

# Checkliste: ownCloud 9 auf Ubuntu Server 16.04 LTS mit nginx, MariaDB, PHP 7 und Let's Encrypt

[decatec.de](http://decatec.de)

Dies ist eine Checkliste ohne weitere Erklärungen zur Installation von ownCloud 9 auf Ubuntu Server 16.04 LTS mit nginx, MariaDB, PHP 7 und Let's Encrypt.

Den kompletten Artikel mit weiteren Erklärungen zu den einzelnen Schritten zur Installation und Konfiguration gibt es unter [decatec.de](http://decatec.de)

Viele Aktionen setzen Root-Rechte voraus, daher geht die Liste davon aus, dass man (temporär) mit dauerhaften Root-Rechten arbeitet (*sudo -s*).

Als Beispiel dient hier die URL *owncloud9tutorial.goip.de*. Diese URL sollte überall durch die tatsächlich eingesetzte URL ausgetauscht werden.

Hinweise zur Formatierung:

Eingaben in der Kommandozeile

Inhalte von Dateien

Hinweis, dass eine Anpassung der Eingaben notwendig ist

- Updates:
  - `apt-get update`
  - `apt-get upgrade`
  - `apt-get dist-upgrade`
  - `reboot`
- Statische IP-Adresse (**hier ins Anpassungen an die eigene Netzwerkkumgebung notwendig**):
  - `nano /etc/network/interfaces`

```
auto eth0
iface eth0 inet static
address 192.168.178.20
netmask 255.255.255.0
network 192.168.178.0
broadcast 192.168.178.255
gateway 192.168.178.1
dns-nameservers 192.168.178.1
```

- nginx installieren:
  - `wget -O - http://nginx.org/keys/nginx_signing.key | apt-key add -`
  - `nano /etc/apt/sources.list`

```
# Nginx (Mainline)
deb http://nginx.org/packages/mainline/ubuntu/ xenial nginx
deb-src http://nginx.org/packages/mainline/ubuntu/ xenial nginx
```

- `apt-get update`
- `apt-get install nginx`
- MariaDB installieren:

- o apt-key adv --recv-keys --keyserver hkp://keyserver.ubuntu.com:80 0xcbc082a1bb943db
- o nano /etc/apt/sources.list

```
# MariaDB 10.1 repository list
# http://mariadb.org/mariadb/repositories/
deb [arch=amd64,i386]
http://ftp.hosteurope.de/mirror/mariadb.org/repo/10.1/ubuntu
xenial main
deb-src
http://ftp.hosteurope.de/mirror/mariadb.org/repo/10.1/ubuntu
xenial main
```

- o apt-get install mariadb-server
- o **Root-Passwort vergeben**
- **PHP installieren:**
  - o apt-get update
  - o apt-get install php7.0-fpm php7.0-gd php7.0-mysql php7.0-curl php7.0-xml php7.0-zip php7.0-intl php7.0-mcrypt php-apcu
- **nginx Konfiguration**
  - o nano /etc/nginx/nginx.conf

```
server_tokens off;
```

- **Default-Seite deaktivieren:**
  - o rm /etc/nginx/sites-enabled/default
- **Vorbereitung Verzeichnis-Struktur:**
  - o mkdir -p /var/www/letsencrypt
  - o mkdir -p /var/www/owncloud
  - o mkdir -p /var/owncloud\_data
  - o chown -R www-data:www-data /var/www
  - o chown -R www-data:www-data /var/owncloud\_data
- **Gateway-Host anlegen:**
  - o nano /etc/nginx/sites-available/owncloud9tutorial.goip.de

```
server {
    listen 80 default_server;
    server_name owncloud9tutorial.goip.de 192.168.178.20;

    root /var/www;

    location ^~ /.well-known/acme-challenge {
        proxy_pass http://127.0.0.1:81;
        proxy_redirect off;
    }
}
```

- **Virtuellen Host für Let's Encrypt anlegen:**
  - o nano /etc/nginx/sites-available/owncloud9tutorial.goip.de\_letsencrypt

```
server {
    listen 81;
    server_name 127.0.0.1;

    location ^~ /.well-known/acme-challenge {
        default_type text/plain;
        root /var/www/letsencrypt;
    }
}
```

- **Virtuelle Hosts aktivieren:**
  - `ln -s /etc/nginx/sites-available/owncloud9tutorial.goip.de /etc/nginx/sites-enabled/owncloud9tutorial.goip.de`
  - `ln -s /etc/nginx/sites-available/owncloud9tutorial.goip.de_letsencrypt /etc/nginx/sites-enabled/owncloud9tutorial.goip.de_letsencrypt`
  - `service nginx restart`
- **SSL-Zertifikat mit Let's Encrypt generieren:**
  - **Port-Forwarding (80, 443) im Router einrichten**
  - **Router am DynDNS-Dienst anmelden**
  - `cd ~`
  - `git clone https://github.com/letsencrypt/letsencrypt`
  - `cd letsencrypt`
  - `sudo -H ./letsencrypt-auto certonly -a webroot --webroot-path=/var/www/letsencrypt --rsa-key-size 4096`
- **Diffie-Hellman-Parameter erzeugen**
  - `mkdir -p /etc/nginx/ssl`
  - `openssl dhparam -out /etc/nginx/ssl/dhparams.pem 2048`
- **Zugriffsrechte anpassen:**
  - `chmod 600 /etc/letsencrypt/live/owncloud9tutorial.goip.de/fullchain.pem`
  - `chmod 600 /etc/letsencrypt/live/owncloud9tutorial.goip.de/privkey.pem`
  - `chmod 600 /etc/letsencrypt/live/owncloud9tutorial.goip.de/chain.pem`
  - `chmod 600 /etc/letsencrypt/live/owncloud9tutorial.goip.de/cert.pem`
  - `chmod 600 /etc/nginx/ssl/dhparams.pem`
- **Gateway-Host für ownCloud erweitern:**

```
server {
    listen 80 default_server;
    server_name owncloud9tutorial.goip.de 192.168.178.20;

    root /var/www;

    location ^~ /.well-known/acme-challenge {
        proxy_pass http://127.0.0.1:81;
        proxy_redirect off;
    }

    # Enforce HTTPS
    return 301 https://$server_addr$request_uri;

    # Use this if you always want to redirect to the DynDNS
    address (no local access).
    #return 301 https://$server_name$request_uri;
}

server {
    listen 443 ssl http2;
```

```

server_name owncloud9tutorial.goip.de 192.168.178.20;

#
# Configure SSL
#
ssl on;

# Certificates used
ssl_certificate
/etc/letsencrypt/live/owncloud9tutorial.goip.de/fullchain.pem;
ssl_certificate_key
/etc/letsencrypt/live/owncloud9tutorial.goip.de/privkey.pem;

# Not using TLSv1 will break:
#   Android <= 4.4.40
#   IE <= 10
#   IE mobile <=10
# Removing TLSv1.1 breaks nothing else!
ssl_protocols TLSv1.2;

# 100 % Security
# Low Compatibility
#   No Android 2
#   No Java
#   No IE < 11 (XP)
# Robust Forward Secrecy
#
# Remarks: This won't work on Chrome
#ssl_ciphers 'AES256+EECDH:AES256+EDH:!aNULL';

# Using the recommended cipher suite from:
https://wiki.mozilla.org/Security/Server\_Side\_TLS
ssl_ciphers 'ECDHE-RSA-AES128-GCM-SHA256:ECDHE-ECDSA-
AES128-GCM-SHA256:ECDHE-RSA-AES256-GCM-SHA384:ECDHE-ECDSA-
AES256-GCM-SHA384:DHE-RSA-AES128-GCM-SHA256:DHE-DSS-AES128-GCM-
SHA256:kEDH+AESGCM:ECDHE-RSA-AES128-SHA256:ECDHE-ECDSA-AES128-
SHA256:ECDHE-RSA-AES128-SHA:ECDHE-ECDSA-AES128-SHA:ECDHE-RSA-
AES256-SHA384:ECDHE-ECDSA-AES256-SHA384:ECDHE-RSA-AES256-
SHA:ECDHE-ECDSA-AES256-SHA:DHE-RSA-AES128-SHA256:DHE-RSA-
AES128-SHA:DHE-DSS-AES128-SHA256:DHE-RSA-AES256-SHA256:DHE-DSS-
AES256-SHA:DHE-RSA-AES256-
SHA:!aNULL:!eNULL:!EXPORT:!DES:!RC4:!3DES:!MD5:!PSK';

# Nginx for Modern Browsers (uncomment this when the
other ssl_ciphers won't work for you)
# Grade A (A+ with HSTS at >= 6 Months)
# 90 % Security
# Medium Compatibility
#   No Java 6 (No DH parameters > 1024 bits)
#   No IE on XP
# Robust Forward Secrecy
#ssl_ciphers 'ECDHE-RSA-AES128-GCM-SHA256:ECDHE-ECDSA-
AES128-GCM-SHA256:ECDHE-RSA-AES256-GCM-SHA384:ECDHE-ECDSA-
AES256-GCM-SHA384:DHE-RSA-AES128-GCM-SHA256:DHE-DSS-AES128-GCM-
SHA256:kEDH+AESGCM:ECDHE-RSA-AES128-SHA256:ECDHE-ECDSA-AES128-
SHA256:ECDHE-RSA-AES128-SHA:ECDHE-ECDSA-AES128-SHA:ECDHE-RSA-
AES256-SHA384:ECDHE-ECDSA-AES256-SHA384:ECDHE-RSA-AES256-
SHA:ECDHE-ECDSA-AES256-SHA:DHE-RSA-AES128-SHA256:DHE-RSA-
AES128-SHA:DHE-DSS-AES128-SHA256:DHE-RSA-AES256-SHA256:DHE-DSS-
AES256-SHA:DHE-RSA-AES256-
SHA:!aNULL:!eNULL:!EXPORT:!DES:!RC4:!3DES:!MD5:!PSK';

```

```

# Diffie-Hellman parameter for DHE ciphersuites,
recommended 2048 bits
ssl_dhparam /etc/nginx/ssl/dhparams.pem;

# Specifies a curve for ECDHE ciphers.
ssl_ecdh_curve secp521r1;

# Fix for Windows phones before 8.1 Update 1
# Slightly lower security.
#ssl_ecdh_curve secp384r1;

# Server should determine the ciphers, not the client
ssl_prefer_server_ciphers on;

# OCSP Stapling
# fetch OCSP records from URL in ssl_certificate and
cache them
ssl_stapling on;
ssl_stapling_verify on;
ssl_trusted_certificate
/etc/letsencrypt/live/owncloud9tutorial.goip.de/fullchain.pem;

# SSL session handling
ssl_session_timeout 24h;
ssl_session_cache shared:SSL:50m;
ssl_session_tickets off;

#
# Add headers to serve security related headers
#

# HSTS (ngx_http_headers_module is required)
# In order to be recognized by SSL test, there must be an
index.html in the server's root
add_header Strict-Transport-Security "max-age=63072000;
includeSubdomains; preload";
add_header X-Content-Type-Options nosniff;
# Usually this should be "DENY", but when hosting sites
using frames, it has to be "SAMEORIGIN"
add_header X-Frame-Options "SAMEORIGIN";
add_header X-XSS-Protection "1; mode=block";
add_header X-Robots-Tag none;
add_header X-Download-Options noopen;
add_header X-Permitted-Cross-Domain-Policies none;

location ^~ /owncloud {
    # Set max. size of a request (important for uploads
to ownCloud)
    client_max_body_size 1G;
    # Besides the timeout values have to be raised in
nginx' owncloud config, these values have to be raised for the
proxy as well
    proxy_connect_timeout 300;
    proxy_send_timeout 300;
    proxy_read_timeout 300;
    send_timeout 300;
    proxy_set_header Host $host;
    proxy_set_header X-Real-IP $remote_addr;
    proxy_pass http://127.0.0.1:82;
    proxy_redirect off;
}
}

```

- Virtuellen Host für ownCloud anlegen:
  - nano /etc/nginx/sites-available/owncloud9tutorial.goip.de\_owncloud (Anpassungen *server\_name/IP an die eigene Konfiguration notwendig*)

```

upstream php-handler {
    server unix:/run/php/php7.0-fpm.sock;
}

server {
    listen 82;
    server_name 127.0.0.1;

    # Add headers to serve security related headers
    # Use 'proxy_set_header' (not 'add_header') as the headers
    have to be passed through a proxy.
    proxy_set_header Strict-Transport-Security "max-
age=15768000; includeSubDomains; preload;";
    proxy_set_header X-Content-Type-Options nosniff;
    proxy_set_header X-Frame-Options "SAMEORIGIN";
    proxy_set_header X-XSS-Protection "1; mode=block";
    proxy_set_header X-Robots-Tag none;
    proxy_set_header X-Download-Options noopen;
    proxy_set_header X-Permitted-Cross-Domain-Policies none;

    # Path to the root of your installation
    root /var/www/;

    location = /robots.txt {
        allow all;
        log_not_found off;
        access_log off;
    }

    # The following 2 rules are only needed for the
    user_webfinger app.
    # Uncomment it if you're planning to use this app.
    #rewrite ^/.well-known/host-meta
    /owncloud/public.php?service=host-meta last;
    #rewrite ^/.well-known/host-meta.json
    /owncloud/public.php?service=host-meta-json last;

    location = /.well-known/carddav { return 301
    $scheme://$host/owncloud/remote.php/dav; }
    location = /.well-known/caldav { return 301
    $scheme://$host/owncloud/remote.php/dav; }

    location /.well-known/acme-challenge { }

    location ^~ /owncloud {

        # set max upload size
        client_max_body_size 1G;
        fastcgi_buffers 64 4K;

        # Disable gzip to avoid the removal of the ETag header
        gzip off;

        # Uncomment if your server is build with the
        ngx_pagespeed module
        # This module is currently not supported.
        #pagespeed off;

```

```

error_page 403 /owncloud/core/templates/403.php;
error_page 404 /owncloud/core/templates/404.php;

location /owncloud {
    rewrite ^ /owncloud/index.php$uri;
}

location ~
^/owncloud/(?::build|tests|config|lib|3rdparty|templates|data)/
{
    deny all;
}
location ~
^/owncloud/(?::\.|autotest|occ|issue|indie|db_|console) {
    deny all;
}

location ~
^/owncloud/(?::index|remote|public|cron|core/ajax/update|status|
ocs/v[12]|updater/.+|ocs-
provider/.+|core/templates/40[34])\.php(?:$|/) {
    include fastcgi_params;
    fastcgi_split_path_info ^(.+\.(php))(/.+)$;
    fastcgi_param SCRIPT_FILENAME
$document_root$fastcgi_script_name;
    fastcgi_param PATH_INFO $fastcgi_path_info;

    # Important: disable HTTPS, otherwise no log in will
be possible!
    #fastcgi_param HTTPS on;

    fastcgi_param modHeadersAvailable true; #Avoid
sending the security headers twice
    fastcgi_param front_controller_active true;
    fastcgi_pass php-handler;
    fastcgi_intercept_errors on;

    # Raise timeout values.
    # This is especially important when the
ownCloud setup runs into timeouts (504 gateway errors)
    fastcgi_read_timeout 300;
    fastcgi_send_timeout 300;
    fastcgi_connect_timeout 300;

    # Pass PHP variables directly to PHP.
    # This is usually done in the php.ini. For more
flexibility, these variables are configured in the nginx
config.

    # All the PHP parameters have to be set in one
fastcgi_param. When using more 'fastcgi_param PHP_VALUE'
directives, the last one will override all the others.
    fastcgi_param PHP_VALUE
"open_basedir=/var/www:/tmp:/var/owncloud_data:/dev/urandom
    upload_max_filesize = 1G
    post_max_size = 1G
    max_execution_time 3600";
}

location ~ ^/owncloud/(?::updater|ocs-provider)(?:$|/) {
    try_files $uri/ =404;
    index index.php;
}

```

```

        # Adding the cache control header for js and css files
        # Make sure it is BELOW the PHP block
        location ~* \.(?:css|js)$ {
            try_files $uri
/owncloud/index.php$uri$is_args$args;
            proxy_set_header Cache-Control "public, max-
age=7200";
            # Add headers to serve security related headers
            # Again use 'proxy_set_header' (not 'add_header')
as the headers have to be passed through a proxy.
            proxy_set_header Strict-Transport-Security "max-
age=15768000; includeSubDomains; preload;";
            proxy_set_header X-Content-Type-Options nosniff;
            proxy_set_header X-Frame-Options "SAMEORIGIN";
            proxy_set_header X-XSS-Protection "1; mode=block";
            proxy_set_header X-Robots-Tag none;
            proxy_set_header X-Download-Options noopen;
            proxy_set_header X-Permitted-Cross-Domain-Policies
none;
            # Optional: Don't log access to assets
            access_log off;
        }

        location ~*
\.(?:svg|gif|png|html|ttf|woff|ico|jpg|jpeg)$ {
            try_files $uri
/owncloud/index.php$uri$is_args$args;
            # Optional: Don't log access to other assets
            access_log off;
        }
    }
}

```

- o ln -s /etc/nginx/sites-available/owncloud9tutorial.goip.de\_owncloud /etc/nginx/sites-enabled/owncloud9tutorial.goip.de\_owncloud
- o nginx -t
- o service nginx restart
- **Konfiguration PHP:**
  - o nano /etc/php/7.0/fpm/pool.d/www.conf
    - Kommentare der Umgebungsvariablen entfernen (nach *Pass environment variables like LD\_LIBRARY\_PATH. ALL &VARIABLES are taken from the current environment.* suchen)
  - o nano /etc/php/7.0/fpm/php.ini
    - cgi.fix\_pathinfo = 0
    - open\_basedir = /var/www/:/tmp/
  - o nano /etc/php/7.0/cli/php.ini
    - cgi.fix\_pathinfo = 0
    - open\_basedir = /var/www/:/tmp/:/var/oc\_data
  - o service php7.0-fpm restart
- **Konfiguration MariaDB:**
  - o mysql\_secure\_installation
  - o Alle Frage (bis auf Ändern des Root-Passwortes) mit Ja (y) beantworten
  - o nano /etc/mysql/my.cnf
    - binlog\_format = MIXED (unter [mysqld])
  - o service mysql restart



- Download ownCloud + Verzeichnisrechte (jeweils aktuelle Version von ownCloud verwenden):
  - `wget https://download.owncloud.org/community/owncloud-9.0.2.tar.bz2`
  - `tar -xjf owncloud-9.0.2.tar.bz2 -C /var/www`
  - `rm owncloud-9.0.2.tar.bz2`
  - `chown -R www-data:www-data /var/www/owncloud`
  - `chown -R www-data:www-data /var/oc_data`
- Datenbank anlegen (Anpassungen bzgl. Datenbankname und User notwendig):
  - `mysql -u root -p`
  - `create database owncloud_db;`
  - `create user owncloud_db_user@localhost identified by 'MeInPasSw0rT';`
  - `grant all privileges on owncloud_db.* to owncloud_db_user@localhost;`
  - `flush privileges;`
  - `exit;`
- ownCloud Setup
  - Im Browser <https://owncloud9tutorial.goip.de/owncloud> aufrufen
  - Admin-Konto anlegen (mit den zuvor eingegebenen Daten: Datenverzeichnis, Datenbankname, Datenbank-Benutzer, Datenbank-Passwort)
- ownCloud Konfiguration anpassen:
  - `nano /var/www/owncloud/config/config.php` (*instanceid*, *passwordsalt* und *secret* ausgelassen; Anpassungen an eigene IP/DynDNS-Adresse notwendig)

```
<?php
$CONFIG = array (
  'instanceid' => '...',
  'passwordsalt' => '...',
  'secret' => '...',
  'trusted_domains' =>
  array (
    0 => 'owncloud9tutorial.goip.de',
    1 => '192.168.178.20',
  ),
  'datadirectory' => '/var/owncloud_data',
  'overwrite.cli.url' =>
'http://owncloud9tutorial.goip.de/owncloud',
  'dbtype' => 'mysql',
  'version' => '9.0.2.2',
  'dbname' => 'owncloud_db',
  'dbhost' => 'localhost',
  'dbtableprefix' => 'oc_',
  'dbuser' => 'owncloud_db_user',
  'dbpassword' => '...',
  'logtimezone' => 'UTC',
  'installed' => true,
  'logtimezone' => 'Europe/Berlin',
  'memcache.local' => '\OC\Memcache\APCu',
);
```

- Zugriffsrechte anpassen:
  - `find /var/www/owncloud/ -type f -print0 | xargs -0 chmod 0640`
  - `find /var/www/owncloud/ -type d -print0 | xargs -0 chmod 0750`

- o `chown -R root:www-data /var/www/owncloud/`
- o `chown -R www-data:www-data /var/www/owncloud/apps/`
- o `chown -R www-data:www-data /var/www/owncloud/config/`
- o `chown -R www-data:www-data /var/www/owncloud/themes/`
- o `chown -R www-data:www-data /var/owncloud_data/`
- o `chown root:www-data /var/www/owncloud/.htaccess`
- o `chown root:www-data /var/owncloud_data/.htaccess`
- o `chmod 0644 /var/www/owncloud/.htaccess`
- o `chmod 0644 /var/owncloud_data/.htaccess`

- **Cronjob einrichten:**

- o `crontab -u www-data -e`

```
*/15 * * * * php -f /var/www/owncloud/cron.php > /dev/null  
2>&1
```

- o **Im ownCloud-Admin-Bereich (Abschnitt Cron) von AJAX auf Cron umstellen**