

Überschrift

Apache2 und PHP5 sind bereits installiert.

```
apt update
apt install mysql-server -y
<mysql root@localhost Passwort setzen>
mysql_secure_installation
```

- NOTE: RUNNING ALL PARTS OF THIS SCRIPT IS RECOMMENDED FOR ALL MySQL SERVERS IN PRODUCTION USE! PLEASE READ EACH STEP CAREFULLY!

In order to log into MySQL to secure it, we'll need the current password for the root user. If you've just installed MySQL, and you haven't set the root password yet, the password will be blank, so you should just press enter here.

```
Enter current password for root (enter for none):
ERROR 1045 (28000): Access denied for user 'root'@'localhost' (using
password: YES)
```

```
Enter current password for root (enter for none):
OK, successfully used password, moving on...
```

Setting the root password ensures that nobody can log into the MySQL root user without the proper authorisation.

You already have a root password set, so you can safely answer 'n'.

```
Change the root password? [Y/n] n
... skipping.
```

By default, a MySQL installation has an anonymous user, allowing anyone to log into MySQL without having to have a user account created for them. This is intended only for testing, and to make the installation go a bit smoother. You should remove them before moving into a production environment.

```
Remove anonymous users? [Y/n] y
... Success!
```

Normally, root should only be allowed to connect from 'localhost'. This ensures that someone cannot guess at the root password from the network.

```
Disallow root login remotely? [Y/n] y
... Success!
```

By default, MySQL comes with a database named 'test' that anyone can access. This is also intended only for testing, and should be removed before moving into a production environment.

```
Remove test database and access to it? [Y/n] y
```

```
- Dropping test database...
```

```
ERROR 1008 (HY000) at line 1: Can't drop database 'test'; database doesn't exist
```

```
... Failed! Not critical, keep moving...
```

```
- Removing privileges on test database...
```

```
... Success!
```

Reloading the privilege tables will ensure that all changes made so far will take effect immediately.

```
Reload privilege tables now? [Y/n] y
```

```
... Success!
```

Cleaning up...

All done! If you've completed all of the above steps, your MySQL installation should now be secure.

Thanks for using MySQL!

```
mysql -u root -p
CREATE DATABASE wp_waffelente;
CREATE USER wp_waffelente@localhost IDENTIFIED BY 'PASSWORD1';
GRANT ALL PRIVILEGES ON wp_waffelente.* TO wp_waffelente@localhost;
FLUSH PRIVILEGES;
exit
service apache2 restart && service mysql restart

wget https://de.wordpress.org/latest-de_DE.zip
mkdir -p /var/www/html/waffeln-am-ententeich.de/
unzip -q latest-de_DE.zip -d /var/www/html/waffeln-am-ententeich.de/
mv /var/www/html/waffeln-am-ententeich.de/wordpress /var/www/html/waffeln-am-ententeich.de/root
mkdir /var/www/html/waffeln-am-ententeich.de/root/wp-content/uploads
chown -R www-data:www-data /var/www/html/waffeln-am-ententeich.de/root
chmod -R 755 /var/www/html/waffeln-am-ententeich.de/root

cp /var/www/html/waffeln-am-ententeich.de/root/wp-config-sample.php
/var/www/html/waffeln-am-ententeich.de/root/wp-config.php
nano /var/www/html/waffeln-am-ententeich.de/root/wp-config.php
```

[wp-config.php](#)

```
<?php
```

```
define('FORCE_SSL_ADMIN', true);

define('DB_NAME', 'wp_waffelente');
define('DB_USER', 'wp_waffelente');
define('DB_PASSWORD', 'PASSWORD1');
define('DB_HOST', 'localhost');
define('DB_CHARSET', 'utf8');
define('DB_COLLATE', '');

define('AUTH_KEY',          'Füge hier deine Zeichenkette ein');
define('SECURE_AUTH_KEY',  'Füge hier deine Zeichenkette ein');
define('LOGGED_IN_KEY',    'Füge hier deine Zeichenkette ein');
define('NONCE_KEY',        'Füge hier deine Zeichenkette ein');
define('AUTH_SALT',        'Füge hier deine Zeichenkette ein');
define('SECURE_AUTH_SALT', 'Füge hier deine Zeichenkette ein');
define('LOGGED_IN_SALT',   'Füge hier deine Zeichenkette ein');
define('NONCE_SALT',       'Füge hier deine Zeichenkette ein');

$table_prefix = 'wp_';
define('WP_DEBUG', false);
if ( !defined('ABSPATH') )
    define('ABSPATH', dirname(__FILE__) . '/');
require_once(ABSPATH . 'wp-settings.php');
```

```
nano /etc/ssl/certs/waffeln.meinekleinefarm.net.pem
nano /etc/ssl/private/waffeln.meinekleinefarm.net.key
nano /etc/apache2/sites-available/waffeln.meinekleinefarm.net.conf
```

[waffeln.meinekleinefarm.net.conf](#)

```
<IfModule mod_ssl.c>
    <VirtualHost *:443>
        ServerAdmin webmaster@meinekleinefarm.net
        ServerName waffeln.meinekleinefarm.net
        ServerAlias waffeln.meinekleinefarm.net

        DocumentRoot /var/www/html/waffeln-am-ententeich.de/root/

        <Directory /var/www/html/waffeln-am-ententeich.de/root/>
            AllowOverride All
        </Directory>

        ErrorLog ${APACHE_LOG_DIR}/error.log
        CustomLog ${APACHE_LOG_DIR}/access.log combined

        SSLEngine on
        SSLCertificateFile
```

```
/etc/ssl/certs/waffeln.meinekleinefarm.net.pem
    SSLCertificateKeyFile
/etc/ssl/private/waffeln.meinekleinefarm.net.key
    SSLCertificateChainFile
/etc/ssl/certs/startcom_class1.pem

    <FilesMatch "\.(cgi|shtml|phtml|php)$">
        SSLOptions +StdEnvVars
    </FilesMatch>
    <Directory /usr/lib/cgi-bin>
        SSLOptions +StdEnvVars
    </Directory>

    BrowserMatch "MSIE [2-6]" \
        nokeepalive ssl-unclean-shutdown \
        downgrade-1.0 force-response-1.0
    BrowserMatch "MSIE [17-9]" ssl-unclean-shutdown

    </VirtualHost>
</IfModule>
```

```
a2ensite waffeln.meinekleinefarm.net.conf
service apache2 reload
```

<https://www.example.com/wp-admin/install.php> Username Password E-Mail → Fertig!

From:
<https://host.docker.internal:552/> - **Meine kleine Dokumentation**

Permanent link:
https://host.docker.internal:552/doku.php?id=waffeln_am_ententeich&rev=1475706451

Last update: **2016/10/05 22:27**

