern. Weiterhin ist SSH zu aktivieren (zu finden unter Interface Options).

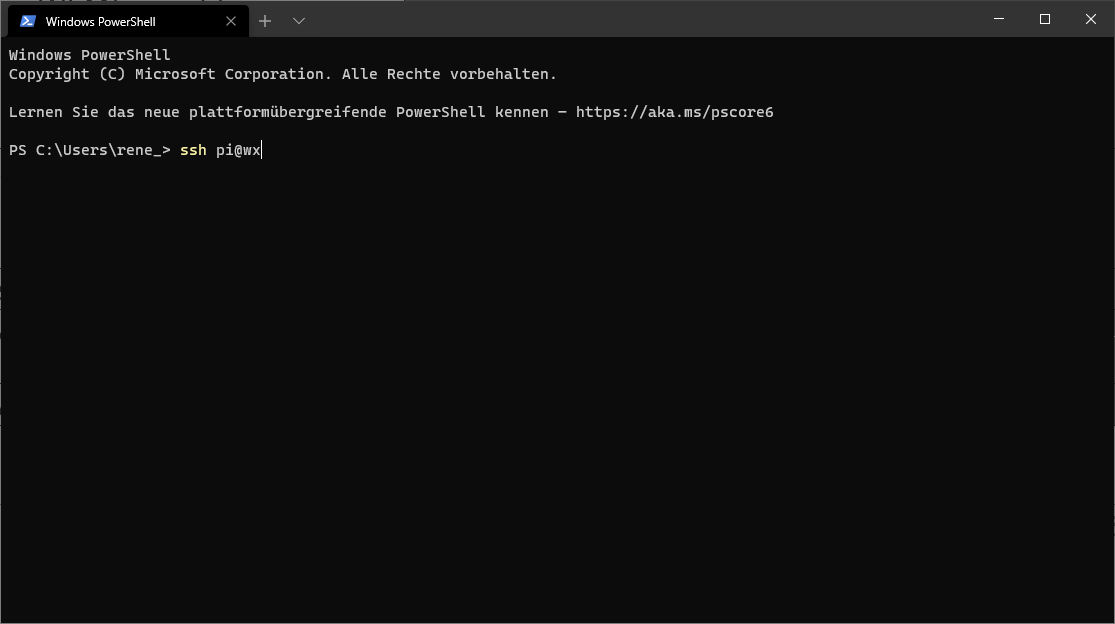


Alle nachfolgenden Schritte können nun über SSH erfolgen und der Raspberry Pi von der Tastatur und Bildschirm getrennt werden.

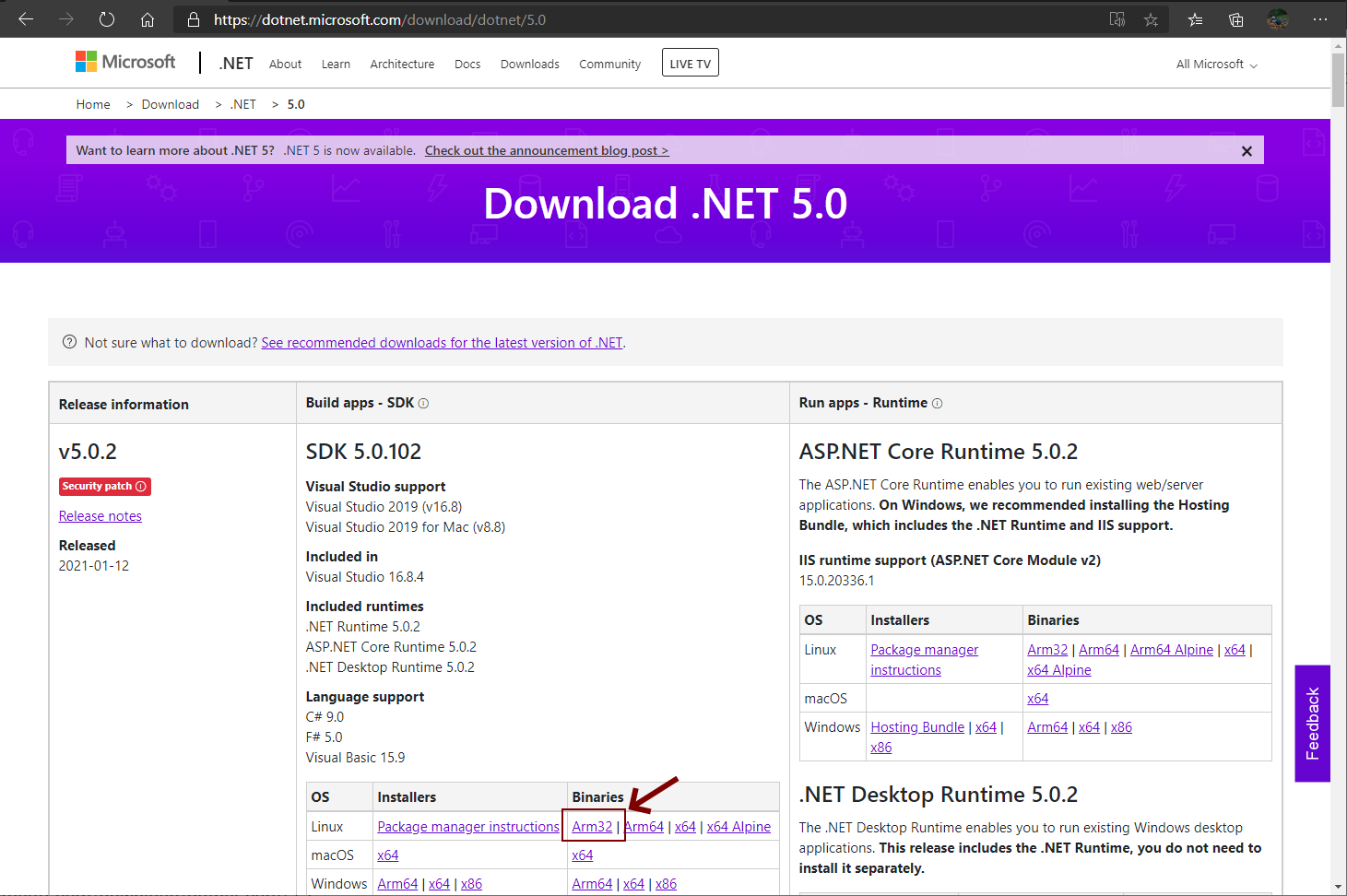
### Anwendungsprogramme einrichten

Im dritten Schritt werden weitere Programme installiert, welche für die Ausführung von WebExpress oder zur Administration benötigt werden.

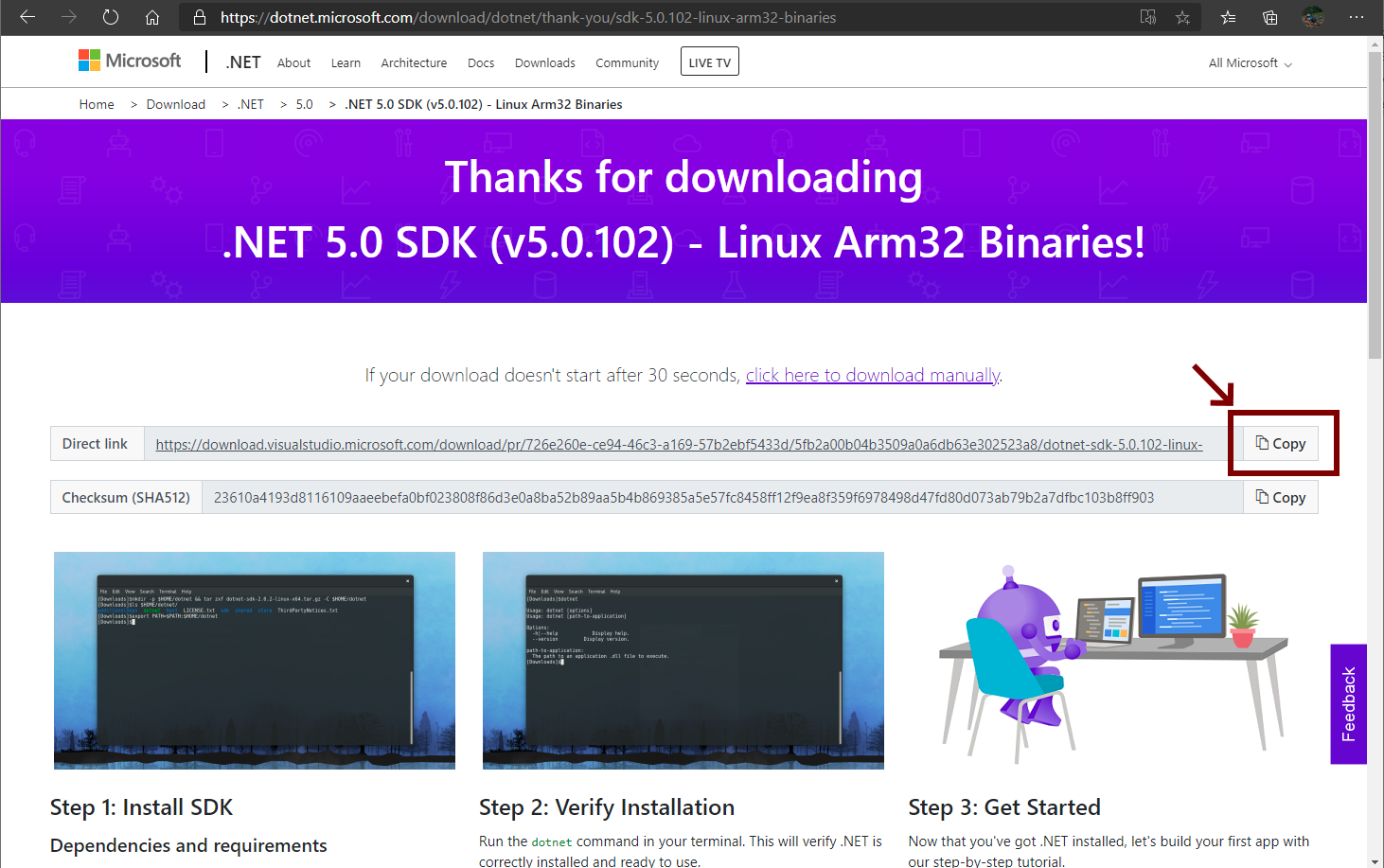
Nachdem SSH aktiviert wurde, kann mit Hilfe eines SSH-Clients (z.B. Putty, Windows Power Shell) eine Verbindung zum Raspberry Pi aufgebaut werden.



Zunächst muss .Net Core installiert werden. Hilfe hierzu wird unter [1] angeboten. Die aktuelle Version (derzeit 5.0.2) kann von Microsoft unter <https://dotnet.microsoft.com/download/dotnet-core> kostenlos bezogen werden.



Für den Raspberry Pi sind die Binaries für Linux-Arm32 zu verwenden. Der Direkt-Link zu den Linux-Arm32 Binaries ist zu kopieren.



Auf den Raspberry wird mittels wget das Linux-Arm32-Archiv heruntergeladen.

pi@wx:~ $ wget https://download.visualstudio.microsoft.com/download/pr/726e260e-ce94-46c3-a169-57b2ebf5433d/5fb2a00b04b3509a0a6db63e302523a8/dotnet-sdk-5.0.102-linux-arm.tar.gz

Vorbereitend für die Installation von .Net Core muss ein Verzeichnis unter /usr/share/dotnet-sdk angelegt werden, indem das .Net Core-Framework entpackt wird.

pi@wx:~ $ sudo mkdir /usr/share/dotnet-sdk

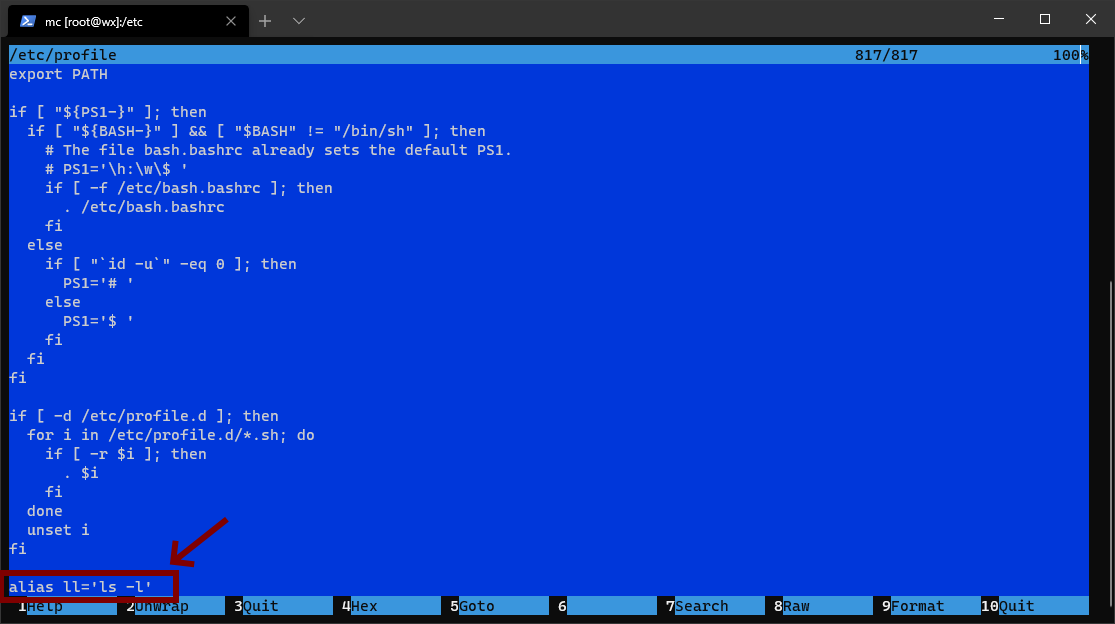
Nach Anlegen des Verzeichnisses /usr/share/dotnet-sdk können die Binaries entpackt werden.

pi@wx:~ $ sudo tar zxf dotnet-sdk-5.0.102-linux-arm.tar.gz -C /usr/share/dotnet-sdk/

Als optionale Anwendung kann noch der Midnight Commander (MC) installiert und das Profil angepasst werden.

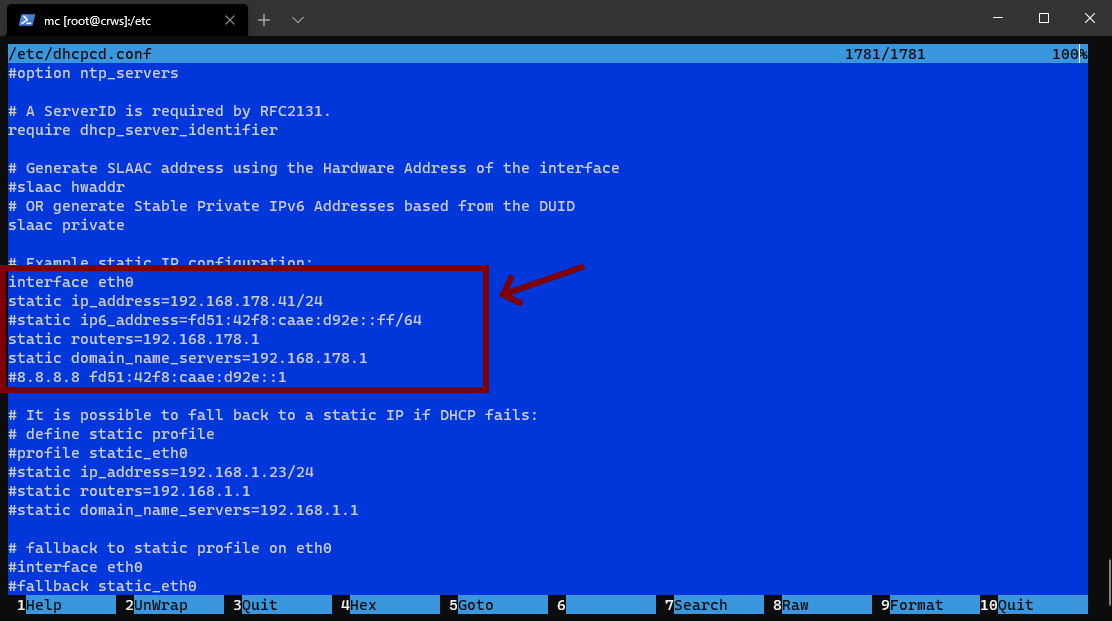
pi@wx:~ $ sudo apt-get install mc -y

Das Profil ist bei Bedarf um alias ll='ls -l' zu erweitern.



### Statische IP

Es wird empfohlen, eine Statische IP-Adresse für den Raspberry unter /etc/dhcpcd.conf zu konfigurieren (siehe [2]).

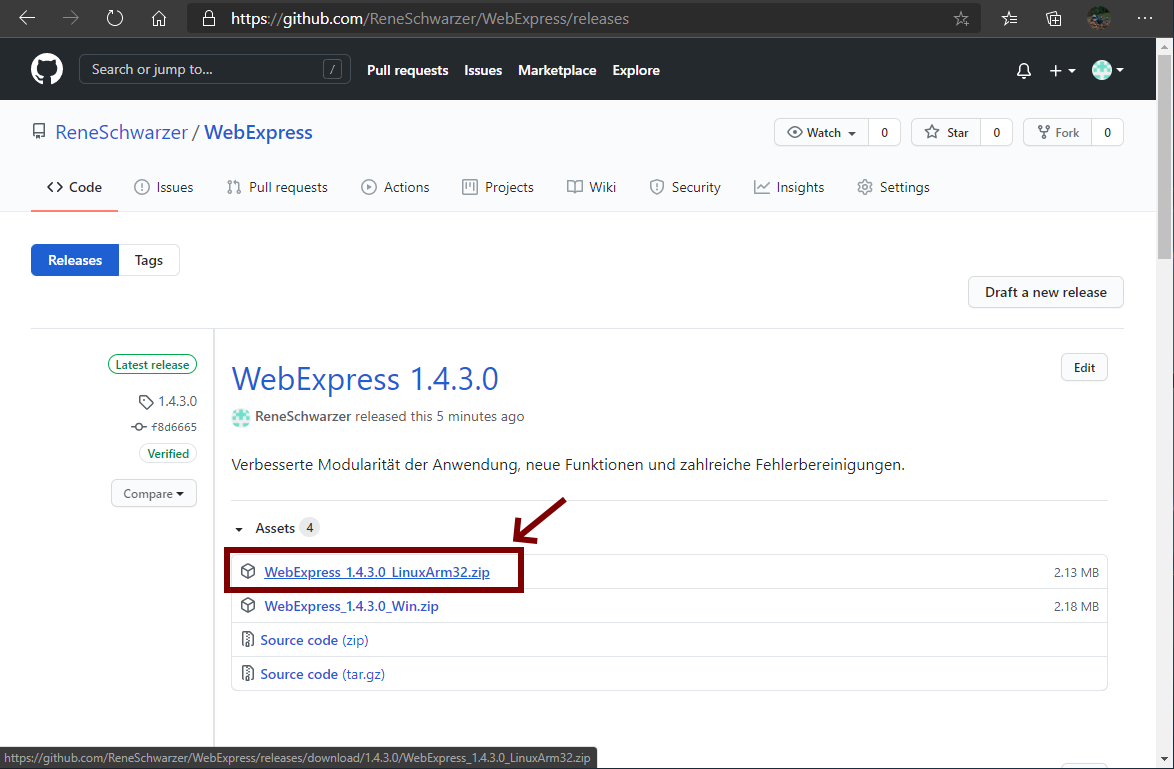


## Installation und Einrichtung WebExpress

WebExpress ist ein schlanker Webserver, welcher in wenigen Schritten einsatzbereit ist.

### WebExpress installieren

WebExpress wird in gepackter Form für den Raspberry Pi im GitHub-Repository https://github.com/ReneSchwarzer/WebExpress/releases kostenlos bereitgestellt.



Die Binaries von WebExpress können mittels wget von GitHub gedownloadet werden.

pi@wx:~ $ wget https://github.com/ReneSchwarzer/WebExpress/releases/download/1.4.3.0/WebExpress\_1.4.3.0\_LiLinuxA32.zip

Als Vorbereitung für die Installation von WebExpress ist ein Verzeichnis unter /opt/wx anzulegen, indem die Binaries entpackt werden.

pi@wx:~ $ sudo mkdir /opt/wx

Anschließend ist das Archiv zu entpacken.

pi@wx:~ $ sudo unzip WebExpress\_1.4.3.0\_LinuxArm32.zip -d /opt/wx

Nachdem WebExpress erfolgreich entpackt wurde, müssen die Ausführungsrechte erteilt werden.

pi@wx:~ $ sudo chmod +x /opt/wx/webexpress.sh /opt/wx/WebExpress.App

Zum automatischen Starten der WebExpress-Anwendung ist die mitgelieferte SystemCtl-Unit zu installieren.

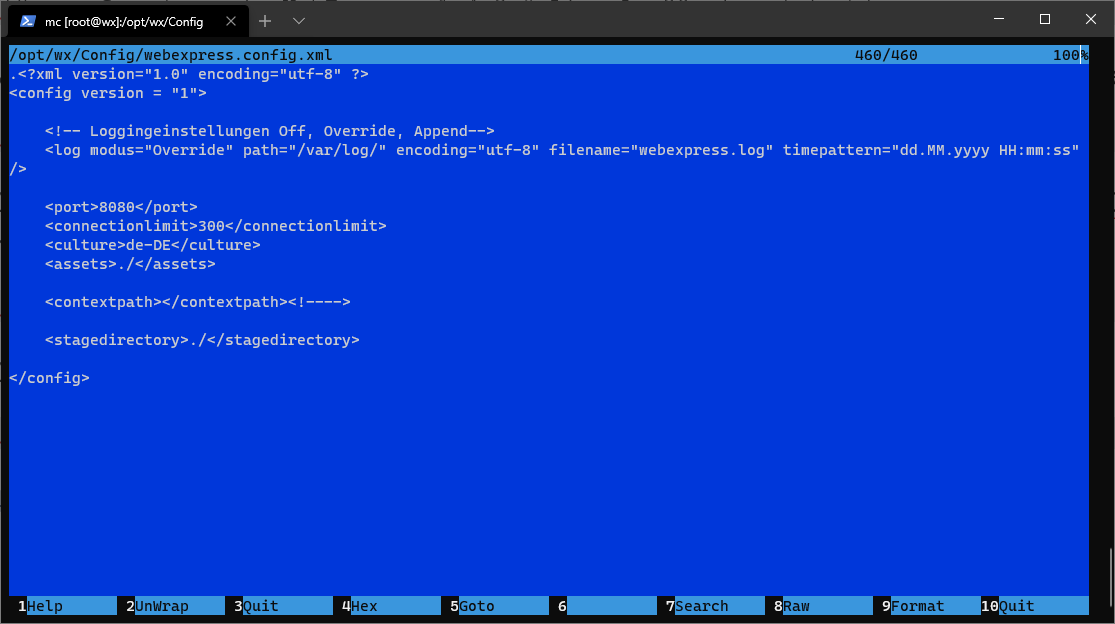
pi@wx:~ $ sudo cp /opt/wx/webexpress.service /etc/systemd/system

Zum Abschluss muss die SystemCtl-Unit aktiviert werden.

pi@wx:~ $ sudo systemctl enable webexpress.service

### WebExpress einrichten

In der Konfigurationsdatei /opt/wx/webexpress.config.xml werden die allgemeinen Einstellungen von WebExpress abgelegt.



Nach einer Änderung der Konfiguration ist WebExpress neu zu starten.

pi@wx:~ $ sudo systemctl restart webexpress

### Weitere WebExpress-Anwendungen installieren

WebExpress verfügt über ein leistungsstarkes Plugin-System. Die zu installierenden Plugins und gegebenen falls Abhängigkeiten werden in das Verzeichnis /opt/wx kopiert. Eventuell muss das Plugin konfiguriert werden. Für die Installation und Einrichtung der Plugins sind die Anleitungen der Plugins heranzuziehen. Die Plugins werden erst aktiv, wenn WebExpress neu gestartet wurde.

pi@wx:~ $ sudo systemctl restart webexpress

# OpenSSL CA erstellen und Zertifikate ausstellen

Eine sichere und vertrauliche Kommunikation zwischen dem WebClient und den Webserver kann durch Nutzung von Zertifikaten gewährleistet werden. Im einfachsten Fall können diese Zertifikate selbst ausgestellt werden und diese im WebClient installiert werden. Weiteren Informationen sind unter [3] zu finden.

## Certificate Authority (CA) erstellen

Zum Ausstellen von Zertifikaten muss zunächst eine Certificate Authority (CA)-Stelle eingerichtet werden.

### Privater Schlüssel

Als erstes muss ein geheimer privater Schlüssel der CA erstellt werden. Dieser wird den Namen caKey.pem erhalten.

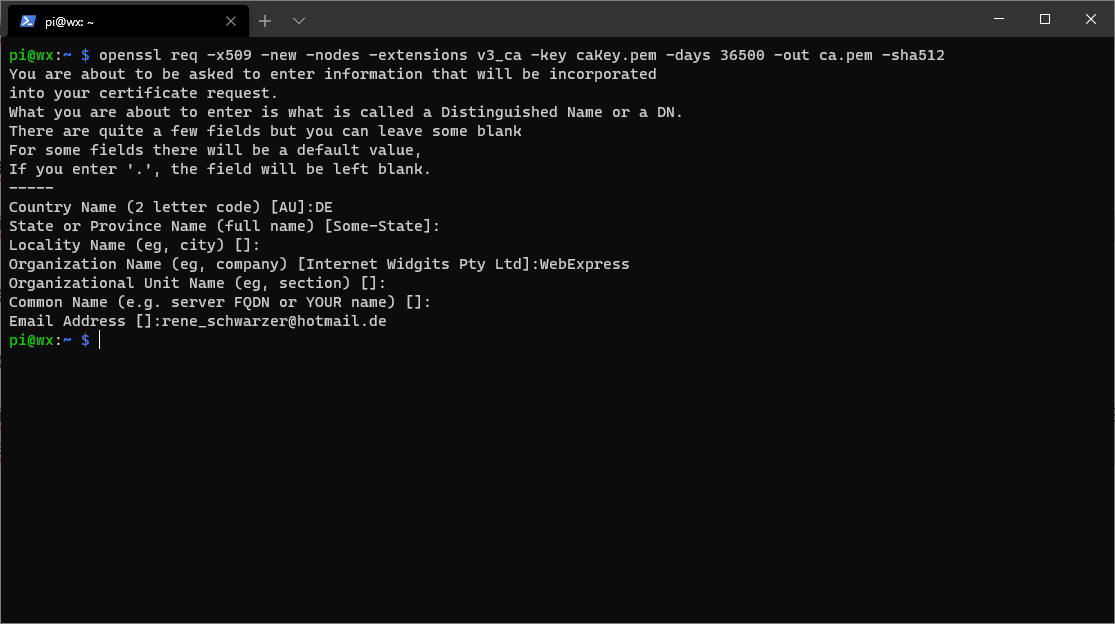
pi@wx:~ $ openssl genrsa -out caKey.pem 4096

### Root-Zertifikat erstellen

Im zweiten Schritt ist das Root-Zertifikat zu erstellen, welches den Namen ca.pem erhält.

pi@wx:~ $ openssl req -x509 -new -nodes -extensions v3\_ca -key caKey.pem -days 36500 -out ca.pem -sha512

Während des Erstellungsprozesses werden verschiedene Daten abgefragt und im Root-Zertifikat gespeichert.



Ist dieser Schritt abgeschlossen, so ist die CA fertig und einsatzbereit.

## Zertifikat ausstellen

Nachdem die CA erstellt wurde, können mit deren Hilfe Zertifikate ausgestellt werden.

### Privater Schlüssel

Für jedes Zertifikat ist ein neuer geheimer privater Schlüssel anzulegen. Für WebExpress wird ein geheimer privater Schlüssel mit den Namen wxKey.pem erstellt.

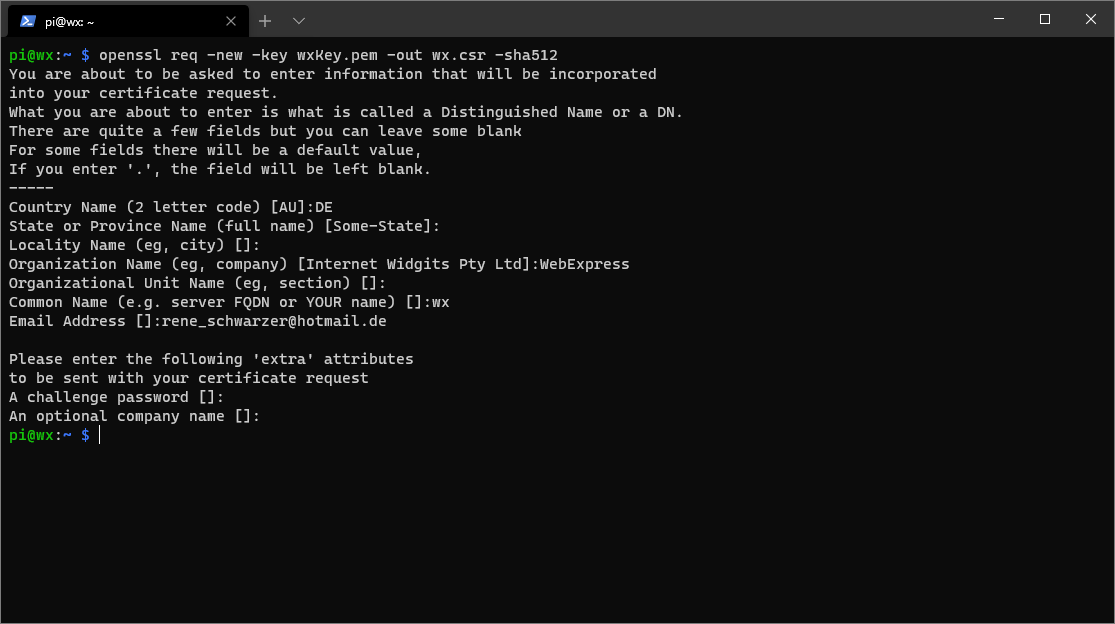
pi@wx:~ $ openssl genrsa -out wxKey.pem 4096

### Zertifikatsanfrage erstellen

Zum Ausstellen eines Zertifikates muss eine Zertifikatsanfrage an die CA gestellt werden. Die Anfrage wird unter den Namen wx.csr gespeichert.

pi@wx:~ $ openssl req -new -key wxKey.pem -out wx.csr -sha512

Im Erstellungsprozess werde verschiedene Daten abgefragt und in der Anfrage gespeichert.



### Zertifikat ausstellen

Nachdem die Zertifikatsanfrage erstellt wurde, kann diese durch die CA verarbeitet werden. Das Zertifikat erhällt den Namen wx.pem.

pi@wx:~ $ pi@wx:~ $ openssl x509 -req -in wx.csr -CA ca.pem -CAkey caKey.pem -CAcreateserial -out wx.pem -days 36500 -sha512

## Zertifikate installieren

Nachdem die Zertifikate erstellt wurden, müssen diese dem Reverse-Proxy (siehe Abschnitt 6) und dem WebClient bekannt gegeben werden.

### Zertifikate in WebExpress installieren

Das CA-Root- (ca.pem), das private (wxKey.pem) und öffentliche (wx.pem) Zertifikat für WebExpress müssen in das Verzeichnis /opt/wx/ssl kopiert werden.Hierfür muss zunächst das Verzeichnis jedoch erstellt werden.

pi@wx:~ $ sudo mkdir /opt/wx/ssl

Anschließend sind die notwendigen Zertifikate nach /opt/wx/ssl zu kopieren.

pi@wx:~ $ sudo cp ca.pem wx.pem wxKey.pem /opt/wx/ssl

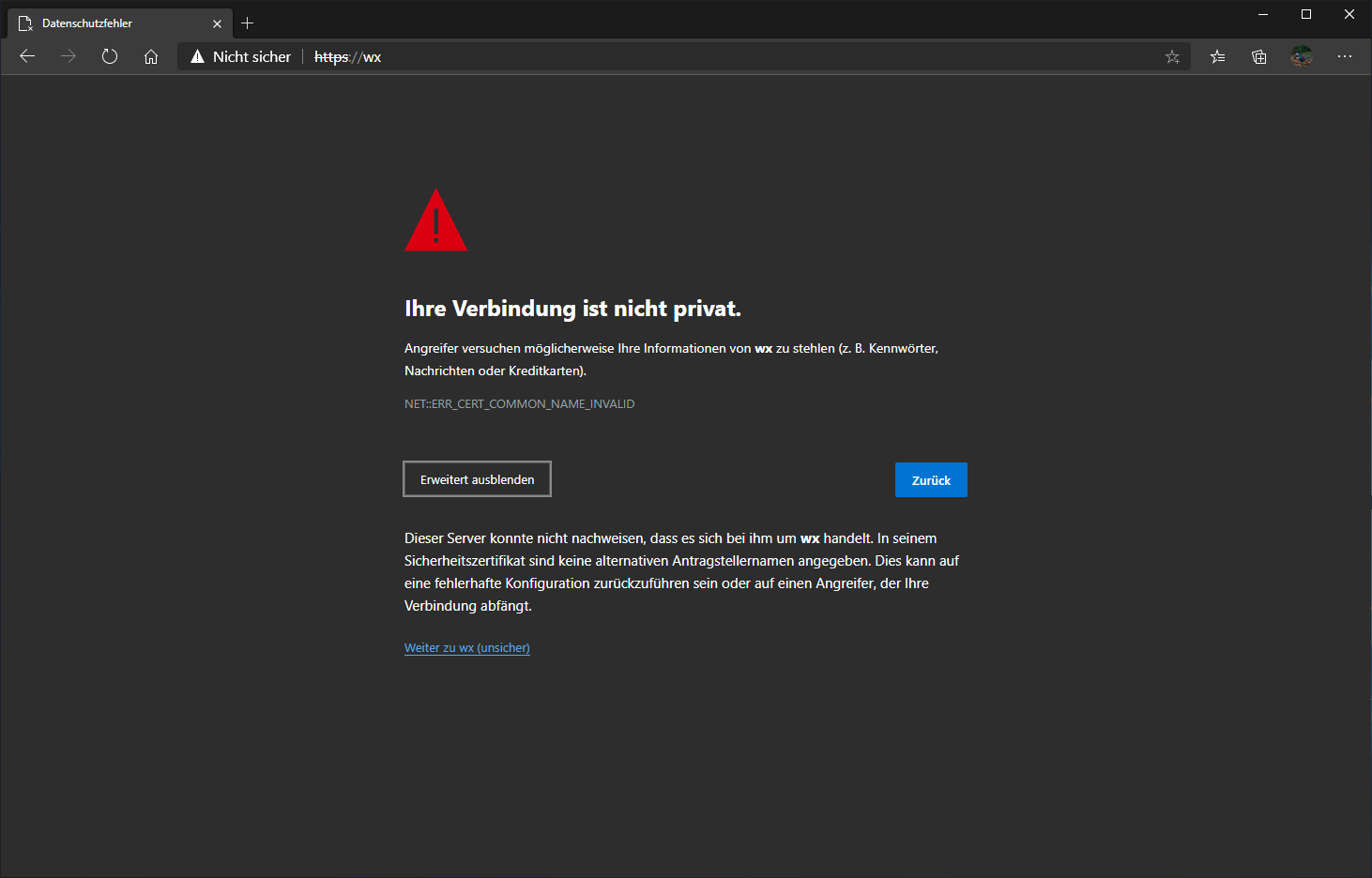
### Zertifikate im Client installieren

Das CA-Root-Zertifikat ist im Zertifikatsspeicher unter vertrauenswürdige Stammzertifizierungsstellen abzulegen.

|  |  |
| --- | --- |
|  |  |
|  |  |
|  |  |

### WebExpress-Zertifikat vertrauen

Im Browser ist das WebExpress-Zertifikat zu vertrauen.



# Installation und Einrichtung eines Reverse Proxys

WebExpress ist nicht in der Lage verschlüsselte Verbindungen über https zu verarbeiten. Hierfür ist ein Reverse-Proxy erforderlich.



## Apache Webserver installieren

Als Reverse-Proxy kann Apache genutzt werden. Dieser steht als Open-Source und kostenlos zur Verfügung.

pi@wx:~ $ sudo apt install apache2 -y

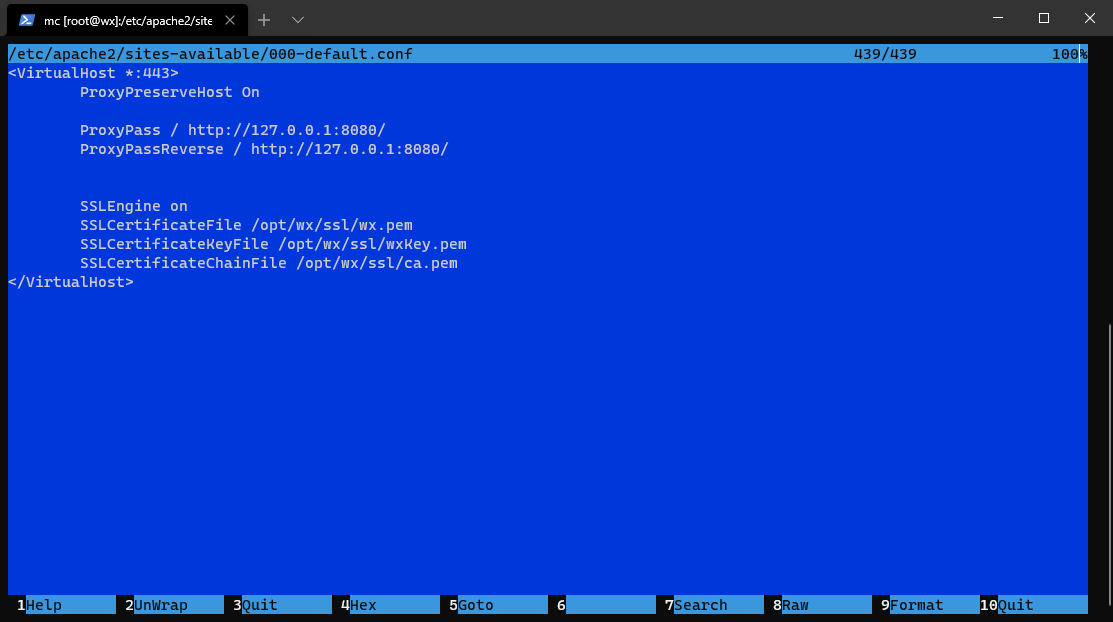
## Apache Module installieren

Für die Verwendung des Apache-Webserver als Reverse-Proxy müssen weitere Module installiert werden.

pi@wx:~ $ sudo a2enmod proxy proxy\_http ssl

## Apache konfigurieren

Unter /etc/apache2/sites-available/000-default.conf sind die Proxy-Weiterleitungen zu hinterlegen.



In der Webserver-Konfiguration müssen die Zertifikatsdateien angegeben werden:

* Privater Key des Zertifikats (wxKey.pem)
* Öffentlicher Key des Zertifikats (wx.pem)
* Öffentlicher Key der CA (ca.pem)

Abschließend muss der Apache-Webserver neu gestartet werden.

pi@webexpress:~ $ sudo systemctl restart apache2

# Update

Um die Sicherheit zu gewährleisten ist der Raspberry Pi, deren Anwendungen und WebExpress regelmäßig zu aktualisieren.

pi@vila:~ $ sudo raspi-config

pi@vila:~ $ sudo apt-get update

pi@vila:~ $ sudo apt-get upgrade

Die WebExpress-Binaries sind ebenfalls zu aktualisieren. Hierzu sind die aktuellen Binaries aus https://github.com/ReneSchwarzer/WebExpress/releases zu verwenden (siehe Abschnitt 4.2).

# Einkaufsliste

Als Hardware werden benötigt:

* Ein Raspberry Pi 4 Modell B mit 8GB
* Ein Steckernetzteil 5V/3A USB Typ-C
* Eine 16GB oder 32GB MicroSD Karte
* Optional ein Gehäuse

# Quellen

[1] <https://dotnet.microsoft.com/download/linux-package-manager/debian10/runtime-current>

[2] <https://www.ionos.de/digitalguide/server/konfiguration/raspberry-pi-mit-fester-ip-adresse-versehen/#:~:text=Den%20Raspberry%20Pi%20mit%20einer%20festen%20IP-Adresse%20ausstatten.,Zeitraum%20mit%20anderen%20Ger%C3%A4ten%20auf%20ihn%20zugreifen%20will>.

[3] <https://legacy.thomas-leister.de/eine-eigene-openssl-ca-erstellen-und-zertifikate-ausstellen/>