

# Mailserver aufsetzen - Praktisches Handbuch

Dimitri Graf und Timo Furrer

April 2014

# Inhaltsverzeichnis

<b>1</b>	<b>Einleitung</b>	<b>2</b>
<b>2</b>	<b>Raspberry Pi</b>	<b>2</b>
2.1	Hardware . . . . .	2
2.2	Betriebssystem . . . . .	3
<b>3</b>	<b>Was man braucht</b>	<b>4</b>
3.1	Betriebssystem installieren . . . . .	5
<b>4</b>	<b>Vorkonfiguratio</b>	<b>5</b>
4.1	Betriebssystem starten . . . . .	5
4.2	Root-Rechte erlangen . . . . .	6
4.3	System aktualisieren . . . . .	7
<b>5</b>	<b>Mailserver aufsetzen</b>	<b>7</b>
5.1	Statische IP-Adresse . . . . .	7
5.2	Voraussetzung einer Domain . . . . .	9
5.3	Komponenten installieren . . . . .	9
5.3.1	MySQL . . . . .	10
5.4	Postfix Administrationstool . . . . .	10
5.4.1	Postfix . . . . .	11
5.4.2	Dovecot . . . . .	14
5.5	Erste E-Mail Adresse anlegen . . . . .	15

# 1 Einleitung

Dieses Handbuch zeigt Schritt für Schritt auf, wie ein Mailserver auf einem Computer installiert und eingerichtet wird. Die diversen Sicherheitsaspekte wie Verschlüsselung der Kommunikation, Verschlüsselung des Speicherortes und SPAM-Schutz sind dabei von zentraler Bedeutung. Dieses Tutorial setzt gewisse Grundkenntnisse voraus:

- Funktionsweise des E-Mailverkehrs (Protokolle, Stationen etc.)
- Solide Linux-Kenntnisse
- Wissen, wie Informationen aus dem Internet beschafft werden können

Für die Erstellung dieses Tutorials wurde ein Raspberry Pi verwendet. Dies empfiehlt sich für ein erstes Ausprobieren. Fühlt man sich aber erst einmal wohl und hat die einzelnen Schritte verinnert, empfiehlt es sich, auf performantere Hardware zu setzen.

## 2 Raspberry Pi

Ein Raspberry Pi ist ein Einplatinencomputer in der Grösse einer Kreditkarte. Er wurde 2009 von der Raspberry Pi Foundation entwickelt, um Schulen einen kostengünstigen Computer zur Verfügung zu stellen. Dabei ging es vor allem um das Erlernen von Computerwissenschaften, wie Programmierung von Software oder Ansteuerung einfacher Hardware, wie LEDs, Displays und einfachen Motoren.<sup>1 2</sup> Der Raspberry Pi ist sehr vielseitig einsetzbar und nicht zuletzt deswegen hat sich eine grosse und begeisterte Community gebildet. Als Betriebssystem kommt auf Grund der leistungsfähig begrenzten Hardware meistens eine Linux-basierte Distribution zum Einsatz. Linux arbeitet sehr ressourcenschonend und die Distributionen sind zudem oft speziell auf den Rechner und das jeweilige Einsatzgebiet angepasst. Der Raspberry Pi ist ein Spielzeug für technikbegeisterte Menschen - jung und alt gleichermassen. Die Community ist ein essentieller Bestandteil des Projektes, denn sie erfinden ständig neue Einsatzmöglichkeiten, kreieren interessante Produkte und Projekte und stellen Anleitungen ins Netz. Um dann selbst loszulegen, braucht man oft nicht mehr als den Raspberry Pi selbst, Monitor, Tastatur und Maus und ein wenig Know-How.

### 2.1 Hardware

Es ist wichtig, für die Inbetriebnahme des Raspberry Pis ein wenig über die Hardware Bescheid zu wissen. Es gibt zur Zeit zwei Modelle des Mini-Rechners: das ursprüngliche Model A und das erweiterte Model B. Model B ist performanter und verfügt über mehr Anschlüsse. Aus diesem Grund ist es Model A in den meisten Fällen vorzuziehen.

Raspberry Pi, Model B verfügt über folgende Anschlüsse:

- Kartenleser für SD/MMC/SDIO (Hauptspeicher, Betriebssystem)

---

<sup>1</sup><http://www.raspberrypi.org/faqs#introWhatIs>

<sup>2</sup><http://www.raspberrypi.org/about>

- 2x USB 2.0
- FBAS (Videoausgabe)
- HDMI (Video & Audio)
- Klinkenstecker, 3.5mm (Audio)
- 10/100-MBit Ethernet Controller (RJ45, Netzwerk)
- 17 GPIO Pins (Externe Hardware)
- 5V Micro-USB (Stromanschluss) <sup>3</sup>

## RASPBERRY PI MODEL B

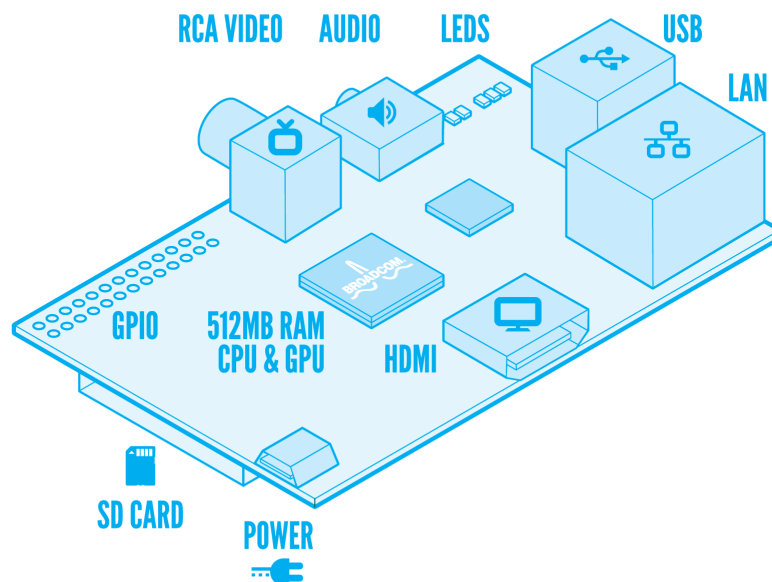


Abbildung 1: Raspberry Pi, Model B

## 2.2 Betriebssystem

Wie bereits erwähnt, wird in diesem Versuch eine Linux-basierte Distribution eingesetzt, die speziell auf den Raspberry Pi zugeschnitten wurde. Raspbian basiert auf der Linux-Distribution Debian und ist neben den Grundfunktionalitäten mit vielen weiteren Programmen ausgestattet, die *out of the box* genutzt werden können. <sup>4</sup> Wie die meisten anderen Linux-basierten Distributionen, kann Raspbian gratis aus dem Internet heruntergeladen

<sup>3</sup>[http://de.wikipedia.org/wiki/Raspberry\\_Pi#Spezifikationen](http://de.wikipedia.org/wiki/Raspberry_Pi#Spezifikationen)

<sup>4</sup><http://www.raspbian.org/>

und gebraucht werden. Natürlich ist es nicht das einzige Linux-Betriebssystem, welches auf dem Raspberry Pi eingesetzt werden kann. Eine Liste bekannter Distributionen kann unter <http://www.raspberrypi.org/downloads> gefunden werden.

### 3 Was man braucht

Damit ein Mailserver auf dem Raspberry Pi installiert und verwendet werden kann, benötigt man folgendes:

- Raspberry Pi, Model B
- Micro-USB Kabel für Stromversorgung
- SD Speicherkarte
- RJ45 Netzkabel

Es gibt gewisse Voraussetzungen, die man bei der Auswahl der Komponenten beachten muss. Das Micro-USB Kabel für die Stromversorgung muss folgende Spezifikationen erfüllen: mind. 700mA, 5V. Die SD-Karte, auf die später das Betriebssystem geladen wird, sollte mindestens eine Kapazität von 4 Gigabyte aufweisen. Zudem muss man beachten, dass nicht jede verfügbare Karte auch tatsächlich unter dem verwendeten Betriebssystem funktioniert. Es gibt im Internet eine Liste, in der eine Reihe kompatibler und nicht kompatibler Karten aufgeführt werden. Es lohnt sich, diese zu Rate zu ziehen. Man findet sie unter: [http://elinux.org/RPi\\_SD\\_cards#Working\\_.2F\\_Non-working\\_SD\\_cards](http://elinux.org/RPi_SD_cards#Working_.2F_Non-working_SD_cards). Das Netzkabel wird verwendet, um den Raspberry Pi mit dem Internet zu verbinden. Dies kann beispielsweise über einen Switch, Router oder eine RJ45-Dose geschehen.

Für die Einrichtung selbst wird noch zusätzliche Peripherie gebraucht, die aber in den meisten Haushalten schon vorhanden sein sollte:

- Computer oder Notebook
- SD-Karten-Leser (oft in Notebooks integriert)
- Monitor mit HDMI-Anschluss
- HDMI-Kabel
- Tastatur
- Maus

Die aufgelistete Peripherie braucht man nur für die Einrichtung selbst oder anfallende Wartungsarbeiten. Ist alles erst einmal korrekt eingerichtet, kann diese wieder andersweitig verwendet werden.

Detaillierte Informationen und Tipps zu den Komponenten sowie der Peripherie findet man unter: <http://www.raspberrypi.org/phpBB3/viewtopic.php?t=4277>

### 3.1 Betriebssystem installieren

Bevor der Computer gestartet und mit der Installation begonnen werden kann, muss noch das Betriebssystem aufgesetzt werden. Die SD-Speicherkarte kann mit wenigen Anweisungen von jedem gängigen Betriebssystem (Windows, OSX, Linux) aus mit einem Raspbian bestückt werden.

Zuerst gilt es, das Betriebssystem selbst aus dem Internet zu laden. Die Download-Datei findet man unter: <http://www.raspberrypi.org/downloads>. Ist die Datei vollständig heruntergeladen und die SD-Karte am SD-Karten-Leser angeschlossen, kann das heruntergeladene Image (Datei) auf die Karte geladen werden. Es gibt eine sehr gute Anleitung im Internet, die die einzelnen Schritte in Details beschreibt. Man findet sie unter: [http://elinux.org/RPi\\_Easy\\_SD\\_Card\\_Setup](http://elinux.org/RPi_Easy_SD_Card_Setup).

## 4 Vorkonfiguratio

Nach erfolgreichem des Betriebssystems auf der SD-Karte folgt die Vorkonfiguration dessen. Damit der Mailserver korrekt eingerichtet werden kann, müssen zuerst gewisse Grundeinstellungen vorgenommen und Komponenten aktualisiert werden.

### 4.1 Betriebssystem starten

Beim ersten Start des Raspberry öffnet sich nach kurzer Wartezeit ein graues Fenster auf blauem Hintergrund mit dem Titel *Raspberry Pi Software Configuration Tool*.

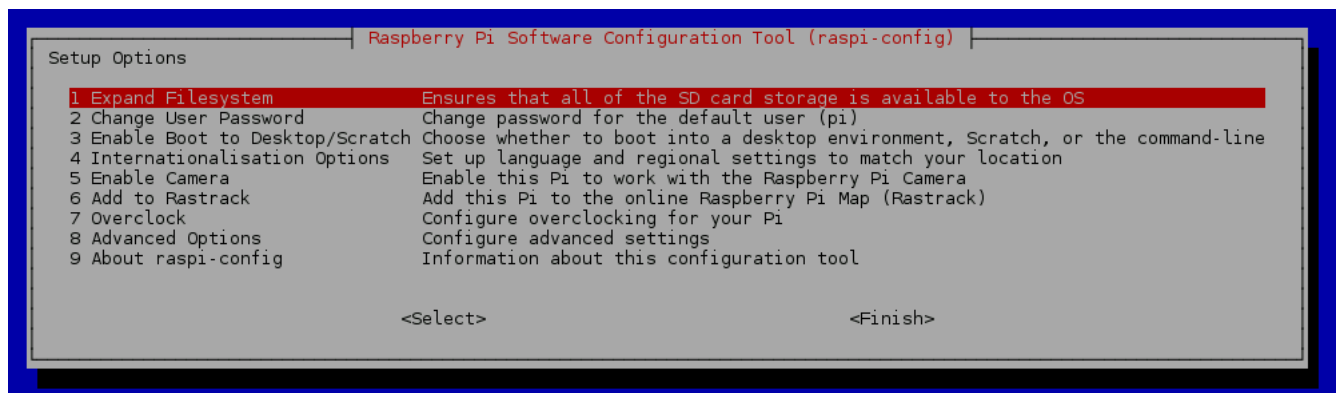


Abbildung 2: Raspberry Pi Konfigurations-Tool

Dieses Werkzeug dient dazu, gewisse Grundeinstellungen vorzunehmen. Folgende Einstellungen sollten vorgenommen werden:

1. Benutzerpasswort ändern (Change User Password)
2. Ausweiten des Dateisystems, um den Speicher der Karte voll auszunutzen (Expand Filesystem)

3. Anpassen des Keyboardlayouts (Internationalisation Options > Change Keyboard Layout)
4. Regionaleinstellungen auf en\_US.UTF-8 setzen (Internationalisation Options > Change Locale)
5. Zeitregion setzen (Internationalisation Options > Change Timezone)
6. Raspberry Pi übertakten auf "Medium, 900MHz"(Overclock)
7. Speicherverteilung für die GPU auf 16 (Megabyte, MB) begrenzen (Advanced Options > A3 Memory Split)

Sollte es mit den getroffenen Einstellungen zu Problemen kommen, kann das Konfigurationstool im laufenden Betrieb erneut aufgerufen werden. Folgender Befehl muss dazu in der Konsole eingegeben werden:

```
sudo raspi-config
```

Sind alle Einstellungen vorgenommen, kann das Menü mittels *Finish* verlassen werden. Die Frage, ob neu gestartet werden soll (reboot), mit "Ja"beantworten, worauf das System neu startet und alle zuvor vorgenommenen Einstellungen übernommen werden.

Nach dem Neustart findet man sich in einem konsolenartigen Fenster mit einem blinkenden Cursor wieder. Dies wird für den Rest des Tutorials die Arbeitsumgebung sein, da die grafische Benutzeroberfläche nicht gebraucht wird. Die Performance ist in der Konsole zudem deutlich besser.

## 4.2 Root-Rechte erlangen

Die meisten der in dieser Anleitung beschriebenen Befehle verlangen erweiterte Rechte. Diese können unter Raspbian ganz einfach erlangt werden mittels:

```
sudo su
```

Grosse Macht bringt auch grosse Verantwortung. Hat man unter Linux Administrator-Rechte (auch Root-Rechte genannt), kann man sehr schnell sehr vieles kaputt machen. Im schlimmsten Fall muss die SD-Karte neu aufgesetzt werden. Es lohnt sich deshalb, ein paar wenige Regeln zum Gebrauch der Konsole zu beachten:

- Ein Befehl wird mittels Drücken der *Enter-Taste* abgesetzt
- Bevor ein neuer Befehl abgesetzt werden, muss der zuvor eingegebene abgeschlossen sein (manchmal ist Geduld gefragt)
- Gross- und Kleinschreibung werden unter Linux unterschieden!
- Immer sicherstellen, dass der Befehl auch wirklich richtig eingegeben wurde

Als Root-User sollte man nur dann unterwegs sein, wenn man die erweiterten Rechte für einen längeren Zeitraum benötigt, wie das in diesem Tutorial der Fall ist. Für einzelne Befehle kann auch *sudo* verwendet werden. Dieses Schlüsselwort wird einfach jedem Befehl vorangestellt, der erweiterte Rechte verlangt.

Ein Beispiel:

```
sudo apt-get update
```

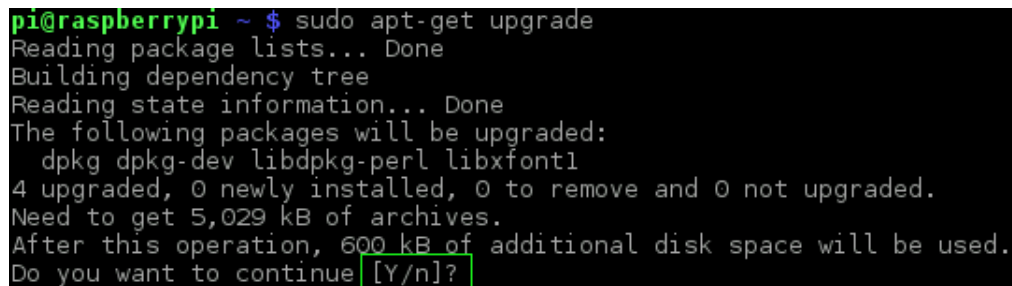
Am Ende dieses Tutorials sollten die Root-Rechte unbedingt wieder abgegeben werden. Um das zu tun, reicht ein einfaches *exit* in der Kommandozeile und schon ist man wieder als normaler User unterwegs.

## 4.3 System aktualisieren

Um sicherzustellen, dass das System auf dem aktuellsten Stand ist, müssen zuerst alle installierten Pakete aktualisiert werden. Dazu müssen folgende zwei Befehle abgesetzt werden:

```
apt-get update  
apt-get upgrade
```

Nach der Eingabe von *apt-get upgrade* fragt das Terminal noch einmal nach, ob man die zur Verfügung stehenden Pakete wirklich installieren will. Standardmässig ist die Antwort auf *Ja* eingestellt, was man an dem grossen Y in *[Y/n]* erkennt. Um fortzufahren reicht ein erneutes Drücken der Enter-Taste. Zukünftige Rückfragen bei abgesetzten Befehlen können auf die gleiche Weise behandelt werden. Es empfiehlt sich dennoch, die angezeigten Meldungen (Informationen, Warnungen) immer durchzulesen und entsprechend zu handeln.



```
pi@raspberrypi ~ $ sudo apt-get upgrade  
Reading package lists... Done  
Building dependency tree  
Reading state information... Done  
The following packages will be upgraded:  
  dpkg dpkg-dev libdpkg-perl libxfont1  
4 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.  
Need to get 5,029 kB of archives.  
After this operation, 600 kB of additional disk space will be used.  
Do you want to continue [Y/n]?
```

Abbildung 3: apt-get upgrade

## 5 Mailserver aufsetzen

### 5.1 Statische IP-Adresse

Die IP-Adresse, durch welche der Computer im Netz eindeutig identifiziert wird, sieht möglicherweise nach jedem Computerstart anders aus. Damit der Router im Heimnetzwerk später weiss, wohin er die einkommenden Mails weiterleiten muss, sollte die IP-Adresse statisch festgelegt werden.



Zuerst muss die Netzwerkkonfigurationsdatei unter *etc/network/interfaces* angepasst werden. Dazu muss man die Datei mit einem Texteditor öffnen. Dazu kann der vorinstallierte Texteditor *Nano* verwendet werden:

```
sudo nano /etc/network/interfaces
```

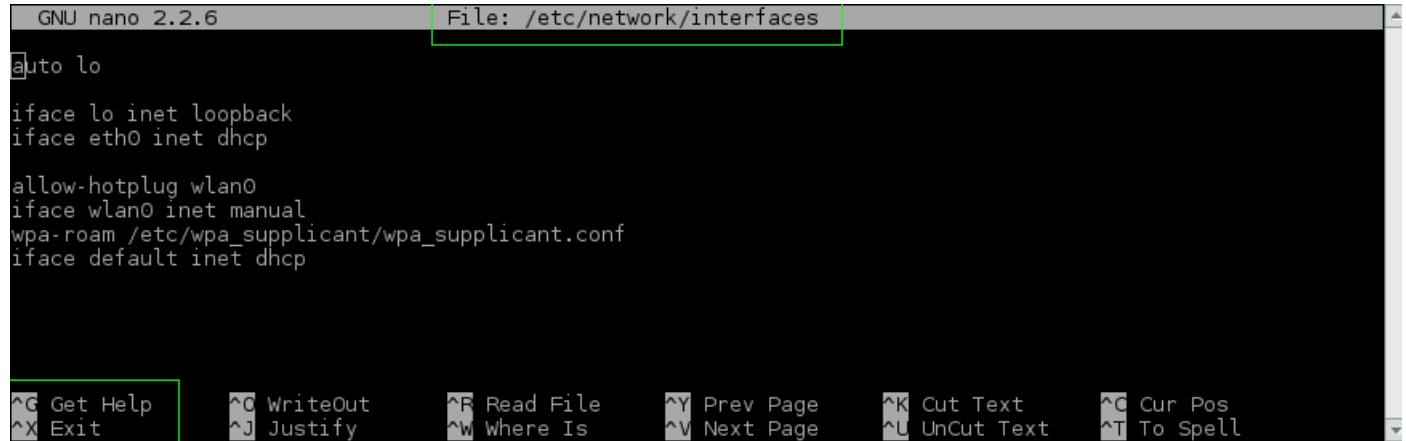


Abbildung 4: Texteditor Nano

Unter Umständen kann es vorkommen, dass folgende Zeile bereits in der Datei steht:

```
iface eth0 inet dhcp
```

Sofern dies der Fall ist, muss diese gelöscht und durch folgende Zeilen ersetzt werden:

```
iface eth0 inet static
address 192.168.1.107
netmask 255.255.255.0
gateway 192.168.1.1
network 192.168.1.0
broadcast 192.168.1.255
```

Die verwendete IP-Adresse in der zweiten Zeile (192.168.1.107) sollte durch die aktuelle IP-Adresse des Systems ersetzt werden. Diese kann mit dem Kommando *ifconfig* ermittelt werden. Sie steht auf der zweiten Zeile bei *eth0* unter *inet addr:*.

```

pi@raspberrypi ~ $ ifconfig
eth0      Link encap:Ethernet  HWaddr b8:27:eb:dc:9a:76
          inet addr:192.168.1.107  Bcast:192.168.1.255  Mask:255.255.255.0
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:9311 errors:0 dropped:0 overruns:0 frame:0
          TX packets:3672 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:13085231 (12.4 MiB)  TX bytes:315538 (308.1 KiB)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:0 (0.0 B)  TX bytes:0 (0.0 B)

```

Abbildung 5: Ausgabe von ifconfig

## 5.2 Voraussetzung einer Domain

Damit ein Mailserver immer erreichbar ist, braucht es auf irgendeinem Domain Name Server einen Eintrag. Dies bedeutet, dass die Verbindung von unserem lokalen Router zu einem beliebigen anderen Computer im Internet gewährleistet ist. ... to be continued

## 5.3 Komponenten installieren

Als nächstes müssen alle benötigten Komponenten installiert werden. Diese sind:

- MySQL Server
- Postfix
- Dovecot

Dovecot verlangt ein aktives IPv6-Modul. Dieses muss zur entsprechenden Liste hinzugefügt und anschliessend aktiviert werden:

```

echo "ipv6" >> /etc/modules
modprobe ipv6

```

Nun können die Pakete installiert werden:

```

apt-get install postfix postfix-mysql dovecot-common dovecot-imapd mysql-
server dovecot-pop3d openssl

```

Nachfolgend werden der Reihe nach für MySQL und Postfix gewisse Grundeinstellungen vorgenommen. Die Programme führen den Benutzer dabei an der Hand und gehen Schritt für Schritt die Fragen durch.

Anschliessend werden alle Pakete installiert, was ein paar Minuten dauern kann.

### 5.3.1 MySQL

Zuerst muss die MySQL-Datenbank mit dem Namen "postfix" erstellt werden. Dies geschieht mit dem folgenden Befehl:

```
mysqladmin -p create postfix
```

Nun wird man nach dem Passwort gefragt. Es handelt sich dabei um jenes, welches während der Installation von MySQL gewählt wurde.

Anschließend kann man sich in die Datenbank einloggen. Es wird erneut nach dem gleichen Passwort gefragt:

```
mysql -p postfix
```

Ein Benutzer für die Datenbank postfix wird jetzt angelegt:

```
CREATE USER 'postfix'@'localhost' IDENTIFIED BY 'SOMEPASSWORD';  
GRANT ALL PRIVILEGES ON postfix.* TO 'postfix'@'localhost' IDENTIFIED BY '  
    SOMEPASSWORD';  
FLUSH PRIVILEGES;
```

## 5.4 Postfix Administrationstool

Damit wir keine manuellen Anpassungen an der Datenbank vornehmen müssen, wie z.B. Tabellen, Benutzer, E-Mail-Adressen und so weiter erstellen, installieren wir uns *Postfix admin* ein Administrationstool um Postfix zu konfigurieren.

Dieses Tool ist leider nicht über unsere Paketverwaltung verfügbar, weshalb wir es aus dem Internet herunterladen und selbst in unser Webserververzeichnis */var/www* verschieben müssen:

```
cd /var/www  
wget --content-disposition http://sourceforge.net/projects/postfixadmin/files/  
    latest/download?source=files  
tar xfvz postfixadmin-*.tar.gz  
mv postfixadmin*/ postfixadmin  
chown www-data:www-data -R postfixadmin  
cd postfixadmin
```

Damit *postfix admin* eine Verbindung zu unserer Datenbank herstellen kann, müssen wir einige Anpassungen vornehmen. Folgende Werte müssen in der Datei */var/www/postfixadmin/config.inc.php* gesucht und entsprechend angepasst werden:

```
$CONF['configured'] = true;  
$CONF['database_type'] = 'mysql';  
$CONF['database_host'] = 'localhost';  
$CONF['database_user'] = 'postfix';  
$CONF['database_password'] = 'SOMEPASSWORD';  
$CONF['database_name'] = 'postfix';  
$CONF['encrypt'] = 'md5crypt';  
$CONF['domain_path'] = 'YES';  
$CONF['default_language'] = 'de';
```

```
$CONF[ 'domain_in_mailbox' ] = 'NO';  
$CONF[ 'emailcheck_resolve_domain' ] = 'NO';
```

In der selben Datei kommt oft die Beispieldomain *change-this-to-your.domain.tld* vor. Diese muss durch die Domain ersetzt werden, die man bei der Installation von *postfix* angegeben hat.

Nun kann mit dem Setup von *postfix* begonnen werden. Dazu einfach in einem Webbrowser eure Domain wie folgt aufrufen:

```
http://deinedomain.tld/postfixadmin/setup.php
```

Hierbei wird ein Check ausgeführt, wo alle Abhängigkeiten geprüft werden. Steht da überall *OK* kann weitergefahren werden, ansonsten müssen die Abhängigkeiten installiert oder Konfigurationen angepasst werden.

Zusätzlich wird nach einem *setup password* gefragt. Dieses sollte auf jeden Fall gesetzt werden und wird dazu gebraucht um sich bei einem weiteren Setupaufruf identifizieren zu können. Dieses Setuppassword muss anschliessend auch in die Datei */var/www/postfixadmin/config.inc.php* eingetragen werden.

```
$CONF[ 'setup_password' ] = 'GEWAEHLTESSETUPPASSWORT';
```

Nun muss erneut die Setup Seite aufgerufen und ein Administrationkonto angelegt werden. Dieses wird gebraucht, um später E-Mail-Adressen und weiteres anzulegen.

### 5.4.1 Postfix

Da wir nun *postfix admin* installiert und konfiguriert haben, können wir mit der eigentlichen *postfix* Konfiguration beginnen.

Damit nur ein bestimmter Benutzer unseres System für den Mailserver Teil verantwortlich ist, legen wir diesen an. Wir folgenen gewissen Linux-Richtlinien und geben diesem E-Mail Benutzer die ID *5000*:

```
mkdir /var/vmail  
groupadd -g 5000 vmail  
useradd -g vmail -u 5000 vmail -d /var/vmail  
chown vmail:vmail /var/vmail
```

Sicherheitstechnisch ist es zudem sehr wichtig, dass *postfix* eine SSL-Verbindung hat. Dazu erstellen wir ein Zertifikat mit *openssl*

```
mkdir /etc/postfix/sslcrt  
cd /etc/postfix/sslcrt  
openssl req -new -newkey rsa:3072 -nodes -keyout postfix.key -days 730 -x509 -  
out postfix.crt  
chmod go-rwx postfix.key
```

Beim Erstellen des Zertifikates werden einige Informationen benötigt. Vorallem ist hier wichtig, dass bei dem *Command Name*-Frage die Domain angegeben wird, mit dem auch *postfix* konfiguriert wurde.

Weiter geht es mit der Hauptkonfiguration von *postfix*. Wir fügen der Datei */etc/postfix/main.cf* folgendes hinzu:

```

disable_vrfy_command = yes
smtpd_sasl_type=dovecot
smtpd_sasl_path=private/auth_dovecot
smtpd_sasl_auth_enable = yes
smtpd_sasl_authenticated_header = yes
broken_sasl_auth_clients = yes
proxy_read_maps = $local_recipient_maps $mydestination $virtual_alias_maps
                  $virtual_alias_domains $virtual_mailbox_maps $virtual_mailbox_domains
                  $relay_recipient_maps $relay_domains $canonical_maps $sender_canonical_maps
                  $recipient_canonical_maps $relocated_maps $transport_maps $mynetworks
                  $smtpd_sender_login_maps
smtpd_sender_login_maps = proxy:mysql:/etc/postfix/mysql_sender_login_maps.cf
smtpd_sender_restrictions = reject_authenticated_sender_login_mismatch
                           reject_unknown_sender_domain
smtpd_recipient_restrictions = permit_sasl_authenticated
                              permit_mynetworks
                              reject_unauth_destination

# Virtual mailboxes
local_transport = virtual
virtual_alias_maps = proxy:mysql:/etc/postfix/mysql_virtual_alias_maps.cf
virtual_mailbox_base = /var/vmail/
virtual_mailbox_domains = proxy:mysql:/etc/postfix/mysql_virtual_domains_maps.
                           cf
virtual_mailbox_limit = 524288000
virtual_mailbox_maps = proxy:mysql:/etc/postfix/mysql_virtual_mailbox_maps.cf
virtual_minimum_uid = 104
virtual_transport = virtual
virtual_uid_maps = static:5000
virtual_gid_maps = static:5000
virtual_transport = dovecot
dovecot_destination_recipient_limit = 1

```

In dieser Datei ist auch wichtig, dass der Eintrag *mydestination* nur *localhost* enthält:

```
mydestination = localhost
```

Zudem muss man auf die vorher erstellen SSL-Zertifikate verweisen:

```
smtpd_tls_cert_file = /etc/postfix/sslcert/postfix.crt
smtpd_tls_key_file = /etc/postfix/sslcert/postfix.key
```

Eine weitere wichtige Konfigurationsdatei von *postfix* ist */etc/postfix/master.cf*. Folgendes muss dieser Datei hinzugefügt werden:

```
dovecot    unix    -        n        n        -        -        pipe
flags=DRhu user=vmail:vmail argv=/usr/lib/dovecot/deliver -d ${recipient}
```

**Wichtig:** Die Einrückung der zweiten Zeile muss unbedingt eingehalten werden. Dies gilt auch für weitere Anpassungen in dieser Datei.

Für die folgenden beiden Zeilen muss die Raute (#) davor entfernt werden, um sie aktiv zu machen:

```
submission inet n        -        -        -        -        smtpd
-o smtpd_tls_security_level=encrypt
-o smtpd_client_restrictions=permit_sasl_authenticated,reject
smtps      inet  n        -        -        -        -        smtpd
-o smtpd_tls_wrappermode=yes
```

Nicht nur *postfix admin* braucht eine Verbindung zu unseren Datenbank, auch *postfix* selbst braucht eine. Dazu folgende Dateien mit entsprechendem Inhalt erstellen:

```
nano /etc/postfix/mysql_virtual_alias_maps.cf
```

```
hosts = localhost
user = postfix
password = SOMEPASSWORD
dbname = postfix
query = SELECT goto FROM alias WHERE address='%s' AND active = '1'
```

```
nano /etc/postfix/mysql_virtual_mailbox_maps.cf
```

```
hosts = localhost
user = postfix
password = SOMEPASSWORD
dbname = postfix
query = SELECT maildir FROM mailbox WHERE username='%s' AND active = '1'
```

```
nano \etc/postfix/mysql_virtual_domains_maps.cf
```

```
hosts = localhost
user = postfix
password = SOMEPASSWORD
dbname = postfix
query = SELECT domain FROM domain WHERE domain='%s' AND active = '1'
```

```
nano \etc/postfix/mysql_sender_login_maps.cf
```

```
hosts = localhost
user = postfix
password = SOMEPASSWORD
dbname = postfix
query = SELECT username AS allowedUser FROM mailbox WHERE username="%s" AND
       active = 1 UNION SELECT goto FROM alias WHERE address="%s" AND active = 1
```

Damit diese Dateien geschützt sind, sollte man die Zugriffsrechte ändern:

```
chmod o-rwx,g+r /etc/postfix/mysql_*
chgrp postfix /etc/postfix/mysql_*
```

Postfix muss nun neu gestartet werden, damit die vorgenommenen Änderungen übernommen werden:

```
service postfix restart
```

### 5.4.2 Dovecot

Nachdem die Konfiguration von Postfix abgeschlossen ist, kann *dovecot* konfiguriert werden. Als erstes ist es wichtig eine Sicherheitskopie der Originalkonfiguration zu machen, sodass im Fall einer kompletten Verwüstung dieser Abhilfe geschafft werden kann:

```
mv /etc/dovecot/dovecot.conf /etc/dovecot/dovecot.conf.orig
```

Nun kann eine komplett neue Konfigurationsdatei angelegt werden und mit folgendem Inhalt befüllt werden:

```
nano /etc/dovecot/dovecot.conf
```

```
auth_mechanisms = plain login
log_timestamp = "%Y-%m-%d %H:%M:%S "
passdb {
    args = /etc/dovecot/dovecot-mysql.conf
    driver = sql
}
protocols = imap pop3
service auth {
    unix_listener /var/spool/postfix/private/auth_dovecot {
        group = postfix
        mode = 0660
        user = postfix
    }
    unix_listener auth-master {
        mode = 0600
        user = vmail
    }
    user = root
}
listen = *
```

```

ssl_cert = </etc/postfix/sslcert/postfix.crt
ssl_key = </etc/postfix/sslcert/postfix.key
userdb {
    args = /etc/dovecot/dovecot_mysql.conf
    driver = sql
}
protocol pop3 {
    pop3_uidl_format = %08Xu%08Xv
}
protocol lda {
    auth_socket_path = /var/run/dovecot/auth-master
    postmaster_address = postmaster@unseredomain.tld
}

```

In dieser Datei gibt man in der *userdb* Konfiguration eine Datei für die SQL-Verbindung an. Diese muss natürlich nun erstellt und mit folgendem Inhalt befüllt werden:

```

driver = mysql
connect = host=localhost dbname=postfix user=postfix password=SOMEPASSWORD
default_pass_scheme = MD5-CRYPT
password_query = SELECT password FROM mailbox WHERE username = '%u'
user_query = SELECT CONCAT('maildir:/var/vmail/',maildir) AS mail, 5000 AS uid
, 5000 AS gid FROM mailbox WHERE username = '%u'

```

Auch diese Datei sollte wieder mit Zugriffsrechten versehen werden, die es nicht erlauben, dass die Datei von jedem Benutzer gelesen werden kann:

```

chmod o-rwx,g+r /etc/dovecot/dovecot_mysql.conf
chgrp vmail /etc/dovecot/dovecot_mysql.conf

```

Damit die Änderungen korrekt übernommen werden können, starten wir *dovecot* neu:

```

service dovecot restart

```

## 5.5 Erste E-Mail Adresse anlegen

Nachdem die Konfiguration von *postfix* und *dovecot* abgeschlossen ist, kann man mit der Erfassung von E-Mail Adressen beginnen. Nicht umsonst haben wir *postfix admin* installiert. Dies kommt nun zum Einsatz. Surft man auf die Seite <http://unseredomain.tld/postfixadmin> kann man sich dort mit dem Administrationsbenutzer einloggen und mit der Erfassung beginnen.

Bevor man eine E-Mail Adresse erfassen kann, muss man eine Domain konfigurieren. Diese Domain ist jene, auf der wir uns gerade befinden.

Dazu geht man in *postfix admin* auf:

```

Domain Liste -> Neue Domain

```

Und legt jene dort an:



Admin Liste	Domain Liste	Virtual Liste	E-Mail Abruf	Email versenden	Passwort ändern	Sicherung	Log ansehen	Logout
-------------	--------------	---------------	--------------	-----------------	-----------------	-----------	-------------	--------

**Domain hinzufügen**

Domain:

Beschreibung:

Aliase:  -1 = ausschalten | 0 = kein Limit

Mailboxen:  -1 = ausschalten | 0 = kein Limit

Standard-Aliase hinzufügen: ☒

Mail server is backup MX: ☐

Postfix Admin 2.3.6 | Angemeldet als sirtux@tachyon.bugspot.net | Auf Updates überprüfen | Return to tachyon.bugspot.net

Abbildung 6: Postfix admin - Neue Domain

Als nächstes kann die E-Mail Adresse eingerichtet werden. Dazu geht man auf:

Virtual Liste → Mailbox hinzufügen

Admin Liste	Domain Liste	Virtual Liste	E-Mail Abruf	Email versenden	Passwort ändern	Sicherung	Log ansehen	Logout
-------------	--------------	---------------	--------------	-----------------	-----------------	-----------	-------------	--------

Virtual Liste

**Legen Sie eine neue Mailbox für Ihre Domain an.**

Benutzername:  @

Passwort:  Passwort für POP3/IMAP

Passwort (nochmal):

Name:  Voller Name

Aktiv: ☒

Mailbox anlegen: ☒

Postfix Admin 2.3.6 | Angemeldet als sirtux@tachyon.bugspot.net | Auf Updates überprüfen | Return to tachyon.bugspot.net

Abbildung 7: Postfix admin - Neue Mailbox

# Abbildungsverzeichnis

1	Raspberry Pi, Model B . . . . .	3
2	Raspberry Pi Konfigurations-Tool . . . . .	5
3	apt-get upgrade . . . . .	7
4	Texteditor Nano . . . . .	8
5	Ausgabe von ifconfig . . . . .	9
6	Postfix admin - Neue Domain . . . . .	16
7	Postfix admin - Neue Mailbox . . . . .	16