

Webserver mit Ubuntu (Komplettlösung)

Wichtig:

Lesen Sie alles genau durch und befolgen Sie exakt das Vorgehen, welches beschrieben ist. Ansonsten kommt es zu Fehlern, welche weder reproduzierbar noch schnell behebbar sind und sie vergeuden sehr viel Zeit mit Fehlersuche!

Inhalt

1. Einleitung.....	2
2. Vorbereitung.....	2
3. SSH installieren.....	4
4. Apache2 installieren.....	5
5. Ändern des Homeverzeichnisses einer Webseite?	6
6. PHP installieren.....	7
7. MySQL installieren	8
8. phpMyAdmin installieren	11
9. FTP-Server (proftpd) mit User einrichten.....	14
10. Zugriff Aussen mit Portforwarding (DynDNS mit No-IP)	19
11. Autostart: Programm automatisch starten lassen	23
12. SSL-Zertifikat installieren (Zertifikat mit Letsencrypt)	25
13. WordPress installieren	27
14. Webserver-Verzeichnis mit Passwort schützen.....	33
15. Virtuelle Hosts hinzufügen	36
16. Mails mit ext. Host (GMAIL) über Webserver versenden	38
17. Open-Source Log-Analyzer AWstats installieren.....	42
18. Berechtigungen in Ubuntu.....	44
19. TODO: rkhunter.....	45
20. TODO: Monit	45
21. Troubleshoot	46
22. Quellen	49

Document-Roots

Default-Server: /var/www/html

Eigener: /var/www/bbztest.com/html

History

20.05.22: Macher, weitere Anpassungen (Einleitung, virtuelle root)

28.04.22: Macher, Anpassung auf Ubuntu 22.04 und VMware 16.2.3

03.06.21: Macher, Diverse Anpassungen nach Unterricht, Troubleshooting

04.06.20: Macher, Rechte Linux, Troubleshooting allgemein

26.05.20: Macher, Anpassungen DDNS, Portforwarding und SSL-Zertifikat, Virtualhost und Stats

15.05.20: Macher, Verschönerungen, Klarifizierungen und Aktualisierungen

1. Einleitung

Arbeiten sie konzentriert und beachten sie die einzelnen Punkte, die in der Anleitung stehen. Es kann sein, dass sie ansonsten eine komplette Neukonfiguration vornehmen müssen.

Achten Sie ausserdem darauf, dass sie oft Backups (Clone der VM, Snapshot der VM oder Image der SD-Karte des Raspi) machen. Ansonsten kann es sein, dass sie nochmal komplett neu beginnen müssen.

Es kann immer mal wieder sein, dass die beschriebenen Versionen nicht mehr aktuell sind, oder dass Befehle nicht mehr funktionieren. In diesem Fall suchen Sie im Internet nach den neuen Versionen oder Befehlen. Auch sonst ist es so, dass sie sicher nicht der erste mit einem spezifischen Problem sind, sondern das ihre Lösung irgendwo mit einer Suchmaschine gefunden werden kann.

2. Vorbereitung

Installieren Sie Ubuntu mit einer ISO-Datei. Wählen Sie beim Keyboard-Layout «Switzerland» aus. Bei der nächsten Frage wegen Installation und Updates können Sie alles standartmässig eingestellt lassen. Sorgen Sie dafür, dass Sie mit NAT eine Internet-Verbindung nach der Installation haben. Auch beim Installation type können Sie die Default-Auswahl mit «Erase disk...» so eingestellt lassen und dann mit «Continue» bestätigen. Danach die Zeitzone «Zurich» auswählen und die Personalien eingeben. Danach sollte die Installation starten.

Nun müssen wir dafür sorgen, dass alles auf dem aktuellsten Stand ist. Öffnen Sie dazu unten links auf der Linux Benutzeroberfläche die App-Übersicht (9 Punkte im Quadrat). Geben Sie dann in der Suche «Terminal» ein und machen einen Rechtsklick und «Add to favorites». Danach klicken Sie das Terminal an und geben folgenden Befehl ein:

```
sudo apt-get update
```

 und bestätigen Sie mit Y und Enter (längere Wartezeit).

```
sudo apt-get upgrade
```

 und bestätigen Sie mit Y und Enter (längere Wartezeit).

Danach installieren wir die open-vm-tools:

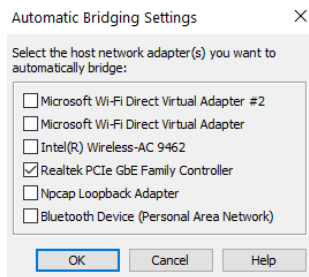
```
sudo apt-get install open-vm-tools
```

Wenn die Anzeige zu klein ist, wechseln Sie in den Fullscreen-Modus, wählen Sie mit einem Rechtsklick auf den Desktop «Anzeige anpassen» und wählen sie bei Skalierung 200% aus und die entsprechende Auflösung.

Dann müssen wir mit dem Terminal die net-tools installieren, damit wir die Standard Netzwerk-Tools an Board haben.

```
sudo apt-get install net-tools
```

Mit `ifconfig` im Terminal finden Sie nun heraus, ob die VM im selben Subnetz ist. Wenn nicht, kann es daran liegen, dass Ihre VM noch NAT ist (<https://bbz.macherit.ch/virtualisierung/#step3>). Das schalten sie in der VM selbst unter Player -> Manage -> Virtual Machine Settings und dort auf den «Network Adapter» von NAT auf Bridged um. Danach die VM neu starten. Dann nochmals im Terminal `ifconfig` eingeben und überprüfen, ob sie nun mit der IPv4-Adresse im selben Netzwerk sind. Hier gibt es immer wieder Probleme mit dem Bridging zum Adapter. Wenn es mit Bridge nicht funktioniert, schalten Sie alle Adapter, ausser Ihrem korrekten Netzwerk-Adapter unter Virtual Maschine Settings -> Bridge «Configure Adapters» aus:



Schalten Sie dann nochmals auf NAT um, dann Neustart, dann nochmals auf Bridged und nochmals Neustart. Sollte es dann immer noch nicht funktionieren, schalten sie wieder auf NAT um und Neustart. Dann werden Sie einfach vom Terminal ausarbeiten und nicht per SSH-Client, wie unten beschrieben.

Nun wird noch der Texteditor "nano" installiert, um direkt im Terminal Textdateien zu bearbeiten:

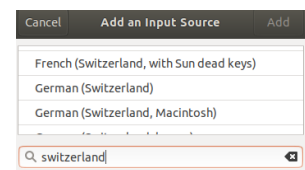
```
sudo apt-get install nano -y
```

Nun sind die Grundvoraussetzungen für die weitere Arbeit mit Ubuntu abgeschlossen.

Troubleshoot:

Wenn das Keyboard noch nicht korrekt ist:

Danach muss das Keyboard gewechselt werden. Gehen Sie wieder ins Applications-Menü und dort auf Settings. Danach «Region & Language» und dort bei den Input Sources auf «+». Danach geben Sie Switzerland ein und gehen auf «Other» und dort wählen Sie German (Switzerland) aus. Entfernen Sie dann «Englisch» mit anklicken und dann «-».



Um die Zeit korrekt einzustellen, geben Sie im Terminal `sudo su` ein und ihr root-Passwort. Danach folgenden Befehl:

```
dpkg-reconfigure tzdata
```

Danach Europa und dann Zürich auswählen. Dann sollte die korrekte Zeit angezeigt werden.

3. SSH installieren

SSH wird mit folgendem Command im Terminal auf Linux installiert:

```
sudo apt-get install openssh-server -y
```

Sobald die Installation beendet ist, ist SSH auf Ubuntu Desktop aktiviert.

Configure SSH

Sobald SSH installiert ist, muss es noch entsprechend konfiguriert werden. Dazu ist es aus Sicherheitstechnischen Belangen unbedingt nötig, den Standard-Port zu ändern, damit niemand Externes auf gut Glück eine SSH-Verbindung mit dem Server hinbekommt. Ausserdem kann man später den "root" deaktivieren und User Logins vorbereiten.

Zuerst ändern wir den default SSH port (22). Dazu muss zuerst das SSH-Konfigurationsfile mit folgendem Befehl geöffnet werden:

```
sudo nano /etc/ssh/sshd_config
```

Sobald das File geöffnet wurde, muss folgende Zeile angepasst werden:

```
# Port 22
```

```
to
```

```
Port 1337
```

Es kann eine beliebige Nummer für den Port verwendet werden (> 1024 bis 65535). Das File wird mit `Ctrl + O` dann `Enter`, dann `Ctrl + X`, dann `Enter` gespeichert und geschlossen.

Nun muss noch der eben beim SSH eingerichtete Port auf der Firewall freigegeben werden:

```
sudo ufw allow 1337
```

Es sollte ausserdem überprüft werden, ob der Port auch noch beim ISP oder auf dem Router freigegeben / weitergeleitet werden muss.

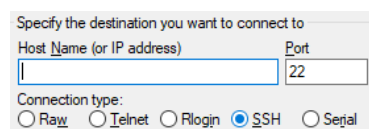
Nun restarten wir SSH mit dem folgenden Befehl, um die Änderungen zu übernehmen:

```
sudo service ssh restart
```

Und das wars. Nun kann SSH verwendet werden. Dazu muss man einfach auf einem Terminal eines anderen Linux-Rechners folgenden Befehl eingeben:

```
ssh username@ip -p1337
```

Wenn ein Windows-Rechner benutzt wird, kann wie gewohnt Putty verwendet werden und dort bei SSH die IP-Adresse der VM und den selbst gewählten Port (1337) angeben. Danach startet die sichere Verbindung zum Server. Danach muss man sich noch mit dem root-Benutzer und Passwort anmelden. Dann wieder `sudo su` und das Passwort angeben und schon hat man root-Rechte mit der Shell auf dem Windows-Rechner:



Die Vorteile in der Praxis liegen auf der Hand. Der Server läuft irgendwo im Netzwerk und man kann mit einem einfachen Programm wie Putty von überall aus dem Netzwerk mit SSH auf den Server zugreifen.

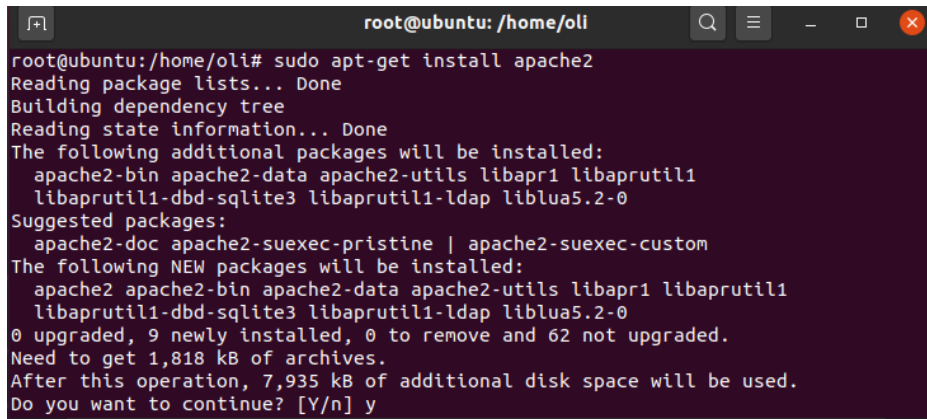
4. Apache2 installieren

Den Anfang dazu bildet die Installation des Apache 2 HTTP Servers. Als aller erstes müssen die Pakete auf dem neusten Stand sein.

```
sudo apt-get update
```

Danach wird der Apache2 heruntergeladen und installiert (bei einer Installation muss immer noch mit Y bestätigt werden):

```
sudo apt-get install apache2
```

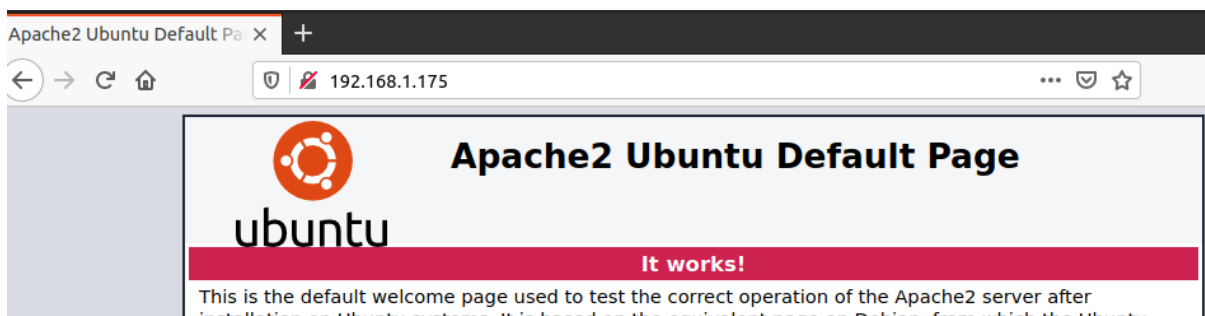


```
root@ubuntu: /home/oli
root@ubuntu:/home/oli# sudo apt-get install apache2
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  apache2-bin apache2-data apache2-utils libapr1 libaprutil1
  libaprutil1-dbd-sqlite3 libaprutil1-ldap liblua5.2-0
Suggested packages:
  apache2-doc apache2-suexec-pristine | apache2-suexec-custom
The following NEW packages will be installed:
  apache2 apache2-bin apache2-data apache2-utils libapr1 libaprutil1
  libaprutil1-dbd-sqlite3 libaprutil1-ldap liblua5.2-0
0 upgraded, 9 newly installed, 0 to remove and 62 not upgraded.
Need to get 1,818 kB of archives.
After this operation, 7,935 kB of additional disk space will be used.
Do you want to continue? [Y/n] y
```

Sobald die Installation beendet ist, muss überprüft werden, ob alles funktioniert. Dazu öffnet man den Browser des Ubuntu-Systems und gibt folgendes ein:

- `http://[IP]/`
Der Webserver greift dabei automatisch über den Port 80 auf die `index.html` im Verzeichnis `/var/www/html/` zu.
- `http://localhost`
Ist dasselbe wie oben, nur ohne IP-Adresse.

Danach sollte folgendes Fenster im Browser ersichtlich sein:



Versuchen Sie auch den Zugriff von einem anderen PC im selben Netz. Achten Sie darauf, dass der andere Rechner sowie der Ubuntu-Rechner im selben Subnetz sind und greifen Sie über die IP-Adresse des Servers im Browser des anderen PCs auf den Webserver des Ubuntu-Rechners zu.

Ab sofort können Dateien ins Verzeichnis `/var/www/html/` hochgeladen werden und sind dann entsprechend unter `http://[IP]/dateiname` zu erreichen, allerdings vorerst nur lokal aus dem Netzwerk und nicht von ausserhalb.

5. Ändern des Homeverzeichnisses einer Webseite?

Dieser Schritt soll nicht jetzt gemacht werden. Machen Sie diesen Schritt erst in Woche 4 oder 5, wenn Sie noch Zeit haben. Machen Sie zuerst unbedingt ein Backup der VM. Im Normalfall sollte man als Beginner mal alles auf Standard lassen und diesen Standard ändern wir hier. Sichern Sie vorher alle Daten.

Hier wird das Ändern des Homeverzeichnisses des zuvor eingerichteten Webservers gezeigt. Wer das Standardverzeichnis von `/var/www` in ein anderes ändern will, geht wie folgt vor: Erstellen Sie testweise im Homeverzeichnis (`/home/oli`) den Order `coolepage.ch`

```
sudo mkdir /home/oli/coolepage.ch
```

Nun muss der Pfad entsprechend in der Konfigurationsdatei von Apache geändert werden. Dazu passen wir die Konfigurationsdatei mit den Standardeinstellungen an.

```
sudo nano /etc/apache2/sites-available/000-default.conf
```

Wir ändern den Pfad `/var/www` in `/home/oli/coolepage.ch` um.

Als nächstes kopiere ich die `index.html` - Datei in das neue Verzeichnis und benenne sie um (sind bereits Dateien vorhanden, ist das natürlich nicht nötig).

```
sudo cp /var/www/html/index.html /home/oli/coolepage.ch/index_neu.html
```

Jetzt noch den Server neustarten:

```
sudo service apache2 restart
```

Rufen wir nun `http://[IP]/index_neu.html` auf (bzw. `192.168.0.xxx/index_neu.html`), wird die Datei im neuen Verzeichnis angezeigt.

Nun müssen noch die entsprechenden Berechtigungen für den Apache2-User `www-data` für das Homeverzeichnis gegeben werden. Wir machen das vereinfacht mit folgendem Vorgehen:

```
sudo nano /etc/apache2/apache2.conf
```

Ändern Sie dann den roten Ausschnitt, dass er wie auf dem Screenshot aussieht:

```

# The former is used by web applications packages, while the latter
# may be used for local directories.  You may want to place your
# own content here, or in any related virtual host.
Directory />
    Options FollowSymLinks
    AllowOverride None
    #Require all denied
    Require all granted
/Directory>

Directory /usr/share>
    AllowOverride None
    Require all granted
/Directory>

Directory /var/www/>
    Options Indexes FollowSymLinks
```

6. PHP installieren

Um nicht nur reines HTML anzeigen und den Content dynamisch gestalten zu können, benötigt man PHP. Dazu richten wir die benötigten Pakete ein und testen die Installation auf Funktionstüchtigkeit. Zuerst wird PHP installiert:

```
sudo apt-get install php libapache2-mod-php
```

Nach der Installation muss wieder getestet werden, ob alles funktioniert hat. Dazu geht man wieder ins Verzeichnis des Webservers:

```
cd /var/www/html
```

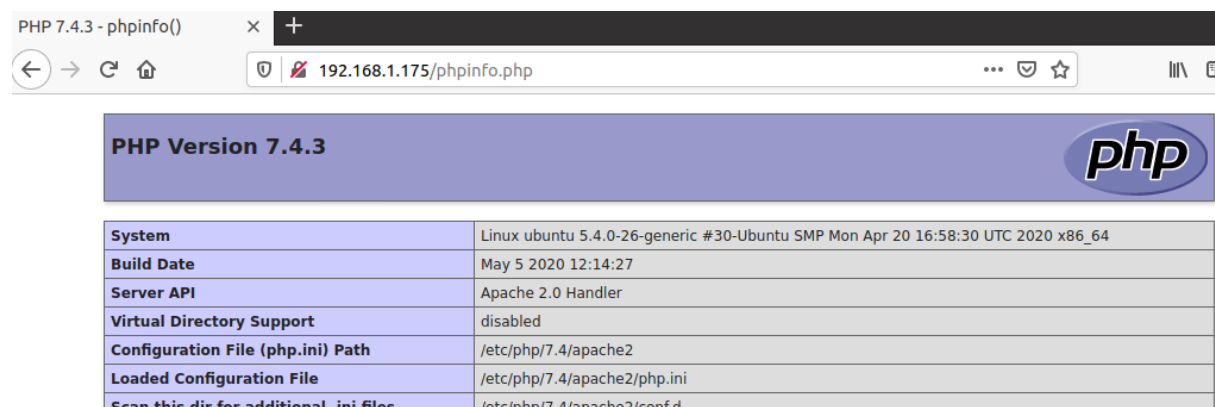
Hier wird eine neue Datei *phpinfo.php* erstellt:

```
sudo nano "phpinfo.php"
```

und schreiben hinein:

```
<?php
    phpinfo();
?>
```

Gespeichert wird die Datei wie immer mit STRG + O und mit STRG + X der Editor geschlossen. Wenn nun im Browser die Adresse [http://\[IP\]/phpinfo.php](http://[IP]/phpinfo.php) geöffnet wird, muss die folgende Informationsseite zu PHP zu sehen sein:



PHP Version 7.4.3	
System	Linux ubuntu 5.4.0-26-generic #30-Ubuntu SMP Mon Apr 20 16:58:30 UTC 2020 x86_64
Build Date	May 5 2020 12:14:27
Server API	Apache 2.0 Handler
Virtual Directory Support	disabled
Configuration File (php.ini) Path	/etc/php/7.4/apache2
Loaded Configuration File	/etc/php/7.4/apache2/php.ini
Scan this dir for additional .ini files	/etc/php/7.4/apache2/conf.d

Wenn das funktioniert hat, können bereits erste Skripte und Funktionen in PHP geschrieben werden.

7. MySQL installieren

Bei MySQL handelt es sich um ein weltweit sehr verbreitetes relationales Datenbanksystem. Es bildet in vielen Content-Management-Systemen die technische Grundlage für die Speicherung der Daten. Das Datenbanksystem kann grosse Datenmengen schnell verarbeiten. Das wollen wir auch, damit wir dynamische Webseiten mit einer Datenbank im Hintergrund haben können.

```
sudo apt-get install mysql-server
```

(Kann auch sein, dass der obige Befehl nicht geht, dann bitte «sudo apt-get install mariadb-server-10.0». Auch hier muss evtl. die Version (10.0) an eine aktuelle angepasst werden.)

Nun müssen wir den Benutzer zum MySQL-Benutzer. Dazu gehen wir in die Befehlskonsole von MySQL:

```
sudo mysql
```

Mit folgendem Befehl wird der root-Benutzer MySQL zugeordnet (für `mynewpassword` ein entsprechendes Passwort eingeben):

```
ALTER USER 'root'@'localhost' IDENTIFIED WITH mysql_native_password by 'mynewpassword';
```

Verlassen Sie die MySQL-Konsole mit `exit`.

Nun ist das MySQL installiert, es müssen nun aber noch einige Standardeinstellungen angepasst werden, damit das System sicher wird:

```
sudo mysql_secure_installation
```

- Would you like to setup VALIDATE PASSWORD component? N
- Change the password for root N
- Remove anonymous users: Y
- Disallow root login remotely: Y
- Remove test database and access to it: Y
- Reload privilege table now: Y

Nun wollen wir überprüfen, welcher User bereits vorhanden sind und gehen dazu in die MySQL-Konsole mit root-Rechten (entweder `sudo` oder schon mit `root` rein):

```
sudo mysql -u root -p
```

Nun muss das root-Passwort der MySQL-DB angegeben werden, welches wir vorhin bei der Secure-Installation angegeben haben. Nun wird mit folgendem Befehl die Benutzerliste ausgelesen:

```
SELECT User FROM mysql.user;
```

```
mysql> SELECT User FROM mysql.user;
+-----+
| User          |
+-----+
| debian-sys-maint |
| mysql.infoschema |
| mysql.session  |
| mysql.sys      |
| root           |
+-----+
5 rows in set (0.00 sec)
```

Um später einen Zugriff auf die MySQL-Datenbank zu gewährleisten, erstellen wir nun einen Benutzer. Bei solchen Zugriffen wird aus Sicherheitsgründen nie der Root-Benutzer genommen.

Dieser sollte nur für administrative Belange benutzt werden. Mit folgender Befehlskette erhält der neue Benutzer `mysql_admin` volle administrative Berechtigungen:

```
create user 'mysql_admin'@'localhost' identified by 'Passwort';
> Enter drücken

grant all privileges on *.* to 'mysql_admin'@'localhost' with grant option;
> Enter drücken

flush privileges;
> Enter drücken

Exit
> Enter drücken

sudo service mysql restart
> Enter drücken
```

Falls das Passwort geändert werden muss (zuerst wieder in die `mysql`-Konsole rein):

```
ALTER USER 'mysql_admin'@'localhost' IDENTIFIED BY 'hierneuespasswort';
```

Mit diesem Benutzer kann man sich jetzt am Webinterface von `phpmyadmin` (welches wir im nächsten Schritt installieren) anmelden.

Wir überprüfen, ob der User auch angelegt wurde:

```
sudo mysql -u root -p

SELECT User FROM mysql.user;
```

Hier sollte nun neben den anderen auch `mysql_admin` ersichtlich sein. Nun wollen wir noch dem `root`-User ein Passwort geben, der hat standardmässig nämlich keines:

```
ALTER USER 'root'@'localhost' IDENTIFIED BY 'hierneuespasswort';
```

Nun haben wir also ein Passwort, um uns bei `MySQL` einzuloggen und es von der Konsole zu bedienen. Ausserdem haben wir nun 3 Benutzer für `phpMyAdmin`:

- `root` `root`
- `mysql_admin` `Superuser`
- `phpmyadmin` `User` (wird mit der Installation von `phpmyadmin` erstellt)

Deinstallieren von `MySQL-Server`:

```
sudo apt-get remove mysql-server
sudo apt-get purge mysql-server
sudo apt-get autoremove
```

Starten oder Statusabfragen des `MySQL-Service`:

```
systemctl start mysql.service
systemctl status mysql service
```

Error 1819 Your password does not satisfy the current policy requirements

Ihr Passwort erfüllt nicht die Bedingungen des Systems. Entweder geben Sie ein komplexes Passwort ein, oder Sie gehen wie folgt vor:

```
sudo mysql -u root -p
```

```
SHOW VARIABLES LIKE 'validate_password%';
```

Dann sollten Sie die Bedingungen sehen, welche Ihr Passwort erfüllen muss:

```
3 +-----+-----+
4 | Variable_name                | Value |
5 +-----+-----+
6 | validate_password_check_user_name | OFF   |
7 | validate_password_dictionary_file |       |
8 | validate_password_length        | 8     |
9 | validate_password_mixed_case_count | 1     |
10 | validate_password_number_count   | 1     |
11 | validate_password_policy         | MEDIUM |
12 | validate_password_special_char_count | 1     |
13 +-----+-----+
```

Die einzelnen Punkte können dann folgendermassen angepasst werden:

```
SET GLOBAL validate_password_policy=LOW;
```

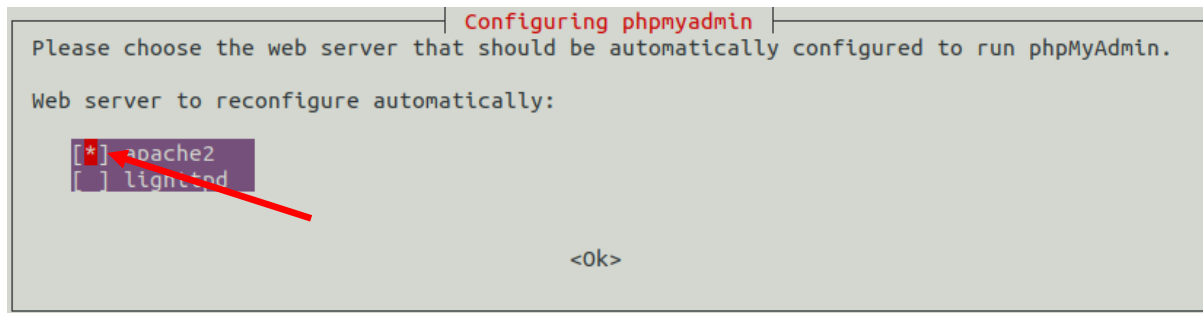
8. phpMyAdmin installieren

Nach der Installation von Apache, PHP und MySQL ist phpMyAdmin an der Reihe. phpMyAdmin ist eine kostenlose Applikation die es ermöglicht MySQL-Datenbanken über den Internetbrowser zu verwalten. So können Sie Ihre Datenbanken auch von Fremden Rechnern oder Standorten aus bearbeiten. Es sind keine Kenntnisse in MySQL notwendig, da phpMyAdmin nach dem Prinzip von WYSIWYG (What you see is what you get) arbeitet.

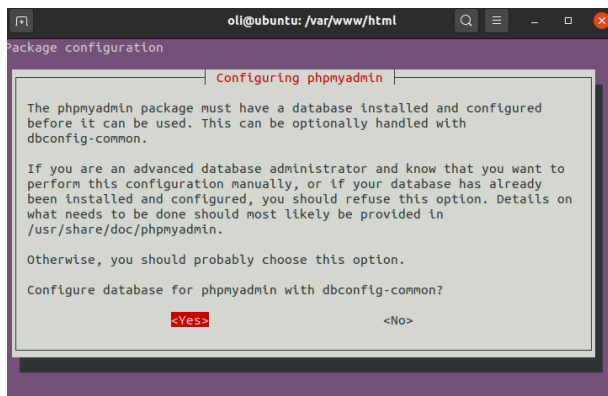
Dieser Schritt ist nicht unbedingt nötig, aber für das einfachere Auslesen der Daten im Browser sehr nützlich.

```
sudo apt-get install -y phpmyadmin
```

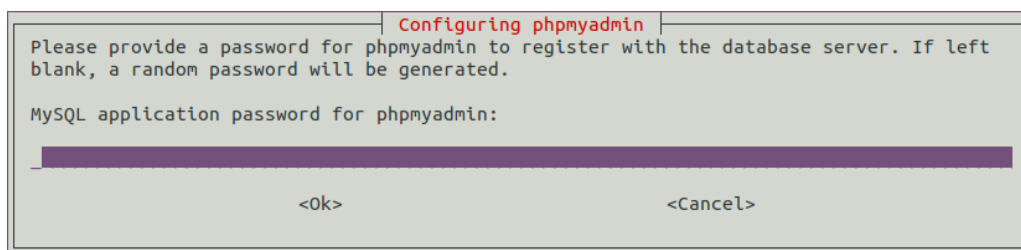
Nach der Bestätigung wählen wir apache2 aus (mittels Leertaste). Es muss zwingend ein Sternchen vor apache2 erscheinen, ansonsten gibt es nachher einen 404-Error!



Daraufhin wird gefragt, ob einige Datenbanken, die phpMyAdmin benötigt erstellt werden sollen. «Yes» weiter.



Jetzt wird noch nach einem Passwort zum Einloggen in phpMyAdmin gefordert (Buchstaben und Zahlen). Dieses Passwort wird für den User phpmyadmin in der Web-Oberfläche von phpMyAdmin benutzt.



Nach der Eingabe und Bestätigung muss Apache mit phpMyAdmin noch verknüpft werden. Dazu bearbeiten wir die Datei `/etc/php/8.1/apache2/php.ini`

Da die 8.1 die Version von PHP ist, kann es sein, dass sich diese Zahl bei Ihrer Installation unterscheidet. Überprüfen Sie, was für ein Verzeichnis in /etc/php vorhanden ist und passen Sie den folgenden Befehl entsprechend an:

```
sudo nano /etc/php/8.1/apache2/php.ini
```

Ganz am Ende der Datei fügen wir **extension=mysql.so** ein.

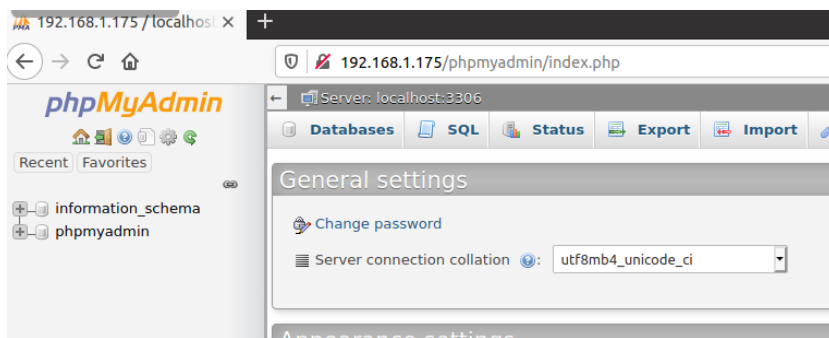
```
[ffi]
; FFI API restriction. Possible values:
; "preload" - enabled in CLI scripts and preloaded files (default)
; "false" - always disabled
; "true" - always enabled
; ffi.enable=preload

; List of headers files to preload, wildcard patterns allowed.
; ffi.preload=

extension=mysql.so
```

Dann nur noch mit STRG + O und STRG + X speichern und beenden.

Ab jetzt kann man sich unter **http://[IP]/phpmyadmin/** einloggen. Der Standardbenutzername ist phpmyadmin und das Passwort ist das, welches bei der Konfiguration von phpMyAdmin angegeben wurde (MySQL application password for phpmyadmin).



Fertig. Jetzt können auch über das Interface Datenbanken erstellt und verwaltet werden.

Hier noch einige zusätzlichen Konfigurationsmöglichkeiten / Troubleshootings, welche im Normalfall nicht angewendet werden müssen:

phpmyadmin rekonfigurieren

```
sudo dpkg-reconfigure phpmyadmin
```

Wenn http://[IP]/phpmyadmin einen 404 gibt, aber alles korrekt installiert wurde:

```
sudo dpkg-reconfigure phpmyadmin
```

Erste Abfrage: ok

Reinstall database: No

Dann zwingend bei apache2 auswählen und mit der Leertaste anwählen, bis ein Stern erscheint:

Web server to reconfigure automatically:

```
[*] apache2
[ ] lighttpd
```

Wenn PW oder Benutzername vergessen:

```
sudo cat /etc/phpmyadmin/config-db.php | grep -iA6 dbuser
```

Neuen Benutzer mit Root Rechten erstellen, damit man sich an [IP]/phpmyadmin anmelden kann

Nach der Installation von MySQL und phpmyadmin auf einem Linux Ubuntu Server 18.04 war das Anmelden mit dem Root Account am Webinterface von phpmyadmin nicht möglich. Allerdings das Anmelden an der MySQL Konsole direkt schon. Erst das Anlegen eines neuen Benutzers lieferte die Lösung, den dieser hatte anschliessend Zugriff auf das Webinterface von phpmyadmin. Da er mit administrativen Rechten erstellt wurde, konnte man damit auch neue Datenbanken erstellen. Hier das Vorgehen:

```
mysql -u root -p
```

Nachdem Anmelden an der MySQL Konsole, kann jetzt der neue Benutzer erstellt werden. Der neue Benutzer Chef erhält dabei volle administrative Berechtigungen:

```
create user 'chef'@'localhost' identified by 'Passw0rt!';
> Enter drücken

grant all privileges on *.* to 'chef'@'localhost' with grant option;
> Enter drücken

flush privileges;
> Enter drücken

Exit
> Enter drücken

Service mysql restart
> Enter drücken
```

Mit diesem Benutzer kann man sich jetzt am Webinterface von phpmyadmin anmelden.

Komplette Deinstallation von phpmyadmin

1. `sudo apt-get remove phpMyAdmin` Deinstalliert phpmyadmin
2. `sudo apt-get purge phpMyAdmin` Entfernt Konfigurationsdateien von phpmyadmin
3. `sudo apt-get autoremove` Entfernt alle Pakete, die nicht gebraucht werden.

9. FTP-Server (proftpd) mit User einrichten

Mit der Abkürzung FTP wird das „File Transfer Protocol“ bezeichnet. Übersetzt bedeutet FTP „Datei-Übertragungsverfahren“. FTP ermöglicht es, Daten zwischen verschiedenen Rechnern auszutauschen – also beispielsweise vom eigenen Rechner auf den Speicherplatz in einem Webhosting-Paket. Das Protokoll funktioniert unabhängig von den benutzten Betriebssystemen und Verbindungswegen.

Zunächst die Installation:

```
sudo apt-get install proftpd
```

Im Grunde wären wir hier bereits fertig, allerdings hätte jetzt jeder Nutzer nur Zugriff auf sein eigenes Home Verzeichnis (z.B. /home/oli). Deshalb erstellen wir einen neuen User. Dazu wechseln wir erst einmal das Verzeichnis.

```
cd /etc/proftpd/
```

Nun soll der Benutzer **webserv_user** mit dem Homeverzeichnis **/var/www/** (falls das Rootverzeichnis von Apache geändert wurde, sollte dies natürlich angepasst werden. Gleiches gilt, wenn ein FTP-User auf andere Verzeichnisse Zugriff haben soll) erstellt werden.

```
sudo ftpasswd --passwd --name webserv_user --gid 33 --uid 33 --home /var/www/ --shell /bin/false
```

Nun nur noch das Passwort eingeben und bestätigen. Falls das Passwort des Benutzers zu einem späteren Zeitpunkt geändert werden soll, einfach wieder in das Verzeichnis wechseln und den Befehl erneut ausführen.

Um den User noch freizuschalten, bearbeiten wir die Konfigurationsdatei:

```
sudo nano /etc/proftpd/proftpd.conf
```

und fügen folgenden Code am Ende ein:

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

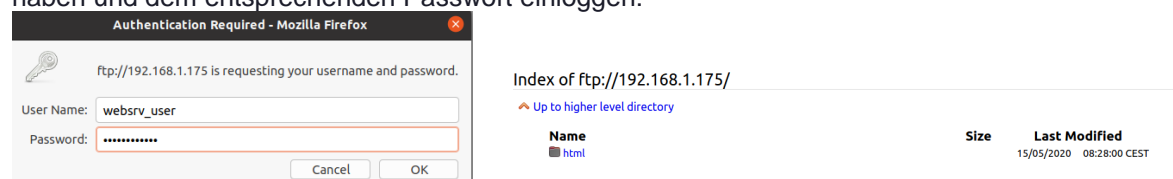
```
# Eventually create file patterns like this: /etc/proftpd/conf.d/*.conf
#
Include /etc/proftpd/conf.d/

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

Mittels STRG + O und STRG + X speichern und beenden wir den Editor. Zum Abschluss muss der Server neu gestartet werden:

```
sudo /etc/init.d/proftpd restart
```

Geben Sie dazu ftp://[IP] im Explorer ein. Danach müssen Sie sich mit dem User, den Sie erstellt haben und dem entsprechenden Passwort einloggen:



Wenn Sie nun auf «html» klicken, sehen Sie die beiden Files, welche am Anfang des Tutorials eingerichtet wurden.

Um Zugriff von ausserhalb des eigenen Rechners zu bekommen, muss die Firewall freigeschaltet werden:

```
sudo ufw allow 20
sudo ufw allow 21
```

Damit wir auch auf File rauploaden können, müssen wir die Rechte des Ordners etwas anpassen:

```
chmod g+s /var/www
chmod 777 /var/www

sudo /etc/init.d/proftpd restart
```

(dieser Befehl gibt nur das Verzeichnis /var/www zum Beschreiben frei, aber keine Unterordner wie /var/www/html)

Bei den Rechten kommt zuerst der Besitzer, dann die Gruppe, dann der User mit r=read, w=write, x=ausführen.

Mögliche Werte für:				
	chmod (octal)	umask (octal)	Symbolisch	Binäre
Lesen, schreiben und ausführen	7	0	rwX	111
Lesen und Schreiben	6	1	rw-	110
Lesen und Ausführen	5	2	r-X	101
Nur lesen	4	3	r--	100
Schreiben und Ausführen	3	4	-wX	011
Nur Schreiben	2	5	-w-	010
Nur Ausführen	1	6	--X	001
Keine Rechte	0	7	---	000

Sollte etwas nicht funktionieren, kann mit folgendem Befehl der Status des FTP-Servers inklusive Fehlermeldungen angezeigt werden:

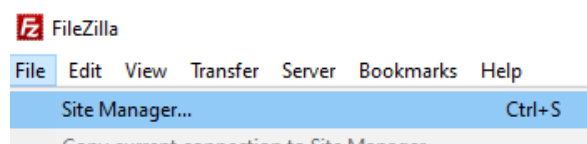
```
systemctl status proftpd
```

Zugriff per FTP-Client

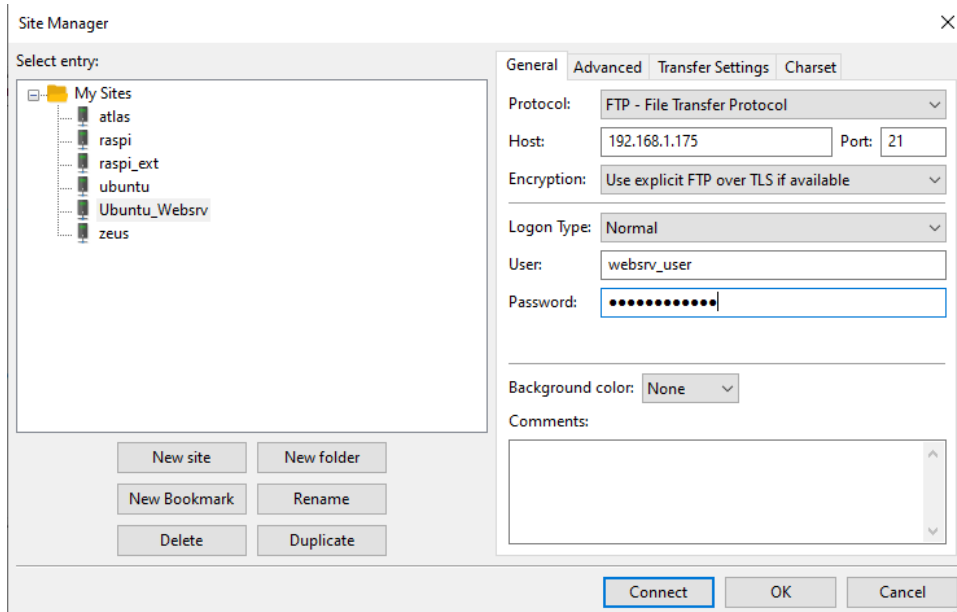
Im Normalfall greift man auf einen privaten FTP-Server aber nicht per Browser, sondern per FTP-Client zu. Laden Sie nun also auf Ihren Windows-Rechner, auf dem Ihre VM läuft und die im selben Subnetz wie die VM ist Filezilla runter und installieren es:

- <https://filezilla-project.org/>

Starten Sie das Programm und gehen Sie auf Datei -> Site Manager:

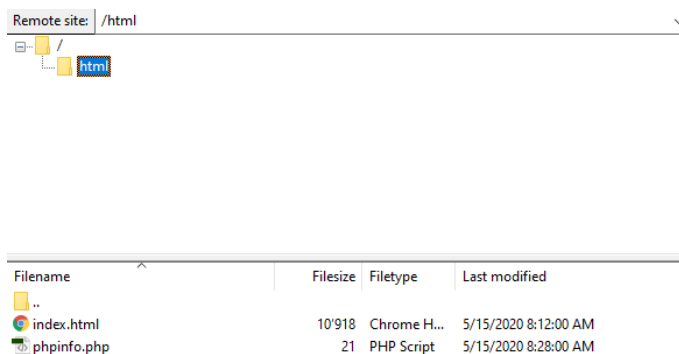


Klicken Sie dann auf «New Site» und geben Sie dem FTP-Server einen Namen (zum Beispiel Ubuntu_Websrv):



Geben Sie danach die IP-Adresse des Ubuntu-Webservers an und den Port 21. Danach wieder ihren Benutzer «webserv_user» und das entsprechende Passwort.

Danach können Sie mit «Connect» auf den FTP-Server connecten. Wenn alles korrekt eingegeben wurde, sollten nun auf der rechten Seite von Filezilla (Remote site) wieder ihre beiden Dateien ersichtlich sein:



Ab hier würde der FTP funktionieren. Wir wollen aber noch eine TLS-Verschlüsselung, damit die Dateien nicht im Klartext übertragen werden.

SSL/TLS-verschlüsselte FTP-Verbindung mit mod_tls

Das TLS Modul ermöglicht eine verschlüsselte Verbindung über SSL/TLS zum ProFTPD Server.

Achtung: Ohne Verschlüsselung überträgt das FTP-Protokoll sowohl Login- als auch normale Daten im Klartext! Der Einsatz von SSL/TLS wird für Produktivumgebungen daher dringend empfohlen.

Das TLS-Modul muss ab Version 1.3 zusätzlich installiert werden:

```
apt install proftpd-mod-crypto
```

Nun ist es in `/etc/proftpd/modules.conf` enthalten und kann aktiviert werden.

Zertifikat erstellen

Es kann mit unterschiedlichen Varianten ein Zertifikat erstellt werden. Wir verwenden eines von openssl. Dazu muss man nach der Installation noch einige Informationen angeben, welche dann im Zertifikat angezeigt werden.

```
openssl req -x509 -newkey rsa:2048 -keyout /etc/ssl/private/proftpd.key -out /etc/ssl/certs/proftpd.crt -nodes -days 365
```

TLS konfigurieren

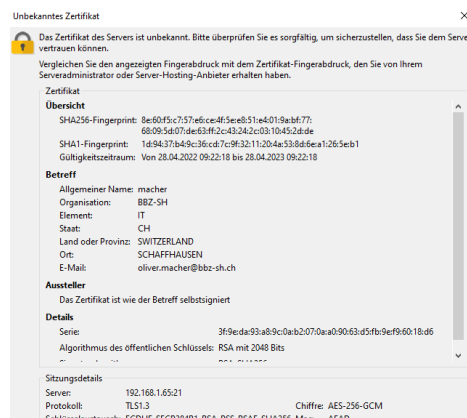
Das in den Paketquellen vorhandene ProFTPD Paket unterstützt auch TLSv1.3. Im `conf.d` Verzeichnis wird eine eigene Konfigurationsdatei für SSL/TLS erstellt:

```
sudo nano /etc/proftpd/conf.d/tls.conf
```

Diese soll folgenden Inhalt haben:

```
<IfModule mod_tls.c>
    TLSEngine on
    TLSLog /var/log/proftpd/tls.log
    TLSProtocol TLSv1.3
    TLSRSACertificateFile /etc/ssl/certs/proftpd.crt
    TLSRSACertificateKeyFile /etc/ssl/private/proftpd.key
    TLSVerifyClient off
    TLSOptions NoSessionReuseRequired
    TLSRequired on
</IfModule>
```

Anschliessend wird ProFTPD neu gestartet. Falls hier eine Fehlermeldung kommt, hat es wahrscheinlich im `tls.conf` einen Schreibfehler. Ab jetzt kann nur noch mit einem FTP-Programm wie FileZilla auf den FTP zugegriffen werden. Aktivieren Sie dort den Active-Mode und akzeptieren Sie das TLS-Certificate.



Analyse bei Verbindungsproblemen

Bei Problemen beim Aufbau der FTP Verbindungen können folgende Dinge überprüft werden:

1. ProFTPD Dienst läuft: `$ sudo service proftpd status`
2. ProFTPD lauscht auf Port 21: `$ sudo netstat -tlnp|grep proftpd`
3. Fehlermeldungen im ProFTPD Log: `$ sudo tail -20 /var/log/proftpd/proftpd.log`
4. Fehlermeldungen im ProFTPD TLS Log: `$ sudo tail -20 /var/log/proftpd/tls.log`
5. Verbindungstest auf Port 21 mit telnet: `$ telnet 192.0.2.10 21`
6. Verbindungstest auf Port 21 mit TLS: `$ openssl s_client -connect 192.0.2.10:21 -starttls ftp`

Probleme mit der Verbindung von extern (PassiveMode)

With the passive mode, most of the configuration burden is on the server side. The server administrator should setup the server as described below.

The firewall and NAT on the FTP server side have to be configured not only to allow/route the incoming connections on FTP port 21,[2](#) but also a range of ports for the incoming data connections. Typically, the FTP server software has a configuration option to setup a range of the ports, the server will use. And the same range has to be opened/routed on the firewall/NAT.

When the FTP server is behind a NAT, it needs to know it's external IP address, so it can provide it to the client in a response to PASV command.

Lösung:

```
sudo nano /etc/proftpd/proftpd.conf
```

Es müssen zwei Anpassungen vorgenommen werden. Zum einen muss `MasqueradeAddress` mit der aktuellen, öffentlichen IP eingefügt werden. Damit kann der FTP-Server im Passive-Modus dem Client die korrekte IP (nicht die vom internen Netz) weitergeben:

```
MasqueradeAddress 83.78.64.23  
MasqueradeAddress blubb.ddns.net (falls vorhanden)
```

Ausserdem braucht der PassiveMode die ganze Port-Range >1024. Dies kann man jedoch nicht alles freigeben, da sonst gefährliche Ports offen waren. Deswegen kann man mit den `PassivePorts` die zu benutzenden Ports freigeben. Diese müssen dann aber auch im Router freigegeben und weitergeleitet werden:

```
PassivePorts 60000 60050
```

10. Zugriff Aussen mit Portforwarding (DynDNS mit No-IP)

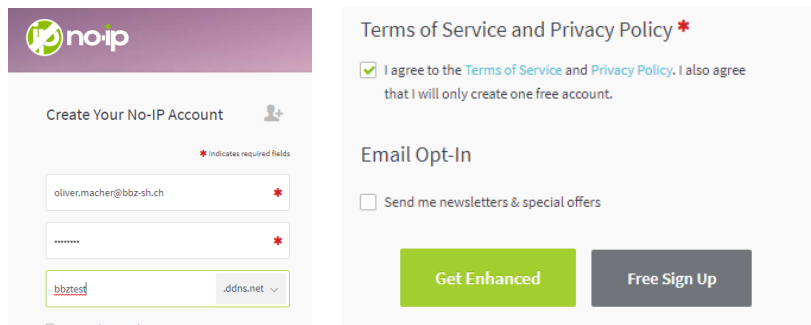
Das Surfen im Internet und das Senden von E-Mails, usw. funktioniert alles ohne Probleme, weil diese Konfigurationen und Ports in E-Mail-Clients und Webbrowsern vordefiniert sind. Zum Beispiel geht der HTTP-Verkehr immer über Port 80. Das wird von der IANA definiert und jeder muss ihm folgen. SMTP, das zum Senden von E-Mails verwendet wird, verwendet standardmässig Port 25. Was passiert jedoch, wenn jemand versucht, sich beispielsweise über Port 80 mit Ihrem Router zu verbinden und so auf Ihren Webserver zuzugreifen? Wenn Sie keine Portweiterleitung eingerichtet haben und Ihre Firewall aktiviert ist, wird diese Verbindung standardmässig beendet. Wenn Sie einen Webserver in Ihrem lokalen Netzwerk ausführen möchten, müssen Sie den an Port 80 eingehenden Datenverkehr an die lokale IP-Adresse des Computers weiterleiten, auf dem der Webserver ausgeführt wird.

In diesem Kapitel der Webserver Installation geht es darum, den Server auch ausserhalb des lokalen Netzwerks mittels eines DNS Servers (wie No-IP, DynDNS) erreichbar zu machen.

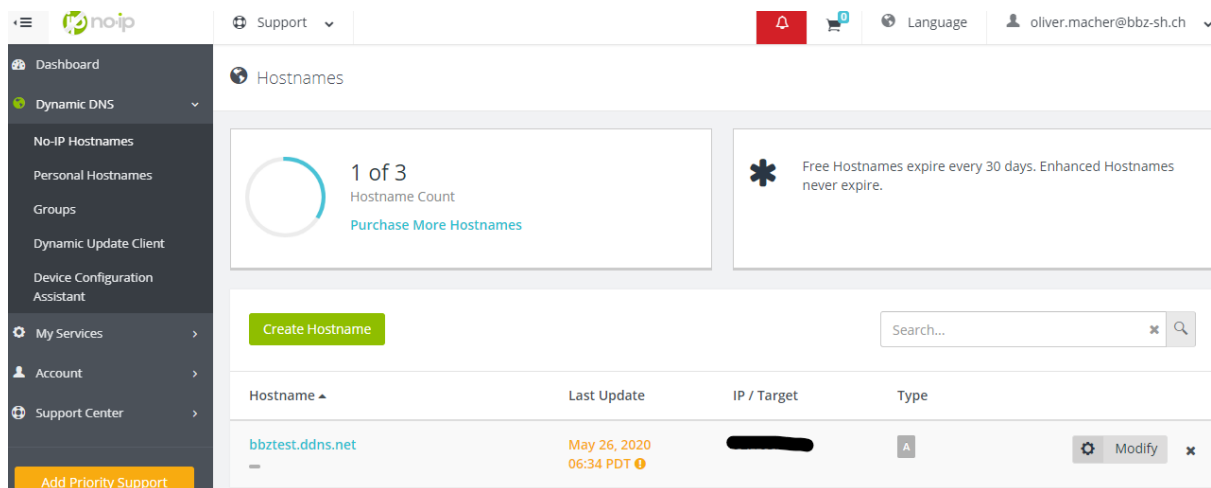
Mit den bisherigen Programmen konnten wir einen vollständigen Webserver aufsetzen, doch sollen die Dateien in den allermeisten Fällen auch über das Internet erreichbar sein.

Als erstes benötigen wir einen DNS-Anbieter, wobei ich No-IP bevorzuge (kostenlos). Also legen wir einen kostenlosen Account unter <https://www.noip.com/sign-up> an. Der einzige „Nachteil“ der Free Version ist, dass jeden Monat eine Mail kommt, in der man aufgefordert wird, den Account zu bestätigen (mittels Capcha Eingabe).

Zuerst muss die eigene Mail-Adresse, ein Passwort und die Domain angegeben werden. Die Domain ist die Adresse, wie ihr Webserver nachher von aussen erreichbar sein soll (hier: bbztest.ddns.net). Danach mit «Free Signup» bestätigen und danach die Mail-Adresse bestätigen (sie bekommen eine Mail).



Nach dem Login auf «My Account» klicken. Danach kann man auf «Dynamic DNS» klicken und sieht dort die vorher angegebene Domain. Im Normalfall sollte die IP-Adresse des verwendeten PC's direkt eingegeben worden sein. Ansonsten diese unter «Modify» anpassen.



Hostname	Last Update	IP / Target	Type
bbztest.ddns.net	May 26, 2020 06:34 PDT	[REDACTED]	A

Beim Feld «Type» sollte ein «A» stehen. Wenn nicht kann unter «Create hostname» eine weitere Domain hinzugefügt werden. Dabei darauf achten, dass «DNS Host (A)» ausgewählt ist. Sie können pro Free-Account bis zu 3 Domains erstellen.

Create a Hostname

Hostname ⓘ	Domain ⓘ
<input type="text" value="myhost"/>	<input type="text" value="ddns.net"/> ▼
Record Type	IPv4 Address ⓘ
<input checked="" type="radio"/> DNS Host (A) ⓘ	<input type="text" value="██████████"/>
<input type="radio"/> AAAA (IPv6) ⓘ	
<input type="radio"/> DNS Alias (CNAME) ⓘ	
<input type="radio"/> Web Redirect ⓘ	

Wenn diese Schritte erfolgreich erstellt worden sind, können wir uns wieder der Konfiguration des Webservers widmen.

Das Paket von no-ip muss nun noch auf dem Webserver installiert werden, damit die dynamische DNS auch immer die aktuelle IP-Adresse des Webservers kennt. Dazu laden wir die Linux-Installationsdatei herunter (könnte auch manuell heruntergeladen und installiert werden):

```
sudo wget http://www.noip.com/client/linux/noip-duc-linux.tar.gz
```

Dieses Packet muss nun noch entpackt werden:

```
sudo tar xf noip-duc-linux.tar.gz
```

Zum Installieren des Pakets muss nun noch der Ordner gewechselt werden:

```
cd noip-2.*
```

Danach wird das Packet installiert:

```
sudo apt-get install build-essential
sudo make install
```

Daraufhin wird eine Abfrage nach der E-Mail-Adresse kommen, sowie dem verwendeten Passwort. Dabei handelt es sich um die Angaben, welche auf der Homepage von noip.com gemacht wurden. Als Intervall 30 Sekunden lassen und auf „Do you wish to run something at successful update?“ mit nein antworten.

```
Auto configuration for Linux client of no-ip.com.
Please enter the login/email string for no-ip.com oliver.macher@bbz-sh.ch
Please enter the password for user 'oliver.macher@bbz-sh.ch' *****

Only one host [bbztest.ddns.net] is registered to this account.
It will be used.
Please enter an update interval:[30] 30
Do you wish to run something at successful update?[N] (y/N) n

New configuration file '/tmp/no-ip2.conf' created.

mv /tmp/no-ip2.conf /usr/local/etc/no-ip2.conf
oli@ubuntu:~/noip-2.1.9-1$
```

Danach muss der das Konfigurationsfile nochmals erstellt werden und die Daten nochmals eingegeben werden:

```
sudo /usr/local/bin/noip2 -C
```

Danach muss der Service gestartet werden:

```
sudo noip2
```

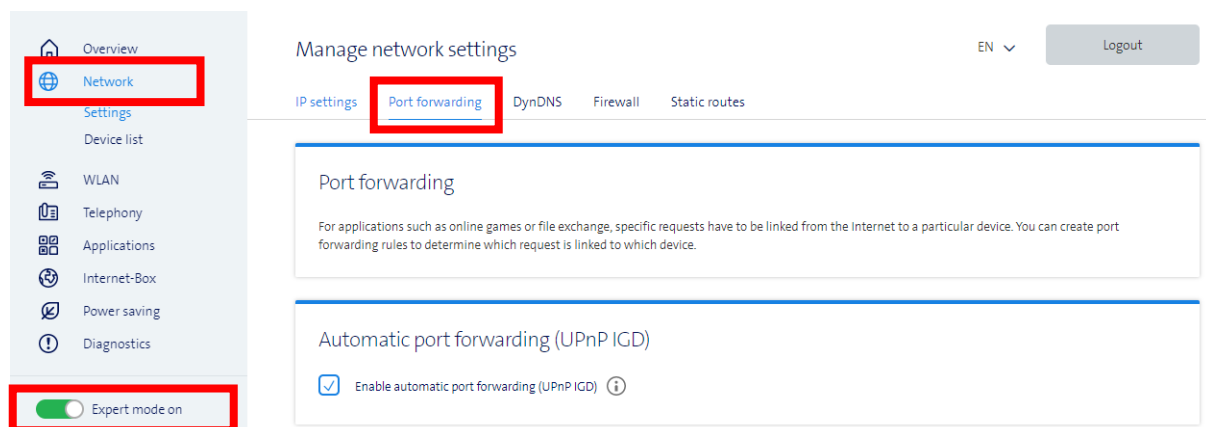
Den Webserver wieder starten:

```
sudo service apache2 start
```

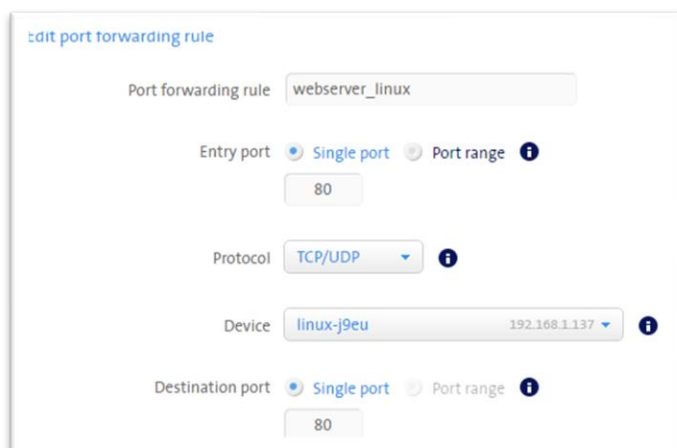
Achtung: Wenn der Webserver neu gestartet wird, startet No-IP nicht automatisch. Es muss also entweder immer manuell gestartet werden oder wir machen einen Eintrag zum automatischen Start (nächstes Kapitel).

Damit der Service nun auch von ausserhalb ihres internen Netzes auf den Server weiterleiten kann (beim Aufruf von *deine-domain.no-ip.org*) müssen im Router noch die Ports **80** und **443** an die IP-Adresse des Webserver weitergeleitet (Portforwarding) werden. Auch hier wieder zuerst die verantwortliche Person konsultieren. Bei den meisten Routern ist dies unter dem Reiter **Port Forwarding** möglich (Bedienungsanleitung konsultieren). Als IP-Adresse die interne IP-Adresse des Ubuntu-Servers angeben. Diese findet man einfach mit dem Terminal-Befehl «ifconfig» heraus.

Folgendes Beispiel gilt im «Expert mode» des Swisscom-Routers:



Danach unten auf «Add rule» und für die entsprechenden Ports eine neue Regeln angeben.



Beim Port-Forwarding kann man auch verschiedene Zugriffe definieren. Wenn ich zum Beispiel auf verschiedene Webserver von aussen zugreifen will:

<http://bbztest.ddns.net:3333/>

Dazu braucht es einfach ein Forwarding von 3333 zur internen IP-Adresse mit dem Port 80.

No-IP deinstallieren

Zuerst überprüfen, welchen Prozess noip gerade braucht:

```
sudo /usr/local/bin/noip2 -C
```

Dies gibt mir die Rückmeldung:

```
llt@ubuntu:~$ sudo /usr/local/bin/noip2 -C
Configuration file '/usr/local/etc/no-ip2.conf' is in use by process 11225
Ending!
```

Also wird der Prozess 11225 verwendet und muss beendet werden:

```
sudo /usr/local/bin/noip2 -K 11225
```

Danach muss noip vom usr/local/bin gelöscht werden:

```
sudo rm /usr/local/bin/noip2
```

Danach müssen die Config-Files gelöscht werden (das zweite File muss nicht vorhanden sein):

```
sudo rm /usr/local/etc/no-ip2.conf
```

```
sudo rm /usr/local/etc/NO-IPalmco0
```

Danach noch die erstellten Verzeichnisse und Dateien löschen:

```
sudo rm -r /usr/local/src/noip-2.1.9-1
```

```
sudo rm noip-duc-linux.tar.gz
```

```
sudo rm -rf noip-2.1.9-1
```

11. Autostart: Programm automatisch starten lassen

Immer wieder passiert es, dass man Programme installiert, aber diese nicht automatisch beim Hochfahren starten. Anhand von noip werden wir in diesem Kapitel zeigen, wie man ein Programm automatisch starten lassen kann.

Zuerst wechseln wir zum root-User und setzen wir die Rechte des Config-Files um (ändern wir später wieder):

```
sudo su
sudo chmod 7777 /usr/local/etc/no-ip2.conf
```

Nun muss im Verzeichnis /etc/init.d/ ein Skript erstellt werden, mittels welchem das Programm gestartet wird, daher erstellen wir ein Skript (es muss nicht unbedingt eine Datei-Endung haben)

```
sudo nano /etc/init.d/noip_bbztest
```

mit folgendem Inhalt:

```
#####
#!/bin/sh
case "$1" in
  start)
    echo "noip2 bbztest wird gestartet"
    /usr/local/bin/noip2
    ;;
  stop)
    echo -n " noip2 bbztest wird beendet"
    for i in `noip2 -S 2>&1 | grep Process | awk '{print $2}' | tr -d ', '`
    do
      noip2 -K $i
    done
    ;;
  *)
    echo "Usage: $0 {start|stop}"
    exit 1
esac
exit 0
#####
```

Danach mit Ctrl + O und Ctrl + X beenden. Danach sollte das Script gestartet und gestoppt werden können:

Anstelle von noip2 kann hier natürlich auch jedes andere installierte Programm stehen. Es muss aber darauf geachtet werden, dass auf keine Benutzerinteraktion gewartet wird (wie das Bestätigen bei apt-get), da es im schlimmsten Fall dazu kommt, dass beim Booten auf die Eingabe gewartet wird und der Webserver nicht startet.

Als nächstes weisen wir die benötigten Rechte zu (Lesen & Schreiben):

```
sudo chmod 755 /etc/init.d/noip_bbztest
```

und testen das Skript indem wir es starten:

```
sudo /etc/init.d/noip_bbztest start
```

und gleich wieder stoppen:

```
sudo /etc/init.d/noip_bbztest stop
```

Danach werden die Rechte der entsprechenden Files wieder zurückgesetzt:

```
sudo chmod 700 /usr/local/bin/noip2
sudo chown root:root /usr/local/bin/noip2
sudo chmod 700 /etc/init.d/noip_bbztest
sudo chown root:root /etc/init.d/noip_bbztest
sudo chmod 700 /usr/local/etc/no-ip2.conf
sudo chown root:root /usr/local/etc/no-ip2.conf
```

Damit das Skript beim Booten auch aufgerufen wird, führen wir folgendes aus:

```
sudo update-rc.d noip_bbztest defaults
```

Das Programm kann mit folgendem Befehl wieder vom Autostart entfernt werden:

```
sudo update-rc.d -f noip_bbztest remove
```

Autostart – weitere Möglichkeiten

Eine andere Option zum Starten eines Skripts oder Programms ist „Cron“. Dadurch ist es möglich einen Befehl (der ein Aufruf eines Programms o.ä. sein kann) zu einem bestimmten Zeitpunkt zu starten. Der Zeitpunkt kann dabei entweder z.B. um die gleiche Uhrzeit am Tag sein oder aber nach dem Hochfahren des Systems.

12. SSL-Zertifikat installieren (Zertifikat mit Letsencrypt)

Etwas mehr Infos über Letsencrypt: https://www.thomas-krenn.com/de/wiki/Let%27s_Encrypt

Um ein SSL / TLS Zertifikat auch sinnvoll nutzen zu können, sollte auf dem Server ein Webserver bzw. eine Anwendung, zu welcher z.B. eine HTTP(S) Verbindung aufgebaut werden soll, laufen.

Darüber hinaus empfehle ich einen dynamischen DNS-Server (Kapitel 9), sofern dein Server keine statische IP-Adresse hat. Normalerweise vergeben Internet Provider an Haushalte nur gegen Aufzahlung eine statische IP. Eine dynamische IP-Adresse kann sich beliebig ändern, das heisst es ändert sich dauernd Ihre öffentliche IP-Adresse, ohne dass Sie diese kennen.

Mit einem dDNS-Service kann dieses Problem umgangen werden, da eine Verbindung zu einer (kostenlosen) Domain aufgebaut wird, sobald sich die dynamische IP-Adresse ändert. So kann man diese Domain anstelle der sich ändernden IP-Adresse aufgerufen werden und der dDNS-Service kümmert sich um alles weitere. Daneben kann deine richtige Domain (sofern eine vorhanden ist) auch per DNS-Einträge auf diese „Zwischen-Domain“ zeigen.

Zum Start müssen wir den Apache-Webserver beenden:

```
sudo service apache2 stop
```

Da wir nun von aussen, also vom öffentlichen Internet, auf den Webserver zugreifen wollen, müssen die Ports 80 (http) und 443 (https) am Router auf Ihren Webserver (interne IP-Adresse) weitergeleitet (Portforwarding) werden. Dazu bitte die Bedienungsanleitung des spezifischen Routers kontrollieren.

Wichtig: Falls sicherheitskritische Anwendungen im Heim-/Firmennetzwerk laufen, zuerst die zuständige Person konsultieren.

Let's Encrypt SSL Zertifikat erstellen

Um das Tool zum Erstellen des Zertifikats herunterzuladen, nutzen wir den Certbot. Dieser installiert sich folgendermassen:

```
sudo apt install certbot python3-certbot-apache
```

Anschliessend werden wir eines der Plugins von Certbot für Apache verwenden:

```
sudo certbot --apache
```

Hier muss unter anderem die Mail-Adresse angegeben werden.

- Mail-Adresse: oliver.macher@bbz-sh.ch
- Terms of Service: a
- Share Mail-Adress: n
- No domains found, enter domain: bbztest.ddns.net
- Redirect HTTP to HTTPS: 2 (redirect)

Danach sollte folgende Bestätigung erscheinen:

```
-----
Congratulations! You have successfully enabled https://bbztest.ddns.net

You should test your configuration at:
https://www.ssllabs.com/ssltest/analyze.html?d=bbztest.ddns.net
-----

IMPORTANT NOTES:
- Congratulations! Your certificate and chain have been saved at:
  /etc/letsencrypt/live/bbztest.ddns.net/fullchain.pem
  Your key file has been saved at:
  /etc/letsencrypt/live/bbztest.ddns.net/privkey.pem
  Your cert will expire on 2020-08-24. To obtain a new or tweaked
  version of this certificate in the future, simply run certbot again
  with the "certonly" option. To non-interactively renew *all* of
  your certificates, run "certbot renew"
- If you like Certbot, please consider supporting our work by:

  Donating to ISRG / Let's Encrypt:  https://letsencrypt.org/donate
  Donating to EFF:                  https://eff.org/donate-le

root@ubuntu:/home/oli#
```

Nun ist das Zertifikat installiert und ein Redirect von http auf https erstellt. Somit wird man nun beim Aufrufen direkt auf die verschlüsselte Domain weitergeleitet. Probieren Sie es aus:

<https://ihredomain.ddns.net>

(Hier im Beispiel: <https://bbztest.ddns.net>)

SSL Zertifikat erneuern

Das Zertifikat von Letsencrypt ist nur 30 Tage gültig. Certbot übernimmt das automatische Erneuern eigentlich schon selbst. Wir überprüfen hier, ob das auch funktioniert.

Mit folgendem Befehl wird überprüft, ob Certbot auch tatsächlich zweimal am Tag aufstartet und das Zertifikat erneuert:

```
sudo systemctl status certbot.timer
```

Als Output sollte sowas ersichtlich sein:

```
root@ubuntu:/home/oli# sudo systemctl status certbot.timer
● certbot.timer - Run certbot twice daily
   Loaded: loaded (/lib/systemd/system/certbot.timer; enabled; vendor preset: enabled)
   Active: active (waiting) since Tue 2020-05-26 17:23:11 CEST; 14min ago
   Trigger: Wed 2020-05-27 05:53:17 CEST; 12h left
   Triggers: ● certbot.service

May 26 17:23:11 ubuntu systemd[1]: Started Run certbot twice daily.
root@ubuntu:/home/oli#
```

Um den Erneuerungsprozess zu überprüfen, kann eine «Trockenübung» gestartet werden:

```
sudo certbot renew --dry-run
```

Wenn alles funktioniert, sollte folgendes ersichtlich sein:

```
-----
Congratulations, all renewals succeeded. The following certs have been renewed:
  /etc/letsencrypt/live/bbztest.ddns.net/fullchain.pem (success)
** DRY RUN: simulating 'certbot renew' close to cert expiry
**          (The test certificates above have not been saved.)
-----
```

13. WordPress installieren

Über die letzten Jahre erfreut sich ein weiteres System besonderer Beliebtheit: WordPress! Immer mehr Websites, private wie kommerzielle, haben das CMS als Motor unter ihrer Haube. WordPress ist eine freie Software, die unter der GNU General Public License (GPLv2) lizenziert wurde, was bedeutet, dass der Quellcode jedermann zur freien Verfügung steht. Laut Entwickler liegt der Fokus des Systems auf leichter Anpassbarkeit, Eleganz, Benutzerfreundlichkeit und auf der Einhaltung von Webstandards.

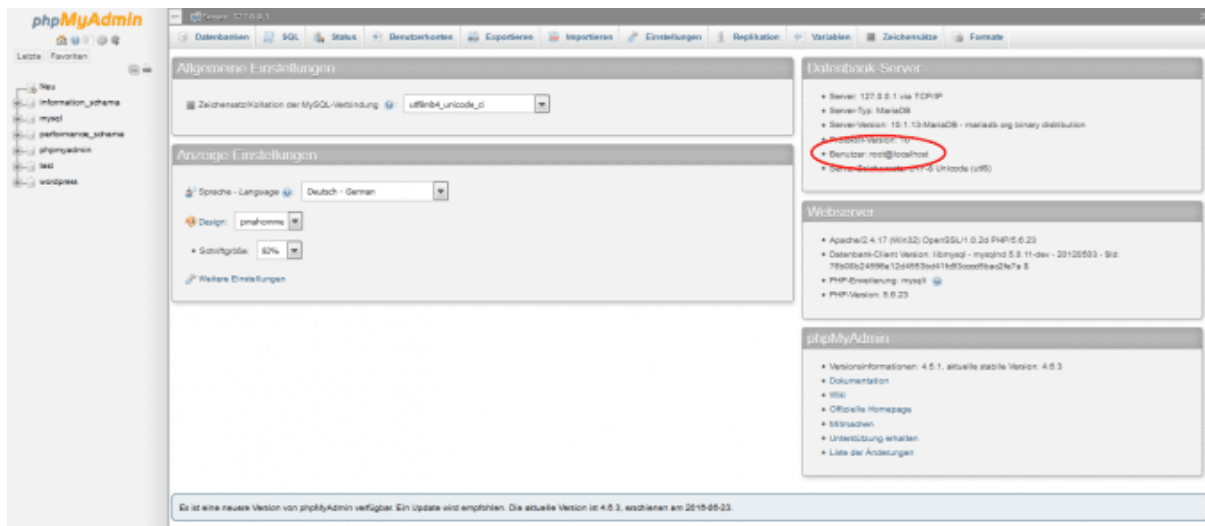
Mittlerweile gibt es eine rege Community die zahlreichen Plug-ins, Erweiterungen und Design-Vorlagen in Form von Templates und Themes entwickelt hat. Damit lassen sich WordPress-basierte Internetseiten und Web-Anwendungen für so ziemlich jeden vorstellbaren Zweck verwirklichen.

Einen Eindruck, was mit WordPress alles möglich ist, bekommt ihr auf der Webseite <http://www.awwwards.com/websites/wordpress/>, die besonders gelungene WordPress-Seiten vorstellt.

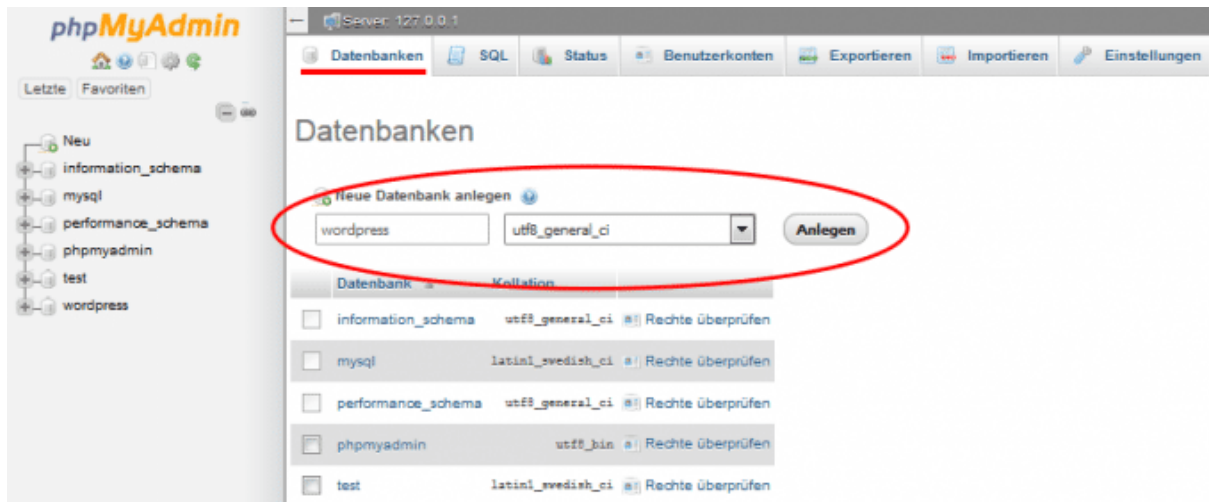
Das technische Grundgerüst für WordPress stellen die Skriptsprache PHP sowie das Datenbank-System MySQL, das alle Beitragsdaten verwaltet. Zum Einsatz wird deshalb ein Webserver benötigt, der als eine Laufzeitumgebung fungiert, die beide Dienste miteinander kommunizieren lässt. Einen solchen Server selbst einzurichten, kann sehr anspruchsvoll sein. Zum Glück gibt es eine komfortable und einsteigerfreundliche Alternative, um so ein System lokal aufzusetzen.

Anlegen einer Datenbank für Wordpress unter phpmyadmin

Da WordPress eine MySQL-Datenbank benötigt, muss diese zuerst angelegt werden: Hierzu im Browser einfach <http://localhost/phpmyadmin/> eingeben und mit dem root einloggen. Dann gelangt man in das phpMyAdmin-Konfigurationsmenü. Rechts neben den allgemeinen Einstellungen sieht man die Webserver-Informationen und die Standard-Benutzerinformationen "Benutzer: root@localhost", welche unter dem Reiter "Benutzerkonten" verwalten und geändert werden können.



Über den Reiter „Datenbanken“ kann nun eine neue Datenbank für WordPress angelegt werden. Es kann einen beliebigen Namen wie beispielsweise „wordpress“ gewählt werden und bei Kollation utf8_general_ci auswählen, damit auch deutsche Sonderzeichen problemlos verwendet werden können.



Installation und Einrichtung von Wordpress

Zum Download der Datei:

```
wget https://de.wordpress.org/latest-de_DE.tar.gz
```

Das File wird direct in den Ordner gespeichert, wo man sich befindet. Falls es noch nicht am richtigen Ort ist, folgendermassen verschieben:

```
sudo mv latest-de_DE.tar.gz /var/www/html
```

Danach kann das File entpackt werden. Mit folgendem Befehl wird das Archiv in einen Ordner «wordpress» extrahiert:

```
cd /var/www/html  
sudo tar -xf latest-de_DE.tar.gz
```

Nach dem Entpacken der Dateien kann die Readme auf dem Webserver mit dem Explorer angeschaut werden:

[http://\[IP\]/wordpress/readme.html](http://[IP]/wordpress/readme.html)

oder wenn es direkt auf dem Ubuntu-Server aufgerufen werden soll:

<http://localhost/wordpress/readme.html>

Einrichten von Wordpress

Es öffnet sich eine Webseite mit Informationen zur "berühmten" 5-Minuten-Installation. Es kann nun jeder Schritt durchgegangen werden.

Willkommen bei WordPress. Bevor wir anfangen, brauchen wir einige Informationen zur Datenbank. Folgende Daten werden benötigt:

1. Datenbank-Name
2. Datenbank-Benutzername
3. Datenbank-Passwort
4. Datenbank-Host
5. Tabellen-Präfix (falls du mehrere WordPress-Installationen innerhalb einer Datenbank aufbauen möchtest)

Diese Informationen werden für die Erstellung der Datei `wp-config.php` genutzt. **Sollte die automatische Erstellung dieser Datei aus irgendeinem Grund nicht funktionieren, keine Sorge. Es werden lediglich Datenbank-Informationen in einer Konfigurationsdatei gespeichert. Alternativ öffnest du die Datei `wp-config-sample.php` einfach in einem Texteditor, ergänzt die notwendigen Informationen und speicherst die Datei als `wp-config.php`.** Du benötigst mehr Hilfe? Dann bitte [hier entlang](#).

Wahrscheinlich kannst du diese Informationen in deinem Webhosting-Konto finden. Wenn du sie nicht parat hast, kontaktiere die Firma, bei der deine Website gehostet wird, bevor du weitermachst.

Los geht's!

Datenbank-Name	wordpress
Datenbank-Benutzername	Benutzer, der bei PHPMyAdmin erstellt wurde (hier: <code>mysql_admin</code>).
Datenbank-Passwort	Passwort, das dem Benutzer <code>mysql_admin</code> gegeben wurde
Datenbank-Host	localhost
Tabellen-Präfix	<code>wp_</code>

Sollte die `wp-config` nicht beschreibbar sein, allen Usern für den Ordner «wordpress» Schreibrechte einräumen:

```
cd /var/www/html
sudo chmod 777 wp-config-sample.php
sudo chmod 777 /var/www/html/wordpress
```

Danach Wordpress Installation nochmals starten und durchführen.



Alles klar! Diesen Teil der Installation hast du geschafft. WordPress kann jetzt mit deiner Datenbank kommunizieren. Wenn du bereit bist, kannst du jetzt die ...

Installation durchführen

Danach muss noch der Titel und der Administrator der Wordpress-Seite bestimmt werden.



Willkommen

Willkommen bei der berühmten 5-Minuten-Installation von WordPress! Gib unten einfach die benötigten Informationen ein und schon kannst du starten mit der am besten erweiterbaren und leistungsstarken persönlichen Veröffentlichungsplattform der Welt.

Benötigte Informationen

Bitte trage die folgenden Informationen ein. Keine Sorge, du kannst all diese Einstellungen später auch wieder ändern.

Titel der Website


Benutzername
Benutzernamen dürfen nur alphanumerische Zeichen, Leerzeichen, Unterstriche, Bindestriche, Punkte und das @-Zeichen enthalten.

Passwort
Wichtig: Du wirst dieses Passwort zum Anmelden brauchen. Bitte bewahre es an einem sicheren Ort auf.

Deine E-Mail-Adresse
Bitte überprüfe nochmal deine E-Mail-Adresse auf Richtigkeit, bevor du weitermachst.

Sichtbarkeit für Suchmaschinen Suchmaschinen davon abhalten, diese Website zu indizieren.
Es ist Sache der Suchmaschinen, dieser Bitte nachzukommen.

Danach auf «WordPress installieren» klicken. Nach dem Anmelden kann man dann im Browser auf den Entwicklerbereich (Backend) zugreifen und sieht die entsprechende Version der Webseite (Frontend).



Installation erfolgreich!

WordPress wurde installiert. Vielen Dank, und nun viel Spaß!

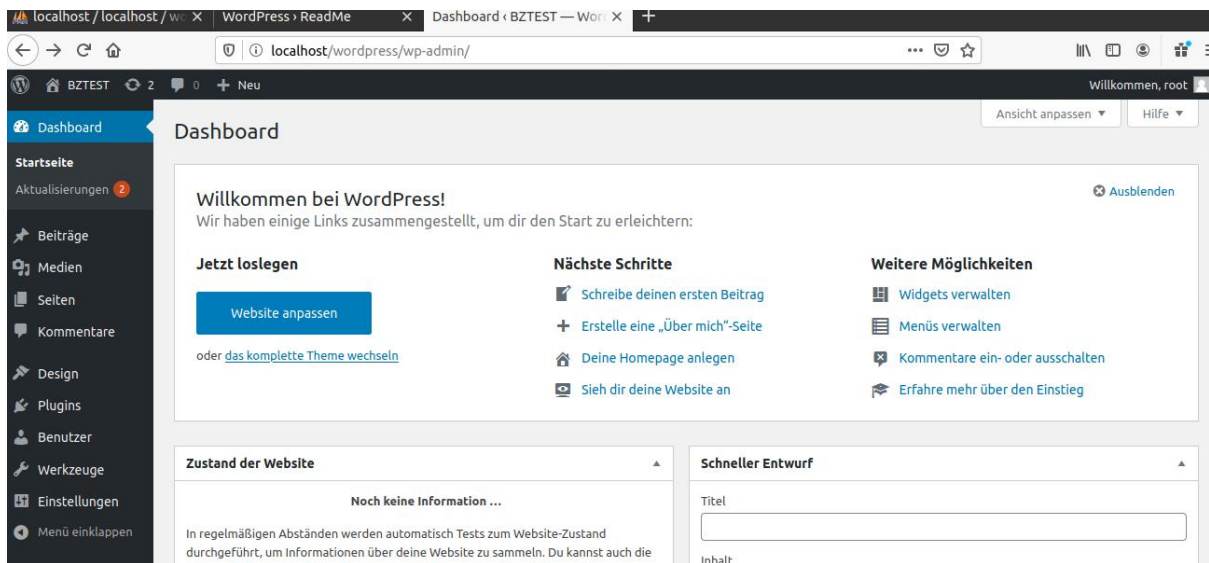
Benutzername root

Passwort *Das von dir gewählte Passwort.*

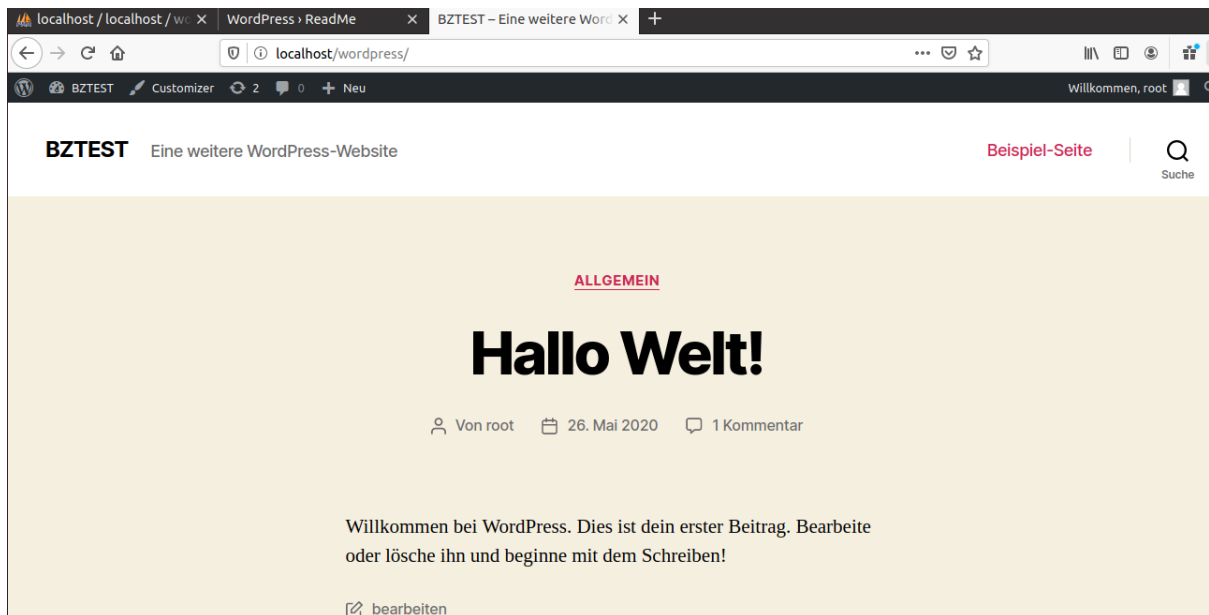
Nun unbedingt wieder die Rechte zurücksetzen. Dabei ist es sehr wichtig, dass alle WP-Ordner und Subdirectories 744 oder 755 haben und alle Files in den Ordnern und Subdirectories 644 oder 640.

```
cd /var/www/html/wordpress  
  
sudo find /var/www/html/wordpress -type d -exec chmod 755 {} \;  
  
sudo find /var/www/html/wordpress -type f -exec chmod 644 {} \;
```

Nach erfolgreicher Installation und Login mit dem Adminkonto ist das Backend der Webseite ersichtlich:



Ein Klick oben Links auf «BBZTEST» öffnet das Frontend:



Jetzt kann das System erforscht werden, Themes geändert, Beiträge verfasst und eine persönliche WordPress-Webseite nach eigenen Wünschen und Ideen eingerichtet werden. Viel Spass dabei!

Weitere Informationen finden sich auf der deutschen WordPress-Supportseite:

<https://de.wordpress.org/hilfe/>.

Hier ein deutschsprachiges Tutorial:

<https://wp-wizard.de/tutorial/>

DocumentRoot des Apache-Webservers so anpassen, dass direkt über die URL von Noip zugegriffen werden kann.

Wenn Sie nicht immer `https://www.bbzttest.ddns.net/wordpress`, sondern nur noch mit `https://www.bbzttest.ddns.net/` auf Ihre WordPress-Seite zugreifen wollen, muss der Document Root des Apache-Servers angepasst werden. Das bedeutet, dass Ihr Server nachher standardmässig nicht mehr auf `/var/www/html`, sondern auf `/var/www/html/wordpress` zugreift.

Dazu müssen folgende Schritte erledigt werden:

Zuerst muss der DocumentRoot bei den http-Seiten (Port 80) angepasst werden. Dazu folgende Datei öffnen und beim Eintrag DocumentRoot den Eintrag `/var/www/html/wordpress` angeben:

```
sudo nano /etc/apache2/sites-available/000-default.conf
```

Das gleiche nun noch beim DocumentRoot der https-Seiten (Port 443). Dazu folgende Datei öffnen und beim Eintrag DocumentRoot den Eintrag `/var/www/html/wordpress` angeben:

```
sudo nano /etc/apache2/sites-available/000-default-le-ssl.conf
```

Bei einer frischen Installation kann nun noch der Apache-Server neu gestartet werden und das wars.

```
sudo service apache2 restart
```

Sollte schon an der `wp-config.php` mit der Home- und Site-URL gearbeitet worden sein, muss der Eintrag wieder auf die Original-DDNS weisen:

```
define('WP_HOME', 'https://bbzttest.ddns.net/');  
define('WP_SITEURL', 'https://bbzttest.ddns.net/');
```

Danach sollte der Zugriff direkt über Ihre entsprechende Domain von noip sichergestellt sein.

14. Webserver-Verzeichnis mit Passwort schützen

Erstellen einer .passwd Datei

Damit man mit dem Apache Webserver Verzeichnisse und Dateien mit einem Passwort schützen kann, braucht man zunächst eine Datei, in welcher die Passwortdaten enthalten sind. Dazu erstellt man sich die Datei am besten nicht im Document-Root des Webservers (z.B. /var/www/html; hier könnte die Datei über das Internet ausgelesen werden), sondern im /root – Verzeichnis oder Benutzerverzeichnis. Wir wollen jetzt einfach zum Spass einen Ordner machen, den wir schützen wollen:

```
sudo mkdir /var/www/html/locked  
sudo htpasswd -c /etc/apache2/.htpasswd webserv_user
```

Das Flag -c bewirkt dabei, dass eine neue Datei angelegt wird. Wenn die Aktion erfolgreich war, sollte ein Passwort eingegeben werden müssen:

```
bl@ubuntu:~/webserv_pw$ sudo htpasswd -c /etc/apache2/.htpasswd webserv_user  
New password:  
Re-type new password:  
Adding password for user webserv_user
```

Man kann sich die Datei z. B. mit dem Befehl nano ansehen:

```
sudo nano /etc/apache2/.htpasswd
```

Und sieht dann folgendes:

```
GNU nano 4.8 /etc/apache2/.htpasswd  
webserv_user:$apr1$AUwX6En9$gzd6j6YQEI3v3bwZuba2V0
```

Erstellen der .htaccess Datei

Um Verzeichnisse und Dateien im Apache Webserver mit einem Passwortschutz zu versehen kann man sich im entsprechenden Verzeichnis eine .htaccess Datei erstellen (z. B. mit dem Editor nano), welche dann einen Passwortschutz ermöglicht. Im folgenden wird davon ausgegangen, dass das im Document-Root des Webservers befindliche Verzeichnis «locked» mit einem Passwort geschützt werden soll.

```
cd /var/www/html/locked  
sudo nano .htaccess
```

Die Datei muss nachher folgendermassen erweitert werden:

```
AuthType Basic  
AuthUserFile /etc/apache2/.htpasswd  
AuthName "Geheimes Webserverteildings"  
order deny,allow  
allow from all  
require valid-user
```

Die Zeile AuthUserFile gibt dabei an, wo sich die .passwd Datei befindet, anhand der sich Benutzer bei der Anmeldung authentifizieren können. Die Zeile require valid-user ermöglicht eine Angabe, wer alles Zugriff auf die Verzeichnisse und Dateien erhalten soll. Dabei gibt man mit valid-user an, dass alle in der .passwd Datei angelegten Benutzer Zugriff auf die Verzeichnisse und Dateien erhalten. (Sollen nur bestimmte Benutzer Zugriff erhalten kann man dies hier angeben, z. B. mit require testuser)

Anpassen der Datei mit dem VirtualHost

Damit die Einstellungen auch wirksam werden, muss noch die Option AllowOverride von None auf All in der Datei /etc/apache2/apache2.conf eingefügt werden. Wenn nun also das Verzeichnis /var/www/html/geheim/ geschützt werden soll, muss dort wo die anderen Directory-Directives in apache2.conf sind folgendes eingefügt werden (dort wo es ähnlich aussieht, wie das, was wir einfügen).

Wichtig: Erstellen Sie einen neuen Eintrag, modifizieren Sie keinen bestehenden!

```
sudo nano /etc/apache2/apache2.conf

#Oli 2020-05-26 fuer geschuetztes Verzeichnis
<Directory /var/www/html/locked/>
    AllowOverride All
</Directory>
```

Nun kopieren wir noch das Apache-Startfile in den Ordner, damit wir auch etwas sehen können:

```
cd /var/www/html
sudo cp index.html locked/
```

Übernehmen der Einstellungen

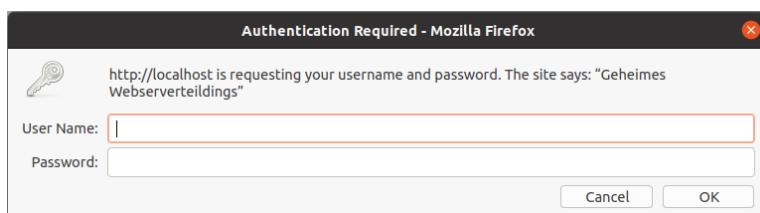
Damit die Einstellung vom Webserver übernommen werden, muss dieser abschliessend nur noch neu gestartet werden.

```
sudo /etc/init.d/apache2 restart
```

Ergebnis

Will man nun Zugriff auf die Website erscheint ein Fenster zur Authentifizierung.

<http://localhost/locked/>



Fehlerhafte Logins in Log anzeigen

Man kann in den Logs nachschauen, wer sich versucht hat einzuloggen:

```
nano /var/log/apache2/access.log
```

```
127.0.0.1 - websrv_user [26/May/2020:20:22:34 +0200] "GET /locked/ HTTP/1.1" 200 3482 "-" "Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:76.0) Gecko/20100101 Firefox/76.0"
127.0.0.1 - - [26/May/2020:20:22:34 +0200] "GET /icons/ubuntu-logo.png HTTP/1.1" 200 3623 "http://localhost/locked/" "Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:76.0) G
127.0.0.1 - - [26/May/2020:20:22:34 +0200] "GET /favicon.ico HTTP/1.1" 404 487 "-" "Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:76.0) Gecko/20100101 Firefox/76.0"
```

Hier ist ersichtlich, dass sich websrv_user mit der IP-Adresse zur angezeigten Zeit erfolgreich eingeloggt hat.

```
nano /var/log/apache2/error.log
```

```
[Tue May 26 20:22:30.217824 2020] [auth_basic:error] [pid 20425] [client 127.0.0.1:58898] AH01618: user halloBBZLeuts not found: /locked/
^G Get Help      ^O Write Out    ^W Where Is     ^K Cut Text      ^J Justify      ^C Cur Pos      M-U Undo       M-A Mark Text    M-T To Bracket
^X Exit          ^R Read File    ^_ Replace      ^U Paste Text   ^T To Spell     ^_ Go To Line   M-E Redo       M-G Copy Text  ^Q Where Was
```

Hier sieht man im error.log, dass sich um 20:22:30 Uhr am 26.05.2020 ein User mit der IP-Adresse 127.0.0.1 (weil ich von lokal zugreifen wollte) mit dem Benutzer «halloBBZLeuts» einloggen wollte, dies aber verwehrt wurde.

15. Virtuelle Hosts hinzufügen

Mit virtuellen Hosts können mehr als nur eine Domain auf einem einzigen Server verfügbar gemacht werden. Als Beispiel wird die Domain bbz-testdomain.com genommen. Ersetzen Sie diese mit einer eigenen.

Es soll der Pfad bbztest2.com erstellt werden. Verwenden sie das -p Flag um notwendige Parent-Directories zu erzeugen:

```
sudo mkdir -p /var/www/bbztest2.com/html
```

Mit folgendem Befehl wird der Eigentümer hinzugefügt:

```
sudo chown -R $USER:$USER /var/www/bbztest2.com/
```

Konfigurieren der Berechtigungen:

```
sudo chmod -R 755 /var/www/bbztest2.com
```

Erstelle eine Sample – index.html mit nano.

```
sudo nano /var/www/bbztest2.com/html/index.html
```

Füge folgende HTML-Befehle hinzu:

```
<html>
  <head>
    <title>Welcome to bbztest2.com!</title>
  </head>
  <body>
    <h1>Success! The bbztest2.com server block is working!</h1>
  </body>
</html>
```

Nun muss noch ein neues virtual Host – File erstellt werden:

```
sudo nano /etc/apache2/sites-available/bbztest2.com.conf
```

Folgendes soll im File stehen:

```
<VirtualHost *:80>
  ServerAdmin oliver.macher@bbz-sh.ch
  ServerName bbztest2.com
  ServerAlias www.bbztest2.com
  DocumentRoot /var/www/bbztest2.com/html
  ErrorLog ${APACHE_LOG_DIR}/error.log
  CustomLog ${APACHE_LOG_DIR}/access.log combined
</VirtualHost>
```

File speichern und schliessen.

Danach muss das File mit a2ensite aktiviert werden:

```
sudo a2ensite bbztest2.com.conf
```

```
oli@ubuntu:/var/log/apache2$ sudo a2ensite bbztest2.com.conf
Enabling site bbztest2.com.
To activate the new configuration, you need to run:
systemctl reload apache2
```

Deaktiviere die Default-Page, welche in 000-default.conf definiert ist:

```
sudo a2dissite 000-default.conf
```

Teste ob Konfigurationsfehler vorhanden sind:

```
sudo apache2ctl configtest
```

Restart den Apache-Server um Änderungen anzunehmen:

```
sudo systemctl restart apache2
```

Nun sollte man mit localhost die neue Page aufrufen können:



Wieder die Default-Page anzeigen

Mit folgendem Befehl kann wieder die Default-Page angezeigt werden:

```
sudo a2ensite 000-default.conf
```

```
sudo systemctl restart apache2
```

16. Mails mit ext. Host (GMAIL) über Webserver versenden

Leider ist das Versenden von E-Mails vom eigenen Server aus in Zeiten von Spam nicht einfach. Möchte man einen eigenen SMTP-Server betreiben, müsste man recht viele Anti-Spam-Massnahmen implementieren, um zu gewährleisten, dass die Emails auch beim Empfänger ankommen. Dieser Aufwand ist aber oftmals viel zu hoch.

Daher sollte man externe SMTP-Server für den E-Mail-Versand nutzen. Die Einrichtung ist einfach und sofern man einen vertrauenswürdigen Anbieter, wie Gmail nutzt, hat man keine Probleme mit Spam.

Es soll nun Ubuntu so konfiguriert werden, dass vom Server aus E-Mails über einen externen SMTP-Server (Gmail) versendet werden können. Anschliessend soll noch veranschaulicht werden, wie PHP so konfiguriert wird, das E-Mails die per mail()-Befehl gesendet werden über den SMTP-Server versendet werden können.

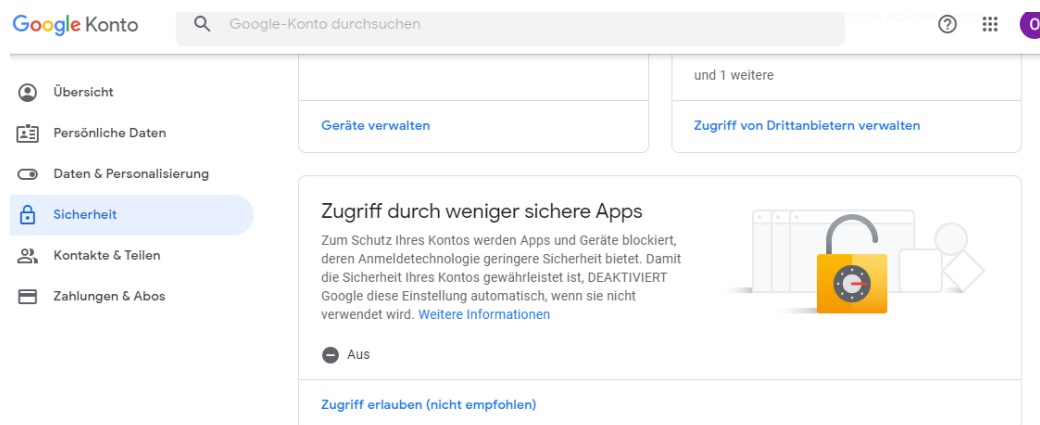
Schritt 1 – Externer SMTP-Server (z.B. Gmail)

Es muss nun also ein kostenloses [Gmail](#)-Konto erstellt werden. Damit der Ubuntu-Server die E-Mails über diesen SMTP-Server versendet, braucht es folgende Infos, welche man direkt vom Anbieter bekommt:

- Serveradresse (des SMTP-Servers)
- Benutzername
- Passwort

GMail – Weniger sichere Apps zulassen

Der Versand per Gmail funktioniert nur, wenn im Gmail-Konto die Einstellung «Weniger sichere Apps zulassen» aktiviert wird. Aktiviert man diese Einstellung, dann können externe Anwendungen (sprich, unser Ubuntu-Server) per Benutzername und Passwort über das Gmail-Konto E-Mails versenden.



Wenn diese Einstellung nicht aktiviert wird, erhält man die Fehlermeldung:

```
msmtp: authentication failed (method LOGIN)...
msmtp: server message: 534-5.7.14 your web browser and then try again.
msmtp: server message: 534-5.7.14 Learn more at
msmtp: server message: 534 5.7.14 https://support.google.com/mail/answer/78754 z4-
v6sm17243464wrt.13 - gsmtp
msmtp: could not send mail (account default from /etc/msmtprc)
```

Wie man die Einstellung weniger sichere Apps zulassen kann, ist hier beschrieben:

[Weniger sicheren Apps den Zugriff auf Ihr Konto gestatten](#)

Schritt 2 – MSMTTP installieren

Wir nutzen das msmtpp Package, welches es erlaubt das unter Ubuntu z.B. per Kommandozeile oder per PHP-Anwendung E-Mails über einen externen SMTP-Server versendet werden kann. Die Installation erfolgt wie folgt:

```
sudo apt install -y msmtpp msmtpp-mta
```

Schritt 3 – MSMTTP konfigurieren

Mit `msmtpp --version` kann herausgefunden werden, wo die Config-Files gespeichert sind. Beide Dateien sind nach einer neuen Installation nicht vorhanden (so dass man diese einfach nur auf die eigenen Bedürfnisse abändern könnte), allerdings gibt es auf der Website von msmtpp eine [Beispiel-Konfiguration](#). Um MSMTTP zu konfigurieren, muss die Datei `/etc/msmtpprc` bearbeitet werden:

```
sudo nano /etc/msmtpprc
```

Diese befüllen wir mit folgenden Informationen (Achtung, es müssen noch Daten ausgefüllt werden):

```
# Set default values for all following accounts.
defaults

# Use the mail submission port 587 instead of the SMTP port 25.
port 587

# Always use TLS.
tls on

# Set a list of trusted CAs for TLS. The default is to use system settings, but
# you can select your own file.
tls_trust_file /etc/ssl/certs/ca-certificates.crt

# If you select your own file, you should also use the tls_crl_file command to
# check for revoked certificates, but unfortunately getting revocation lists and
# keeping them up to date is not straightforward.
#tls_crl_file ~/.tls-crls

# Mail account
# TODO: Use your own mail address
account bob@meindedomain.de

# Host name of the SMTP server
# TODO: Use the host of your own mail account
host smtp.meindedomain.de

# As an alternative to tls_trust_file/tls_crl_file, you can use tls_fingerprint
# to pin a single certificate. You have to update the fingerprint when the
# server certificate changes, but an attacker cannot trick you into accepting
# a fraudulent certificate. Get the fingerprint with
# $ msmtpp --serverinfo --tls --tls-certcheck=off --host=smtp.freemail.example
#tls_fingerprint 00:11:22:33:44:55:66:77:88:99:AA:BB:CC:DD:EE:FF:00:11:22:33

# Envelope-from address
# TODO: Use your own mail address
from bob@meindedomain.de

# Authentication. The password is given using one of five methods, see below.
auth on

# TODO: Use your own user name fpr the mail account
user bob@meindedomain.de

# Password method 1: Add the password to the system keyring, and let msmtpp get
# it automatically. To set the keyring password using Gnome's libsecret:
```

```

# $ secret-tool store --label=msmtp \
#   host smtp.freemail.example \
#   service smtp \
#   user joe.smith

# Password method 2: Store the password in an encrypted file, and tell msmtp
# which command to use to decrypt it. This is usually used with GnuPG, as in
# this example. Usually gpg-agent will ask once for the decryption password.
#passwordeval gpg2 --no-tty -q -d ~/.msmtp-password.gpg

# Password method 3: Store the password directly in this file. Usually it is not
# a good idea to store passwords in plain text files. If you do it anyway, at
# least make sure that this file can only be read by yourself.
# TODO: Use the password of your own mail account
password pAssW0Rd123

# Password method 4: Store the password in ~/.netrc. This method is probably not
# relevant anymore.

# Password method 5: Do not specify a password. Msmtplib will then prompt you for
# it. This means you need to be able to type into a terminal when msmtp runs.

# Set a default account
# TODO: Use your own mail address
account default: bob@meinedomain.de

# Map local users to mail addresses (for crontab)
aliases /etc/aliases

```

- *account* Gmail-Account (bla@gmail.com)
- *host* smtp.gmail.com
- *from* die Absendeadresse (beliebig)
- *user* Username von Gmail (bla@gmail.com)
- *password* Passwort von Gmail

Noch ein Hinweis auf die Authentifizierungs-Methode: In diesem Beispiel nutzen wird die einfachste Form der Authentifizierung, indem das Passwort des Mail-Accounts direkt in der Konfiguration gespeichert wird. `msmtp` unterstützt hier allerdings auch weitere Authentifizierungs-Methoden. Mehr Informationen dazu findet man in der Dokumentation zu `msmtp`.

Als nächstes sorgen wir dafür, dass nicht jeder Zugriff auf die Datei hat (besonders wichtig, wenn das Passwort direkt in der Konfiguration gespeichert ist):

```
sudo chmod 600 /etc/msmtp.rc
```

Zum Schluss wird noch ein Alias angelegt, so dass die Empfänger-Adresse des Root-Accounts (oder des Accounts, mit dem später E-Mail versendet werden sollen) bekannt ist. Dies wird in der Datei angegeben, die auch schon ganz am Ende der Konfiguration von `msmtp` aufgeführt wurde:

```
sudo nano /etc/aliases
```

Hier wird nun die Empfänger-Adresse des Root-Accounts angegeben. An diese Adresse werden nun E-Mails verschickt, wenn z.B. ein Cronjob fehlschlagen sollte. Daneben wird noch eine allgemeine „Fallback-Empfänger-Adresse“ angegeben, falls System-Meldungen nicht im Kontext des Root-Accounts auftreten:

```
root: admin@meinedomain.de
default: admin@meinedomain.de
```


Schritt 4 – E-Mail Versand testen

Mit folgendem Befehl kann man nun die richtige Konfiguration von msmtptesten. Sofern alles funktioniert hat, sollte test@email.com eine E-Mail erhalten:

```
sudo su  
echo "Test Nachricht von meinem Server" | msmtptest@email.com
```

Schritt 5 – PHP für den E-Mail-Versand konfigurieren

MSMTP lässt sich verwenden, damit PHP-Anwendungen wie z.B. WordPress mittels dem PHP-Mail-Befehl E-Mails versenden kann. Hierzu müsst ihr die php.ini-Datei anpassen.

Den Pfad der php.ini-Datei lässt sich aus der [phpinfo.php](#) Datei entnehmen (kann auch eine andere Versionsnummer als 7.4 sein, es müssen bereits Daten im File geschrieben sein).

```
sudo nano /etc/php/7.4/apache2/php.ini
```

Nun muss in dieser Datei nach `sendmail_path` gesucht werden und diesen Abschnitt wie folgt angepasst werden:

```
[mail function]  
; For Win32 only.  
;SMTP = localhost  
;smtp_port = 25  
  
; For Win32 only.  
;sendmail_from = me@example.com  
  
; For Unix only. You may supply arguments as well (default: "sendmail -t -  
i").  
sendmail_path = "/usr/bin/msmtptest -t"
```

Die Einstellungen SMTP und `smtp_port` müssen auskommentiert werden, indem ein `;` an den Satzanfang gestellt wird.

Danach muss man `sendmail_path` auf `/usr/bin/msmtptest -t` setzen.

```
sudo systemctl restart apache2
```

Anschliessend muss überprüft werden, dass in der `phpinfo.php` - Datei die Einstellung `sendmail_path` korrekt auf `/usr/bin/msmtptest -t` verweist:

<code>sendmail_from</code>	<code>no value</code>	<code>no value</code>
<code>sendmail_path</code>	<code>/usr/bin/msmtptest -t</code>	<code>/usr/bin/msmtptest -t</code>

17. Open-Source Log-Analyzer AWstats installieren

Wir wollen die Log-Dateien unseres Webserver etwas aufhübschen. Das machen wir mit dem Open-Source Log analyzer AWstats (<https://awstats.sourceforge.io/>).

Zuerst installieren wir ihn mit folgendem Befehl:

```
sudo apt install awstats libgeo-ip-perl libgeo-ipfree-perl
```

Danach müssen wir ihn noch konfigurieren. Zuerst müssen wir eine Kopie von awstats.conf für unsere Domain machen. Dies werden wir mit folgendem Befehl machen:

```
sudo cp /etc/awstats/awstats.conf /etc/awstats/awstats.example.com.conf
sudo nano /etc/awstats/awstats.example.com.conf
```

In diesem neuen File müssen wir nun einige Änderungen vornehmen. Suchen Sie die entsprechenden Einträge und passen sie diese genauso an, wie sie hier aufgelistet sind:

```
LogFormat=1
SiteDomain="example.com"
HostAliases="localhost 127.0.0.1 example.com"
AllowFullYearView=3
AllowAccessFromWebToFollowingIPAddresses="127.0.0.1 192.168.1.0-192.168.1.255"
LoadPlugin="tooltips"
LoadPlugin="graphgooglechartapi"
LoadPlugin="geoipfree"
```

Speichern und Schliessen Sie nun das File. Wir müssen nun noch folgende Datei mit entsprechendem Befehl anpassen:

```
sudo nano /etc/awstats/awstats.conf.local
```

Stellen Sie sicher, dass folgende Einträge vorhanden sind:

```
SiteDomain="example.com"
HostAliases="localhost 127.0.0.1 example.com"
```

File wieder speichern und schliessen.

Nun müssen wir noch einige alten Daten entfernen und Berechtigungen vergeben:

```
sudo mv /etc/cron.d/awstats /root
sudo rm /var/lib/awstats/*
```

```
sudo chgrp www-data /var/log/apache2 /var/log/apache2/*log
/var/log/apache2/access.log
```

```
sudo chmod 755 /var/log/apache2
sudo chmod 644 /var/log/apache2/*
```

Nun müssen wir noch folgendes File bearbeiten:

```
sudo nano /etc/logrotate.d/apache2
```

Suchen Sie die Zeile, welche mit «create 640» startet und überprüfen Sie, dass diese genauso aussieht:

```
create 640 root www-data
```

Speichern und Schliessen Sie das File.

Nun müssen wir das Cronjob-File wieder zurückkopieren:

```
sudo mv /root/awstats /etc/cron.d
```

Zuletzt müssen wir noch Apache anpassen. Passen Sie die Werte Ihrem Webserver entsprechend an. Wechseln Sie zu diesem File:

```
sudo nano /etc/apache2/sites-available/000-default.conf
```

Fügen Sie dort am Schluss vor </VirtualHost> folgendes ein:

```
Alias /awstatsclasses "/usr/share/awstats/lib/"
Alias /awstats-icon/ "/usr/share/awstats/icon/"
Alias /awstatscss "/usr/share/doc/awstats/examples/css"
ScriptAlias /cgi-bin/ /usr/lib/cgi-bin/
ScriptAlias /awstats/ /usr/lib/cgi-bin/
Options +ExecCGI -MultiViews +SymLinksIfOwnerMatch
```

Starten Sie dann das CGI-Modul:

```
sudo a2enmod cgi
```

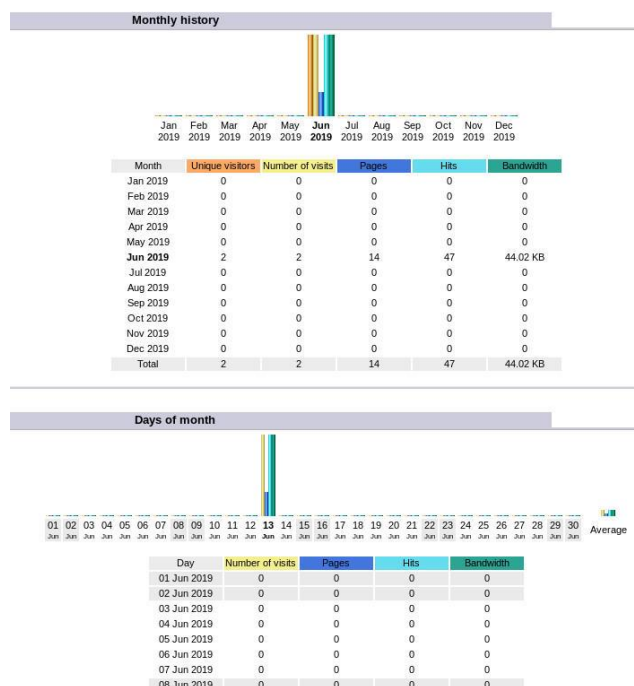
Starten Sie dann noch den Webserver neu:

```
sudo systemctl restart apache2
```

Sollte es dabei zu einer Fehlermeldung kommen, ist die Chance gross, dass sie im 000-default.conf einen Schreibfehler drin haben oder einen Backslash vergessen haben. Überprüfen Sie dies.

Danach kann der Browser geöffnet und die Auswertedatei angezeigt werden:

<http://localhost/cgi-bin/awstats.pl>



18. Berechtigungen in Ubuntu

Die Berechtigungen können in Ubuntu teils recht komplex sein. Hier eine kleine Aufschlüsselung. Dazu eine kleine Warnung:

Man sollte ausserhalb des Benutzerverzeichnisses (und /tmp) keine Zugriffsrechte ändern, sofern man nicht genau weiss, was man tut!

Prinzipiell wird in Linux zwischen

- Besitzer (u)
- Gruppe (g)
- Andere Benutzer (o)

Unterschieden (alle zusammen = a). Diese werden folgendermassen angezeigt:

```
-rwxrwxrwx 1 eigentümer gruppe 0 Sep 20 02:06 datei*
```

Hier haben alle 3 Berechtigungsstufen Leserechte «r», Schreibrechte «w» und Ausführungsrechte «x».

	chmod (octal)	umask (octal)	Symbolisch	Binäre Entsprechung
Lesen, schreiben und ausführen	7	0	rwx	111
Lesen und Schreiben	6	1	rw-	110
Lesen und Ausführen	5	2	r-x	101
Nur lesen	4	3	r--	100
Schreiben und Ausführen	3	4	-wx	011
Nur Schreiben	2	5	-w-	010
Nur Ausführen	1	6	--x	001
Keine Rechte	0	7	---	000

Alle Dateien im aktuellen Verzeichnis mit entsprechenden Rechten anzeigen lassen:

```
ls -al
```

Rechte einer bestimmten Datei anzeigen lassen:

```
ls -al /var/www/html/index.html
```

Mit folgenden Befehlen können die Rechte eines Ordners angepasst werden

```
sudo chmod 755 /var/www/html
```

Mit folgenden Befehlen können die Rechte eines Ordners sowie den Unterordnern angepasst werden

```
sudo chmod 755 -c -R /var/www/html
```

- 19. **TODO: rkhunter****
- 20. **TODO: Monit****

Wir woll

21. Troubleshoot

Login auf FTP-Server funktioniert nicht (Login incorrect)

Lösung (Quelle: <https://www.digitalocean.com/community/questions/proftpd-login-failing-with-530>):

Diese Benutzer existieren als Besitzer bestimmter Dateien oder Prozesse und sind nicht als Anmeldekonto gedacht. Wenn der Wert des Felds "shell" nicht in / etc / shells aufgeführt ist, erlauben Programme wie FTP-Daemons keinen Zugriff. Für Programme, die / etc / shells nicht überprüfen, nutzen sie die Tatsache, dass / bin / false sofort zurückkehrt und eine interaktive Shell ablehnt.

Zuerst sollten wir die shells-Datei erweitern mit /bin/false ganz am Schluss:

```
nano /etc/shells
```

Dann am Schluss folgendes einfügen:

```
/bin/false
```

Danach der Benutzer:

```
useradd USERNAME -d /home/USERNAME -s /bin/false
passwd USERNAME
mkdir /home/USERNAME
chown -R USERNAME:USERNAME /home/USERNAME
```

Danach ein Restart des Proftpd und es sollte funktionieren.

```
sudo /etc/init.d/proftpd restart
```

Bei der Installation von Plugins in WordPress kommt eine Abfrage nach dem FTP-Server – Daten.

Lösung

Hier muss in der Datei wp_config unter dem FTP-Verzeichnis vom WordPress folgender Eintrag gemacht werden:

```
* Diese Datei wird zur Erstellung der wp-config.php verwendet.
* Du musst aber dafür nicht das Installationsskript verwenden.
* Stattdessen kannst du auch diese Datei als wp-config.php mit
* deinen Zugangsdaten für die Datenbank abspeichern.
*
* @package WordPress
*/
define('FS_METHOD', 'direct');
// ** MySQL-Einstellungen ** //
/** Diese Zugangsdaten bekommst du von deinem Webhoster. **/
```

Beim Zugriff auf den Apache-Webserver kommt folgende Meldung: Forbidden You don't have permission to access this resource:



Lösung 1:

Der angemeldete User hat keine Berechtigung auf die Datei zuzugreifen. Daher müssen die Rechte der involvierten Dateien angepasst werden.

```
sudo chmod 755 -c -R /var/www/  
sudo service apache2 restart
```

Mit diesem Befehl geben Sie allen Unterordnern und Dateien die Berechtigung rwx für root, rx für Gruppen und rx für User.

Lösung 2:

Es kann sein, dass die Konfigurationsdatei des Apache so modifiziert wurde, dass es nicht mehr funktioniert (vor allem nach der Erstellung des Passwortgeschützten Verzeichnis tritt dieser Fehler auf). Öffnen Sie das die Konfigurationsdatei des Apache-Servers:

```
sudo nano /etc/apache2/apache2.conf
```

Danach müssen folgende Einträge vorhanden sein:

```
GNU nano 4.8 /etc/apache2/apache2.conf  
# The former is used by web applications packaged in Debian,  
# the latter may be used for local directories served by the w  
# your system is serving content from a sub-directory in /srv  
# access here, or in any related virtual host.  
<Directory />  
    Options FollowSymLinks  
    AllowOverride None  
    Require all denied  
</Directory>  
  
<Directory /usr/share>  
    AllowOverride None  
    Require all granted  
</Directory>  
  
<Directory /var/www/>  
    Options Indexes FollowSymLinks  
    AllowOverride None  
    Require all granted  
</Directory>  
  
#01i 2020-05-26 fuer geschuetztes Verzeichnis  
<Directory /var/www/html/locked/>  
    AllowOverride All  
</Directory>  
  
#<Directory /srv/>  
#    Options Indexes FollowSymLinks  
#    AllowOverride None  
#    Require all granted  
#</Directory>
```

```
sudo service apache2 restart
```

Aufruf von <https://blabla.ddns.org/wordpress> geht, aber bei Anklicken eines Links verweist es mich immer auf localhost/wordpress/irgendwas und alles sieht komisch aus.

Powered by WordPress

Benutzername oder E-Mail-Adresse

Passwort

Angemeldet bleiben

[Passwort vergessen?](#)

[← Zurück zu Gumihans](#)

Lösung:

In den Ordner wordpress/ und dort die Datei wp-config.php öffnen.

```
sudo nano wp-config.php
```

Danach vor dem Eintrag "define('DB_NAME', 'wordpress');" folgendes einfügen:

```
define('WP_HOME', 'https://bbztest.ddns.net/wordpress');  
define('WP_SITEURL', ' https://bbztest.ddns.net/wordpress ');
```

Konfiguration von Apache überprüfen

```
sudo apachectl configtest
```


22. Quellen

<https://tutorials-raspberrypi.de/webserver-installation-apache2/>

<https://tutorials-raspberrypi.de/webserver-installation-homeverzeichnis-aendern/>

<https://tutorials-raspberrypi.de/webserver-installation-php-5/>

<https://tutorials-raspberrypi.de/webserver-installation-teil-3-mysql/>

<https://pimylifeup.com/raspberry-pi-mysql/>

<https://tutorials-raspberrypi.de/webserver-installation-teil-4-phpmyadmin/>

<https://tutorials-raspberrypi.de/webserver-installation-teil-5-ftp-server/>

https://www.thomas-krenn.com/de/wiki/FTP-Server_unter_Debian_einrichten

<https://tutorials-raspberrypi.de/webserver-installation-teil-6-dns-server-via-no-ip/>

<https://tutorials-raspberrypi.de/raspberry-pi-ssl-zertifikat-kostenlos-mit-lets-encrypt-erstellen/>

https://www.thomas-krenn.com/de/wiki/Webserver_Verzeichnisse_mit_Passwort_sch%C3%BCtzen

<https://www.webhosterwissen.de/know-how/eigener-webserver/tutorial-e-mails-mit-eigenem-server-versenden-per-smtp-ubuntu-18-04/>

<https://www.techrepublic.com/article/how-to-install-awstats-on-ubuntu-server-18-04/>

<https://www.digitalocean.com/community/tutorials/how-to-secure-apache-with-let-s-encrypt-on-ubuntu-20-04>

<https://wiki.ubuntuusers.de/chmod/>

<https://wiki.ubuntuusers.de/Rechte/>