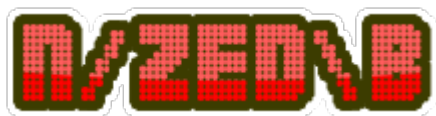


Usenet Indexer Installation - Redhat / CentOS 7

Im folgenden, wird ein neuer, eigener nZEDb Usenet Indexer mit mindestens 12 GB RAM aufgesetzt und Konfiguriert. nZEDb ist ein open-source (GPL-Lizenzierter) Usenet Indexer.



Voraussetzungen

- Fertig installiertes **RHEL 7 / CentOS 7** System mit **statischer IP**
- **Minimum 12 GB verbaute / in VM allozierte RAM!**
- Putty Client

<https://github.com/nZEDb/nZEDb/wiki/Install-Guide:-CentOS-7>

<https://github.com/nZEDb/nZEDb/wiki/Install-Guide:-Fedora>

Installation des Indexers



System Vorbereitungen

1. Hinzufügen von benötigten Repos und aktualisieren des Systems:

```
#  
# yum-config-manager --add-  
repo=http://negativo17.org/repos/epel-multimedia.repo  
  
# yum update -y
```

2. Installation der benötigten Grundpakete:

```
# yum groupinstall "Development Tools" "Development Libraries"  
# yum install yasm git p7zip tmux time composer mediainfo wget vim curl  
ffmpeg boost-regex boost-devel swig gcc-c++ httpd php56u-cli php56u-  
mcrypt php56u-mysqldb php56u-gd php56u-xml php56u-pear php56u-mbstring  
php56u-pecl-jsonc-devel php56u-process php56u-pecl-jsonc php56u-pecl-  
imagick php56u php56u-common php56u-pdo php56u-devel mariadb100u  
mariadb100u-common mariadb100u-libs mariadb100u-server mariadb100u-
```

```
config mariadb100u-errmsg perl-Text-Template libevent-devel ncurses-  
devel  
y
```

3. Weitere System Vorbereitungen:

```
# useradd -c nzedb nzedb  
# usermod -a -G nzedb apache
```

Konfiguration des Webservers

Im ersten Schritt, erstellen wir unter: **/etc/apache2/sites-available/** einen neuen Virtual-Host, für unseren späteren Usenet Indexer:

```
# vim /etc/apache2/sites-available/nzedb.conf
```

```
<VirtualHost *:80>  
    ServerName 127.0.0.1  
    ServerAdmin webmaster@localhost  
    ServerSignature Off  
  
    DocumentRoot "/var/www/nZEDb/www"  
  
    LogLevel warn  
    ErrorLog /var/log/apache2/error.log  
    Alias /nZEDb /var/www/nZEDb/www  
    Alias /covers /var/www/nZEDb/resources/covers  
    <Directory "/var/www/nZEDb/www">  
        Options FollowSymLinks  
        AllowOverride All  
        Require all granted  
    </Directory>  
  
    ProxyPassMatch ^/(.*\.php(/.*)?)$ unix:/run/php/php7.0-  
fpm.sock|fcgi://localhost/var/www/nZEDb/www  
</VirtualHost>
```



```
# sed -i "s/ServerTokens OS/ServerTokens Prod/" /etc/apache2/conf-  
available/security.conf  
# sed -i "s/ServerSignature On/ServerSignature Off/" /etc/apache2/conf-  
available/security.conf
```

Nun wird als erstes der Default-Virtual-Host deaktiviert, unser neuer aktiviert und das Rewrite-Modul enabled:

```
# mkdir -p /var/www/nZEDb/www

# a2dissite 000-default
# a2ensite nzedb.conf

# a2enmod rewrite

# systemctl enable apache2
```



Konfiguration von PHP 7.0

We already installed all needed PHP extensions earlier. For best performance, we also need to configure some PHP settings. :

```
# php --ini |grep Loaded
```

```
Loaded Configuration File:      /etc/php/7.0/cli/php.ini
```

In our case, we have to make changes first to the `/etc/php/7.0/cli/php.ini` file. We will increase the default upload limit to 2000 MB. You can set the values according to your needs. Run the following commands:

```
# sed -i "s/max_execution_time = 30/max_execution_time = 120/"
/etc/php/7.0/cli/php.ini
# sed -i "s/;date.timezone =/date.timezone = Europe\Zurich/"
/etc/php/7.0/cli/php.ini
```

Now we have to do the same for our php7.0-fpm Configuration:

```
# sed -i "s/max_execution_time = 30/max_execution_time = 120/"
/etc/php/7.0/fpm/php.ini
# sed -i "s/memory_limit = 128M/memory_limit = 1024M/"
/etc/php/7.0/fpm/php.ini
# sed -i "s/;date.timezone =/date.timezone = Europe\Zurich/"
/etc/php/7.0/fpm/php.ini
```

```
# echo "<?php phpinfo(); ?>" > /var/www/nZEDb/www/phpinfo.php
# systemctl start php7.0-fpm
# systemctl enable php7.0-fpm

# a2enmod proxy_fcgi
# systemctl restart apache2
```

Check PHP-Configuration at: [IP-From-SERVER/phpinfo.php](#)

Konfiguration Mariadb

Erstellen einer neuen MariaDB spezifischen Konfiguration:

```
# vim /etc/mysql/mariadb.conf.d/50-server.cnf
```

```
# configurations by nZEDb
innodb_file_per_table = 1
max_allowed_packet = 16M
group_concat_max_len = 8192
```

Nach dem Speichern, wird im nächsten Schritt die MariaDB enabled und gestartet!

```
# systemctl start mysql.service
# systemctl enable mysql.service
```

Wichtig: Das anschließende Hardening sollte auf produktiv-Systemen keinesfalls übersprungen werden!!

```
# /usr/bin/mysql_secure_installation

Enter current password for root (enter for none): Enter
Set root password? [Y/n]: Y
New password: <your-password>
Re-enter new password: <your-password>
Remove anonymous users? [Y/n]: Y
Disallow root login remotely? [Y/n]: Y
Remove test database and access to it? [Y/n]: Y
Reload privilege tables now? [Y/n]: Y
```

Erstellen der Datenbank & Konfiguration von AppArmor

Erstellen der von nZEDb benötigten Datenbank und Users:

```
mysql -u root -p

MariaDB [(none)]> CREATE DATABASE nzedb;
MariaDB [(none)]> GRANT ALL PRIVILEGES ON nzedb.* TO 'nzedbuser'@'localhost'
IDENTIFIED BY 'YOUR-PASSWORD';
MariaDB [(none)]> GRANT FILE ON *.* TO 'nzedbuser'@'localhost';
MariaDB [(none)]> FLUSH PRIVILEGES;
MariaDB [(none)]> quit
```

Installation der Media Tools

1. Installation der packetierten Media-Tools:

```
# apt-get install time unrar p7zip-full mediainfo lame ffmpeg libav-tools
```

2. Installation von yEnc, um das header und message processing durch das Indexing zu beschleunigen:

```
# mkdir /tmp/yenc
# cd /tmp/yenc
# wget
https://ayera.dl.sourceforge.net/project/yydecode/yydecode/0.2.10/yydecode-0.2.10.tar.gz
# tar xzf yydecode-0.2.10.tar.gz
# cd yydecode-0.2.10

# ./configure
# make
# make install
# cd /tmp
# rm -rf yenc
```

3. Installation der php-yenc extension, für noch schnelleres Encoding:

```
# wget
https://github.com/niel/php-yenc/releases/download/v1.3.0/php7.0-yenc_1.3.0_amd64.deb

# dpkg -i php7.0-yenc_1.3.0_amd64.deb
```

Download und Installation von nZEDb

```
# rm -Rf /var/www/nZEDb/
# cd /var/www/
# git clone https://github.com/nZEDb/nZEDb.git

# chown -R www-data:www-data /var/www/nZEDb
# chmod -R g+w /var/www/nZEDb/

# cp /var/www/nZEDb/_install/install.sh /var/www/
# chmod a+x /var/www/install.sh

# ./install.sh
```

During the installation, Composer will install all needed PHP dependencies and you will need to enter your sudo password to set permissions. Once the script finished, we need to set up the web server.

Tmux Version 2 installieren

- <https://forums.nzedb.com/index.php?topic=2464.0>

Wichtige Arbeiten, von vor dem konfigurieren / einrichten von nZEDb

- CA Zertifikat installieren:

```
# wget -P /etc/ssl/certs/ http://curl.haxx.se/ca/cacert.pem
# chmod 744 /etc/ssl/certs/cacert.pem
```

- Environment Ergänzungen:

```
# mysql_tzinfo_to_sql /usr/share/zoneinfo | sudo mysql -u root -p mysql

# mkdir -p /var/www/nZEDb/resources/tmp/unrar
# cp /var/www/nZEDb/configuration/settings.example.php
/var/www/nZEDb/configuration/settings.php
# chown www-data:www-data /var/www/nZEDb/configuration/settings.php

# chmod 777 /var/www/nZEDb/resources
```

nZEDb konfigurieren / Einrichten

server-ip/install

or

domain.com/install

Step1 → System Check should be OK! Step2 → Setup Database! Step3 → Enter CA-Cert Path: → /etc/ssl/certs/cacert.pem Step4 → Configure primary News-Server! ..

Abschliessende Schritte zur Inbetriebnahme des Indexers

Web-GUI Konfiguration des frisch installierten Indexers



Systemd Unit-File erstellen zur Steuerung der Background Services

```
# cat >/etc/systemd/system/nzedb.service <<EOD
[Unit]
Description=nzedb background service
After=syslog.target

[Service]
```

```
WorkingDirectory=/var/www/nZEDb/misc/update/nix/tmux
ExecStart=/bin/php start.php
ExecStop=/bin/php stop.php
Type=simple
```

```
[Install]
WantedBy=multi-user.target
EOD
```

```
# systemctl enable nzedb
# systemctl start nzedb
```

Weitere Indexer / Informationen

- Indexer mit Python: → <https://github.com/Murodese/pynab>
- nZEDb Indexer in Docker: → <https://hub.docker.com/r/slydetector/simply-nzedb/>
- Offizielle Dokumentation: → http://nzedb.readthedocs.io/en/master/general_guides.html

Last update: **2017/11/17 10:30**